UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Sede Amministrativa: Università degli Studi di Padova
Dipartimento di *Matematica*

SCUOLA DI DOTTORATO DI RICERCA IN : SCIENZE DELL'INGEGNERIA CIVILE ED AMBIENTALE
CICLO XXV

REDUNDANT MULTIRESOLUTION UNCERTAINTY PROPAGATION

Direttore della Scuola : Ch.mo Prof. Stefano Lanzoni

Supervisore :Ch.mo Prof. Mario Putti

Supervisore :Ch.mo Prof. Gianluca Iaccarino

Dottorando : Daniele Schiavazzi

Propagazione dell'incertezza tramite schemi multi-risoluzione a base ridondante

Sommario

Metodi non intrusivi basati sull'espansione della risposta di un dato sistema nello spazio dei parametri (Chaos expansion methods) consentono di risolvere equazioni differenziali stocastiche con un numero di soluzioni deterministiche minori rispetto ad approcci tradizionali alla Monte Carlo con campionamento classico o stratificato.

In tale ambito gli sforzi di ricerca odierni sono volti allo sviluppo di metodologie atte alla riduzione del costo computazionale in problemi caratterizzati da alta dimensionalitá (numero significativo di variabili aleatorie in input) ed al trattamento di problemi con risposta discontinua nello spazio dei parametri.

La ricerca condotta si é concentrata sull'utilizzo di recenti tecniche di *Compressive Sampling* per la minimizzazione del numero di soluzioni deterministiche necessarie alla ricostruzione di risposte dotate di sparsitá secondo un pre-definito dizionario di basi. Inoltre, tecniche di approssimazione multi-risoluzione sono state estese a metodologie non intrusive di propagazione dell'incertezza. Infine, tecniche di Importance Sampling sono state utilizzate per determinare in modo adattativo l'ubicazione di nuovi samples al fine di cogliere le scale maggiormente importanti nelle risposte approssimate.

Le metodologie approfondite ed implementate nell'ambito della ricerca svolta sono state applicate ad un insieme di funzioni analitiche, sistemi descritti da equazioni differenziali stocastiche, sistemi dinamici con risposte caratterizzate da elevati gradienti o discontinuitá, problemi ingegneristici con particolare riferimento all'ottimizzazione robusta della performance aerodinamica di profili per pale eoliche e sistemi passivi di smorzamento delle vibrazioni operanti sotto incertezza.

Vengono inoltre presentate metodologie atte a ripristinare doti di conservazione di massa in flussi numerici e sperimentali.

Parole chiave

Quantificazione dell'incertezza, Metodi non intrusivi di propagazione dell'incertezza, Alpert multiwavelets, Campionamento per importanza, Compressed Sensing, Approssimazioni multirisoluzione, Espansioni secondo chaos polinomiale, Equazioni differenziali stochastiche, Stochastic Collocation, filtri a divergenza nulla.

Redundant Multiresolution Uncertainty Propagation

Abstract

Stochastic partial differential equations can be efficiently solved using collocation approaches combined with polynomial expansion in parameter space. Estimators based on these concepts show smaller variance than traditional or stratified Monte Carlo approaches under mild dimensionality.

Research efforts in this context are focused on improving the efficiency of these methodologies for high dimensional problems (increasing number of input random variables) or for problems with discontinuous response in parameter space.

In the present work, we use *Compressive Sampling* in order to minimize the number of deterministic computations needed to evaluate expansion coefficients for stochastic responses which are sparse in selected dictionaries of basis. Moreover, multiresolution approximation techniques are extended in the context of non-intrusive uncertainty propagation. Finally, an adaptive Importance Sampling strategy is used where samples are iteratively added to locations containing relevant features of increasingly smaller size.

Applications are presented for analytical functions, stochastic differential equations, dynamical systems whose response is discontinuous or characterized by large gradients. Engineering problems involving robust optimization of windmill airfoils and passive damping of structures under uncertainty are also discussed.

The last Chapter is devoted to methodologies aiming to restore element conservativeness for numerical and experimental velocity fields.

Keywords

Uncertainty quantification, Non intrusive uncertainty propagation, Alpert multiwavelets, Importance Sampling, Compressed Sensing, Multiresolution approximation, Polynomial Chaos expansion, Stochastic partial differential equations, Stochastic Collocation, divergence-free filtering.

...a volte basta un Complice e tutto è già più semplice!
Brava Giulia.
Vasco Rossi - C'è chi dice no, 1987.

Ringraziamenti

Questo lavoro di tesi corona un'esperienza magnifica di ricerca spesa in uno dei luoghi maggiormente dinamici e attivi al mondo. Esso si colloca, inoltre, al culmine di una riflessione personale che, in questo ambiente, ha trovato modelli ed ispirazione.

Molti sono i ringraziamenti dovuti.

Il primo e più profondo ringraziamento va a mia moglie Giulia, cui questa tesi è dedicata, per il continuo supporto, la pazienza, e l'aiuto a perseverare in percorsi non sempre facili da intraprendere. Per l'amore e la cura che mette nelle cose e per la simpatia con cui decora tutte le mie giornate, rendendo Casa dovunque siamo.

L'ordinamento della società, in particolare nei paesi caratterizzati dalle maggiori contraddizioni, limita la reale flessibilità nelle professioni intese come accrescimento di competenze e voglia di cimentarsi con problemi di reale complessità tecnica. In questo senso, risulta a volte difficile seguire le proprie inclinazioni e passioni senza qualcuno di altrettanto entusiasta ed appassionato, qualcuno che ha già percorso gli stessi sentieri, che sia di consiglio e di aiuto. Per questo ringrazio sinceramente il mio relatore presso Stanford University, Prof. Gianluca Iaccarino, per la sua disponibilità, modernità ed entusiasmo.

Al mio relatore in Italia, Prof. Mario Putti, un grazie per la costante gentilezza e pazienza, anche nei momenti di maggiore turbolenza di questo mio "flashback" universitario.

Un sentito ringraziamento va, inoltre, al Prof. Alireza Doostan di UCB per la gentile ospitalità presso Bolder, Colorado e la qualità della collaborazione sfociata nel presente lavoro di tesi.

Durante il periodo di visita presso Stanford University ho tratto grande beneficio dalla collaborazione o da semplici scambi di idee con Gary Tang, Per Pettersson, Paul Constantine, Saman Ghili, Akshay Mittal e tutte le persone coinvolte nello UQ Lab. Sinceri ringraziamenti vanno a tutte le persone al Centre for Turbulence Research e Fluid Physics and Computational Engineering Department per l'amicizia e le lezioni giornaliere di impegno, curiosità ed attitudine positiva, tra cui Julien, Javier, Filippo, Dave, Riccardo, Joe, Hiroki, Ricardo, Catherine, Guido, Remi, Mihailo, tutti i docenti, il personale amministrativo e i visitatori. Ringrazio, in particolare, Marlene Lomuljo-Bautista per l'efficienza e la simpatia nella gestione amministrativa del periodo trascorso negli US.

Un sentito Grazie ai miei genitori Nelly e Vittorio, che mi hanno sempre indirizzato verso la migliore possibile educazione, e a mio zio Giuseppe per la costante gentilezza e la vicinanza alla nostra famiglia.

Un sincero ringraziamento va a Nadia, Enrico e Matteo per il costante supporto ed incoraggiamento, oltre all'ineguagliabile ospitalità (★ ★ ★ ★ ★ L).

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Background

1.1 Uncertainty in engineering systems

It appears clear even to the youngest engineer how calculations and modeling are inevitably affected by uncertainty. Example in this regard are: insufficient availability of data, approximation in estimating physical quantities of interest, dispersion of sampling in experimental results, lack of knowledge of the basic mechanisms of a physical system and approximation in evaluating th correct boundary conditions. The reduction of uncertainty in judgment through experience or systematic application of rigorous principles of mathematical physics is therefore the primary challenge of a professional engineer. Understanding the possible implications of working with quantities that are not always fully known in advance is of paramount importance in design, verification and manufacturing.

First of all, an engineering design should be *robust*, meaning that it should have the best possible performance across a range of possible variabilities in the operating conditions. This simple statement allows us to introduce two new concepts, namely an *average performance* over an ensemble of possible system's states and the *variance*, i.e. the sensitivity, to changes in this conditions. Therefore, *probabilistic* concepts like average value or variance are naturally introduced when uncertainty is embraced and *measures* associated to quantities of interest (e.g. performance) become important to formulate educated engineering judgments.

The same concepts above apply when verifying selected designs. Note that this can happen many years after the original design was formulated and the same properties of the system, once accurately known, may become uncertain. Even if widely used in verification codes, heuristic factors accounting for lack of knowledge in the geometry or material properties might fail to appropriately quantify the reliability of a given system. Probabilistic descriptions of loads or operating conditions also allow to mitigate the effect of rare events by accounting for their probability of occurrence. Estimates of expected life can be therefore formulated in a much more informative way, resulting in significant cost reduction.

As a last example, it is important to highlight the role played by uncertainty in manufacturing and experimental testing. A low *dispersion* in the properties of products is a key aspect of series production and the possibility of obtaining two identical realization of the same experiment is practically zero even for the most careful setup.

Due to its fundamental role in engineering, an effort is required to foster the systematic quantification of uncertainty in design, through accessible methodologies and tools providing a broader understanding on how it affects physical phenomena.

1.2 Motivation

To motivate the present work, we fist define the problem we are trying to solve, together with the associated assumptions; "Uncertainty quantification" is in addition a general term and it is used for a wide range of problems.

Even a complex engineering system can be schematically visualized through an input-output mapping. This representation is valid both for systems responding through simple analytical formulae or described by a system of coupled partial differential equations (PDE). Uncertainty permeates every aspect of the problem, from quantifying the distribution and correlation of the input random variables/processes/fields or system's coefficients (e.g. conductivity, elasticity, etc.), to the problem of efficiently propagating the latter input quantities to the system's outcome. In particular, this last problem is traditionally addresses as *uncertainty propagation* (UP); it spans from evaluating the statistical moments of a function of random variables to solving a system of stochastic differential equations.

In the present study, we are mainly interested in the uncertainty propagation from input data to output responses. In the following, we therefore assume all input variables and system coefficients to be completely defined in probability, space and time. In case of correlation for the above variables, we also assume this to be completely known.

The widely used Monte Carlo approach can be considered as a reference methodology for UP. It is a fair blend of simplicity and robustness. It is easy to implement and it naturally handles systems characterized by discontinuous stochastic responses. Besides, its convergence is proportional to the squared root of the number of samples but it does not depend, in general, on the number of uncertain input parameters, becoming of great appeal for high dimensional stochastic problems.

Various approaches have been developed in literature with the aim of improving the rate of convergence of the Monte Carlo method. For example, approaches based on polynomial chaos approximations, lead to improvements in the convergence rate for sufficiently smooth responses, failing to do so if applied to discontinuous problems. On the other hand, given a fixed approximation order, the number of terms needed for polynomial expansions grows significantly with the dimensionality of the problem.

Our efforts are therefore in the direction of developing methodologies performing better than Monte Carlo strategies and at the same time applicable to both discontinuous and high dimensional problems. To do so, statistical regression is tackled in two separate contexts. In the first (*expansion*) a dictionary of basis is selected providing sparse representation for piecewise smooth signals. In the second (*reconstruction*), the Compressive Sampling paradigm is used to minimize the number of samples needed for accurately approximate stochastic responses.

1.3 Types of Uncertainty

Before introducing more rigorous concepts and methodologies to handle uncertainty related to engineering applications, we must ask ourself what "uncertainty" is and about the possible ways in which it affects our problems. We therefore identify two types of uncertainty which are relevant in the present developments, i.e., aleatoric and epistemic uncertainty.

Aleatoric uncertainty relates to the complexity and intrinsic variability of nature. As an example, it is almost impossible to obtain exactly the same result by successive repetitions of the same physical experiment. This leads to the statistical characterization of quantities popular in engineering where,

for example, material strength, elastic modulus, wind forces, external vibrations are all associated to a given probability distribution typically deduced from experimental data or field measurements.

Other examples of Aleatory uncertainty are natural randomness, value diversity, behavioral variability, social randomness, technological surprise.

Epistemic or systematic uncertainty is typically introduced by engineering surrogates (i.e. numerical models, indirect measurements, etc.) trying to quantify a physical phenomenon.

Other examples of Epistemic uncertainty are inexactness, lack of observation, conflicting evidence, ignorance, indeterminacy.

While our developments mainly focus on Aleatoric uncertainty, it is worth noting that interactions between these two aspects might be observed. As an example, an analytic approximation of a given law of nature might not have the same accuracy over the whole parameter space. This means that, for different ranges of parameters, the contribution of Aleatoric and Epistemic uncertainty might be different. This might require a careful selection of the propagation methodology, as non-intrusive or Monte Carlo-like approaches might not always work in this case.

1.4 Road map

The present document is organized as follows:

The first chapter offers a general introduction and motivation on the subject of uncertainty quantification, presenting the main ideas and discussing the assumptions that will hold throughout. It also introduces some basic concepts of probability theory and statistical inference as Monte Carlo estimation with associated sampling techniques. Spectral expansion techniques are introduced in Chapter 2, where polynomial chaos expansion is first illustrated in the context of functional expectation and successively applied to the solution of stochastic PDEs (sPDEs). Intrusive uncertainty propagation methodologies based on stochastic Galerkin projection are discussed for the solution of a transport equation with random wave speed. Sparse grid and regression techniques are presented in the context of non-intrusive uncertainty propagation. The last part of the Chapter is entirely devoted to a multiresolution generalization of a Fourier-Legendre expansion particularly well suited for interpolating piecewise continuous responses. Compressive sampling (CS) is presented in Chapter 3 as a way to approximate sparse stochastic responses using the least possible number of samples. The combination of CS with adaptive Importance sampling is also discussed, in an attempt of improve convergence to the output statistics for responses exhibiting sharp gradients or discontinuities. Sparsity-undersampling phase diagrams are built for various basis systems and sampling strategies. In Chapter 4 and 5, emphasis is given to the application of the proposed methodologies to real engineering problems.

1.5 Rudiments of Probability Theory

Basic concepts are presented next, that will act as a reference for the developments in later Chapters. In particular we define probability spaces, random variables, convergence of r.v. and expectation, focusing on events generated from subspaces of $\mathbb{R}^d$.

1.5.1 Elements of measure theory in probability

First, a *sample space* Ω is the set of all possible outcomes $y^{(i)} \in \Omega$ with $i \in \mathbb{N}$.

We also define the *event space* $\mathcal{F}$ containing all possible sets of outcomes. In particular, if 2^Ω is defined as the set of all possible subset of Ω, then $\mathcal{F} \subseteq 2^\Omega$. The structure of a σ-algebra is assigned to $\mathcal{F}$ on the sample space Ω. Note that this extends the concept of an algebra closed respect to the union, intersection and complement operations, accounting for the union of infinitely many partitions of Ω. The *complement* of $A \subset \Omega$ in Ω is also defined as $A^c = \Omega \backslash A$.

Definition 1 (σ-algebra). $\mathcal{F} \subseteq 2^\Omega$ is a σ-algebra of Ω, if

1. $\Omega \in \mathcal{F}$
2. If $A \in \mathcal{F}$, then $A^c \in \mathcal{F}$.
3. If $A_i \in \mathcal{F}$ for $i = 1, 2, \ldots$, then $\bigcup_{i=1}^{\infty} A_i \in \mathcal{F}$.

In other words, a σ-algebra over a set Ω is a nonempty collection $\mathcal{F}$ of subsets of Ω (including Ω itself) that is closed under the complement and countable unions of its members. This concept allow us to define a *measurable space* as follows:

Definition 2 (Measurable space). A measurable space is a pair $(\Omega, \mathcal{F})$ where $\mathcal{F}$ is a σ-algebra of subsets of Ω.

Before introducing the concept of a *probability measure* we recall that $A_i \subset \Omega$, $i = 1, 2, \ldots$ are *disjoint sets* if $A_i \bigcap A_j = \{0\}$, $\forall i \neq j$.

Definition 3 (Probability measure). A probability measure $\mathbf{P}$ is a function $\mathbf{P} : \mathcal{F} \to [0, 1]$ with the following properties:

1. $\mathbf{P}(A) \in [0, 1]$, $\forall A \in \mathcal{F}$.
2. $\mathbf{P}(\Omega) = 1$.
3. $\mathbf{P}(A) = \sum_{i=1}^{\infty} \mathbf{P}(A_i)$ whenever $A = \bigcup_{i=1}^{\infty} A_i$ is a countable union of disjoint sets $A_i \in \mathcal{F}$.

We have now all the tools for the definition of a *probability space*.

Definition 4 (Probability space). A probability space is a triplet $(\Omega, \mathcal{F}, \mathbf{P})$, with $\mathbf{P}$ a probability measure on the measurable space $(\Omega, \mathcal{F})$.

We conclude the current section by introducing the concepts of *generated* σ-fields and *Borel* σ-fields in $\mathbb{R}$.

Definition 5 (Generated σ-fields). Given a collection of subset of $A_\alpha \subseteq \Omega$, where $\alpha \in \Gamma$ is a not necessarily countable index set, we define $\sigma(\{A_\alpha\})$ as the *smallest* σ-field $\mathcal{F}$ such that $A_\alpha \in \mathcal{F}$. We call $\sigma(\{A_\alpha\})$ the σ-field generated by the collection $\{A_\alpha\}$.

An example of generated σ-field is the *Borel* σ-field in $\mathbb{R}$ defined as $\mathcal{B} = \sigma(\{(a, b) : a, b \in \mathbb{R}\})$.

1.5.2 Random Variables

As a preliminary concept, we introduce the *image* of a function y defined over a measurable space $(\Omega,\mathcal{F})$ as $img(y) = \{y(\omega) : \omega \in \Omega\}$.

Definition 6 (Random Variable)**.** A random variable (r.v.) is an $\mathcal{F}$-measurable function $y : \Omega \to \mathbb{R}$ over the measurable space $(\Omega,\mathcal{F})$, that is:

$$\{\omega : y(\omega) \in \mathcal{B}\} \in \mathcal{F} \quad \forall \mathcal{B} \in \mathbb{R}$$

The *indicator* function $I_A(\omega)$ can be considered as the simplest example of a r.v.:

$$I_A(\omega) = \begin{cases} 1, \, \omega \in A \\ 0, \, \omega \notin A \end{cases} \tag{1.1}$$

if we apply the definition above, we see that:

$$\{\omega : y(\omega) \in \mathcal{B}\} = \begin{cases} A, \text{ if } \quad \mathcal{B} \geqslant 1 \\ A^c, \text{ if } \quad 0 \leqslant \mathcal{B} < 1 \\ \{0\}, \text{ if } \quad \mathcal{B} < 0 \end{cases} \in \mathcal{F} \tag{1.2}$$

We also introduce the concept of a *simple function* S which is a finite linear combination of indicator functions defined over subsets of $\mathcal{F}$. In other words:

$$S(\omega) = \sum_{n=1}^{N} c_n \, I_{A_n}(\omega) \tag{1.3}$$

It can be shown that for every r.v. $y(\omega)$ there exists a sequence of simple functions $y_n(\omega)$ such that $y_n(\omega) \to y(\omega)$ as $n \to \infty$, for each fixed $\omega \in \Omega$.

A this point, it is worthwhile to introduce the concept of *convergence* for r.v.s, in particular *almost sure* (a.s.), *almost everywhere* (a.e.) convergence as well as convergence *with probability 1* (w.p.1) that prevail through probability theory. They will be used interchangeably.

Definition 7 (Almost surely)**.** We say that the r.v. y and z defined over the same probability space $(\Omega,\mathcal{F},\mathbf{P})$ are almost surely the same if $\mathbf{P}(\omega : \{x(\omega) \neq y(\omega)\}) = 0$.

The *mathematical expectation* of a random variable $x(\omega)$, indicated with $\mathbb{E}\{x\}$ is a key concept in probability theory and is introduced as follows:

Definition 8 (Mathematical expectation)**.** Let's assume $x_{k,n} = k2^{-n}$ and the intervals $I_{n,k} = (x_{k,n}, x_{k+1,n}]$. For $k = 0, 1, \ldots$ we define the mathematical expectation of the random variable $y(\omega) \geqslant 0$ as:

$$\mathbb{E}\{y\} = \lim_{n\to\infty} \sum_{k=0}^{\infty} x_{k,n} \mathbf{P}(\{\omega : y(\omega) \in I_{k,n}\}) \tag{1.4}$$

When the range of $y(\omega)$ is countable, the definition above reduces to the elementary definition of expectation $\mathbb{E}\{y\} = \sum_i y^{(i)} p_i$, where $p_i = \mathbf{P}(\{\omega : y(\omega) = y^{(i)}\})$. Another important case where the expectation can be explicitly computed is when a r.v. is associated to a probability distribution function.

Definition 9 (pdf associated to r.v.). A r.v. $y(\omega)$ is associated with a probability density function (pdf) f_y if $\mathbf{P}(\{a \leqslant y \leqslant b\}) = \int_a^b f_y(y)\,dy$ for every $a \leqslant b \in \mathbb{R}$. Such function f_y must be not negative and $\int_{\mathbb{R}} y\, f_y(y)\,dy = 1$.

Therefore, for a non-negative r.v. y with an uncountable range associated to a pdf f_y the given definition of expectation coincides with the elementary formula $\mathbb{E}(y) = \int_0^\infty y\, f_y\, dy$. This could be also referred to as the Lebesgue integral *respect to the probability measure* $\mathbf{P}$. We also note that a random variable y of arbitrary sign can be decomposed into the difference of two non-negative r.v.s $y_+ = max(y, 0)$ and $y_- = -min(0, y)$, i.e. $y = y_+ - y_-$. In this case we say that the r.v. y is *integrable* if $\int_{-\infty}^{+\infty} |y| f_y\, dy < \infty$ and in such a case we have that $\mathbb{E}\{y\} = \int_{-\infty}^{+\infty} y\, f_y\, dy$.

We conclude this section by recalling the properties of expectation, together with some inequalities used to bound probabilities and expected values.

The following properties hold for the expectation operator:

1. $\mathbb{E}\{\mathcal{I}_A\} = \mathbf{P}(A) \quad \forall A \in \mathcal{F}$.
2. If $y(\omega) = \sum_{n=1}^N c_n \mathcal{I}_{A_n}$ is a simple function, then $\mathbb{E}\{y\} = \sum_{n=1}^N c_n \mathbf{P}(A_n)$.
3. For integrable r.v.s y, z the expectation is a linear operator, i.e. $\mathbb{E}\{\alpha y + \beta z\} = \alpha\mathbb{E}\{y\} + \beta\mathbb{E}\{z\}$.
4. $\mathbb{E}\{y\} = c$ if $y(\omega) = c$ with probability 1.
5. If $y \geqslant z$ a.s., then $\mathbb{E}(y) \geqslant \mathbb{E}(z)$. Further, if $y \geqslant z$ a.s. and $\mathbb{E}(y) = \mathbb{E}(z)$, then $y = z$ a.s.

Theorem 1 (Markov's inequality). Suppose f is a non-descreasing, Borel measurable function with $f(x) > 0$ for any $x > 0$. Then, for any random variable y and all $\epsilon > 0$,

$$\mathbf{P}(|y(\omega)| \geqslant \epsilon) \leqslant \frac{1}{f(\epsilon)} \mathbb{E}\{f(|y|)\}. \tag{1.5}$$

Markov's inequality is often useful in connecting probabilities with expectations. We show two possible applications of this inequality with $f(x) = x$ and with $f(x) = x^2$, $y = z - \mathbb{E}\{z\}$, respectively. In the first case, for a given constant $a > 0$, we have:

$$\mathbf{P}(|y| \geqslant a) \leqslant \frac{\mathbb{E}\{|y|\}}{a}. \tag{1.6}$$

which gives us a loose upper bound on the CDF of y. As an example, consider the case where $a = \mathbb{E}\{y\}$. The expression above simply tells us that $\mathbf{P}(|y| \geqslant \mathbb{E}\{y\}) \leqslant 1$. We could use the second case to estimate the distance of a given random variable from its mean value:

$$\mathbf{P}(|z - \mathbb{E}\{z\}| \geqslant a) \leqslant \frac{\mathbb{E}\{|z - \mathbb{E}\{z\}|^2\}}{a^2} = \frac{Var\{z\}}{a^2}. \tag{1.7}$$

A sharper estimate in this regard, is provided by the McDiarmid's inequality.

Theorem 2 (McDiarmid's inequality). Let $\mathbf{y} = \{y_1, \ldots, y_n\}, \mathbf{z} = \{z_1, \ldots, z_n\}$ be vectors of independent r.v.s. which differ only for the i-th component, i.e. $y_j = z_j\ \forall j = 1, \ldots, n,\ j \neq i$. Suppose that $g : \mathbb{R}^n \to \mathbb{R}$ to be a function with associated coefficients c_i, $i = 1, \ldots, n$ such that:

$$\sup_{\mathbf{y},\mathbf{z}} |g(\mathbf{y}) - g(\mathbf{z})| \leqslant c_i \quad \text{for} \quad i = 1, \ldots, n. \tag{1.8}$$

Then

$$\mathbf{P}\left(g(\mathbf{y}) - \mathbb{E}\{g(\mathbf{y})\} \geqslant \epsilon\right) \leqslant \exp\left\{-\frac{2\epsilon^2}{\sum_{i=1}^n c_i^2}\right\} \tag{1.9}$$

The following inequality provides bounds on expected values.

Theorem 3 (Schwarz inequality). Suppose $y, z \in \Omega$ and both $\mathbb{E}\{y^2\}, \mathbb{E}\{y^2\} < \infty$, then

$$\mathbb{E}\{|yz|\} \leqslant \sqrt{\mathbb{E}\{y^2\}\,\mathbb{E}\{z^2\}} \tag{1.10}$$

1.5.3 Convergence of Random Variables

As asymptotic behavior (i.e. for sufficiently large samples) is a key issue in probability, the present section focuses on the most common notions of convergence for r.v.s and how they relate. Since the concept of convergence is closely related to that of limit, we need to make sure that limits of sequence of r.v.s are also random variables. We therefore introduce the concept of *complete* probability space and assume this property to be true in our developments hereafter.

Definition 10 (Complete probability space). We say that $(\Omega, \mathcal{F}, \mathbf{P})$ is a complete probability space if any subset S of $A \in \mathcal{F}$ such that $\mathbf{P}(A) = 0$ is also in $\mathcal{F}$.

Note that a σ-field is made complete by adding to it all the subset of sets of zero probability.

A strong form of convergence, i.e., point-wise convergence is presented first.

Definition 11 (Pointwise Convergence). Given a sequence of r.v. $y_i(\omega)$ for $i = 1, 2, \ldots$, we say that it converges *pointwise* to the variable $y(\omega)$ and we write $\lim_{i\to\infty} y_i(\omega) = y(\omega)$ or $y_i(\omega) \to y(\omega)$ if the latter expression is true for *all* $\omega \in \Omega$.

Pointwise convergence is usually not very useful as it is defined irrespectively from the measure or probability of the single values of ω. A slightly weaker but way more used form of convergence is the following.

Definition 12 (Almost sure convergence). Given a probability space $(\Omega, \mathcal{F}, \mathbf{P})$, we say the the sequence y_i of r.v.s converge to y *almost surely* (and we write $y_i \xrightarrow{a.s.} y$) if $\mathbf{P}(y_i \to y) = 1$.

Almost sure convergence (as pointwise convergence) is invariant under the application of a continuous map. In other words, if $y_i \xrightarrow{a.s.} y$ and $f : \mathbb{R} \to \mathbb{R}$ is a continuous function, then $f(y_i) \xrightarrow{a.s.} f(y)$.

The weaker notion of converge in probability is defined next.

Definition 13 (Convergence in probability). We say that y_i converges to y in probability and write $y_i \xrightarrow{p} y$ if $\mathbf{P}(\{\omega : |y_i(\omega) - y(\omega)| > 0\}) \to 0$ as $i \to \infty$, for any fixed $\epsilon > 0$.

The following relationships between convergence a.s. and convergence in probability hold:

- If $y_i \xrightarrow{a.s.} y$ then $y_i \xrightarrow{p} y$
- If $y_i \xrightarrow{p} y$ then there exists a subsequence i_k such that $y_{i_k} \xrightarrow{a.s.} y$ for $k \to \infty$.

We generalize the idea of convergence in probability by introducing the concepts of L^q spaces and convergence in q-mean.

Definition 14 (L^q space). For a fix $1 \leqslant q < \infty$ we denote as $L^q(\Omega, \mathcal{F}, \mathbf{P})$, or simply L^q the collection of r.v.s y defined over the measurable space $(\Omega, \mathcal{F})$ such that $\mathbb{E}\{|y|^q\} < \infty$

Examples in this regard are L^1, i.e., the space of integrable r.v.s and L^2, i.e., the space of *square integrable* r.v.s.

Definition 15 (Convergence in q-mean)**.** We say that y_n converges to y in q-mean or in L^q sense, and we write $y_n \xrightarrow{q.m.} y$, if $y_n, y \in L^q$ and $\|y_n - y\|_q \to 0$ for $n \to \infty$, that is, $\mathbb{E}\{|y_n - y|^q\} \to 0$ for $n \to \infty$.

An equivalence between convergence in q-mean and in probability can be stated as follows: if $y_n \xrightarrow{q.m.} y$ then $y_n \xrightarrow{p} y$.

The notions of *distribution* and *independence* are responsible for the difference between measure theory and probability theory; they are addressed next.

We start by the concept of *law* of a r.v. and associated convergence. This is actually the weaker form of convergence we have explored so far.

Definition 16 (Law of a r.v.)**.** The law of a random variable y, denoted $\mathcal{P}_y$ is the probability measure on $(\mathbb{R}, \mathcal{B})$ such that $\mathcal{P}_y(A) = \mathbf{P}(\{\omega : y(\omega) \in A\})$ for any Borel set A.

In other words, the law of a r.v. associates probabilities to subsets of the sample space Ω.

Definition 17 (Probability distribution function)**.** The *probability distribution function* F_y of a real valued r.v. is defined as:

$$F_y(\alpha) = \mathbf{P}(\{\omega : y(\omega) \leqslant \alpha\}) = \mathcal{P}_y((-\infty, \alpha]) \tag{1.11}$$

It is worthwhile to point out that the knowledge of the cumulative distribution function F_y uniquely determines $\mathcal{P}_y$. A measure of convergence different from the ones defined above can be introduced at this point:

Definition 18 (Convergence in law)**.** We say the a random variable y_n converges in law (or weakly, or in distribution) to y and we write $y_n \xrightarrow{\mathcal{L}} y$ if $F_{y_n}(\alpha) \to F_y(\alpha)$ as $n \to \infty$ for each α where $F_y(\alpha)$ is continuous.

We shall now introduce another important concept in probability, i.e., independence. We say that two events $A, B \in \mathcal{F}$ are $\mathbf{P}$-mutually independent if $\mathbf{P}(A \cap B) = \mathbf{P}(A)\mathbf{P}(B)$. This concept can be easily extended to any set of events $A_i \in \mathcal{F}$ of finite size. In particular, we can write

$$\mathbf{P}(A_{i_1} \cap A_{i_2} \cap \cdots \cap A_{i_L}) = \prod_{k=1}^{L} \mathbf{P}(A_{i_k}) \quad \text{where} \quad L < \infty. \tag{1.12}$$

Independence for random variables and random vectors can be derived from the concepts above, passing through the associated σ-fields.

Definition 19 ($\mathbf{P}$-independent σ-fields)**.** Two σ-fields $\mathcal{H}, \mathcal{G} \subset \mathcal{F}$ are $\mathbf{P}$-independent if

$$\mathbf{P}(G \cap H) = \mathbf{P}(G)\mathbf{P}(H), \quad \forall G \in \mathcal{G}, \forall H \in \mathcal{H}. \tag{1.13}$$

The random vectors $(y_1, y_2, \ldots, y_n)$ and $(z_1, z_2, \ldots, z_m)$ are independent if the corresponding σ-fields $\sigma(y_1, y_2, \ldots, y_n)$, $\sigma(z_1, z_2, \ldots, z_m)$ are independent.

Definition 20 (Uncorrelated r.v.s)**.** Two r.v.s $y, z \in L^2(\Omega, \mathcal{F}, \mathbf{P})$, i.e., defined on the same probability space, are called *uncorrelated* if $\mathbb{E}\{yz\} = \mathbb{E}\{y\}\mathbb{E}\{z\}$.

It follows that any two independent r.v.s $y, z \in L^2(\Omega, \mathcal{F}, \mathbf{P})$ are also uncorrelated.

1.6 Monte Carlo estimation

Monte Carlo estimation solves, using sampling methods, the problem of calculating expectations of complicated multi-dimensional functions of vectors of random variables characterized by complicated distribution functions. Consider a function $g(y) : \mathbb{R} \rightarrow \mathbb{R}$ of a singled-valued random variable y, characterized by a pdf f_y and a cdf F_y. If we assume we can generate samples from F_y, i.e., n realizations of the r.v. $y^{(j)}$, then we can approximate the quantity $I = \mathbb{E}\{g(y)\}$ using the following expression:

$$I_M = \frac{1}{M} \sum_{j=1}^{M} g(y^{(j)}) \tag{1.14}$$

We can alternatively refer to a new random variable $I_M(\mathbf{y}) = I_M(y_1, y_2, \ldots, y_M) = \frac{1}{M} \sum_{j=1}^{M} g(y_j)$ as a Monte Carlo *estimator* of I. The notation above uses $y^{(j)}$ as realizations of the r.v. y, while y_j is the j-th random variable in the domain of $I_M(\mathbf{y})$. Note that $I_M(\mathbf{y}) : \mathbb{R}^M \rightarrow \mathbb{R}$ is an unbiased estimator of I:

$$\mathbb{E}\{I_M(\mathbf{y})\} = \int_\Omega \left(\frac{1}{M} \sum_{j=1}^{M} g(y_j) \right) f_y \, dy = \frac{1}{M} \sum_{j=1}^{M} \int_\Omega g(y_j) \, f_y \, dy = \mathbb{E}\{g(y)\} = I, \tag{1.15}$$

with variance equal to:

$$Var\{I_M(\mathbf{y})\} = \mathbb{E}\left\{ \left(\frac{1}{M} \sum_{j=1}^{M} g(y_j) - I \right)^2 \right\} = \mathbb{E}\left\{ \frac{1}{M^2} \sum_{j=1}^{M} \left(g(y_j) - I \right)^2 \right\} = \frac{1}{M} Var\{g(y)\}, \tag{1.16}$$

and the following relationship can be established for the standard deviation of $I_M(\mathbf{y})$:

$$\sigma\{I_M(\mathbf{y})\} = \frac{1}{\sqrt{M}} \sigma\{g(y)\} \tag{1.17}$$

Note that in equation (1.16) we used the fact that $(g(y_i) - I)$ and $(g(y_j) - I)$ are two independent, zero average random variables. They are also uncorrelated, therefore $\mathbb{E}\{(g(y_i) - I)(g(y_j) - I)\} = \mathbb{E}\{(g(y_i) - I)\}\mathbb{E}\{(g(y_j) - I)\} = 0$.

Equation (1.17) is at the core of the Monte Carlo method and states that the deviation of the estimate reduces proportionally to $\sqrt{M}$. The convergence rate above might appear slow if compared to some of the results obtained using *Polynomial Chaos* (see next Chapter) where *exponential* (proportional to e^{-M}) rates are observed under conditions. However, note that the variance of the Monte Carlo estimator depends only on the number of samples, and not on the number of r.v.s considered for g. Moreover, the only required assumption is that of $y \in L^2(\Omega, \mathcal{F}, \mathbf{P})$, i.e., the r.v. y has finite variance.

1.7 Variance reduction in Monte Carlo estimation

In this section, we briefly review two well known strategies to obtain Monte Carlo estimators of reduced variance. In particular, we introduce Importance Sampling as it will be used in later chapters combined with information extracted from a wavelet representation.

1.7.1 Stratified Sampling

A stratified sampling estimator assumes the sample space to be partitioned into r disjoint subsets $\{\Gamma_1, \dots \Gamma_r\}$ associated with *known* probabilities $p_s = \Pr\{y \in \Gamma_s\}$, $s = 1, \dots, r$. By the total probability law, the exact expectation can be written as

$$\mathbb{E}\{g(\mathbf{y})\} = I = \sum_{s=1}^{r} p_s \mathbb{E}\{g(\mathbf{y}) | \mathbf{y} \in \Gamma_s\}. \tag{1.18}$$

Similarly, the stratified sampling estimator of I is

$$I_s = \sum_{s=1}^{r} p_s \left[\frac{1}{M_s} \sum_{j=1}^{M_s} g(y_j^s) \right] = \sum_{s=1}^{r} p_s \, I_{M,s} \tag{1.19}$$

where $y_j^s \in \Gamma_s$ are r.v.s drawn from the conditional probability density $f_y(y|y \in \Gamma_s)$, $I_{M,s}$ is a Monte Carlo estimator restricted to a single *strata* and $\sum_{s=1}^{r} M_s = M$. If we evaluate the variance of the stratified sampling estimator we have:

$$\sigma_s^2\{I_s(\mathbf{y})\} = \sum_{s=1}^{r} p_s^2 \frac{\sigma_{M,s}^2}{M_s} \tag{1.20}$$

We can now define $\alpha_s = M_s/M$ as the fraction of the samples in stratum s, with the constraint $\sum_{s=1}^{r} \alpha_s = 1$. The set $\boldsymbol{\alpha} = \{\alpha_s,\, s = 1, \dots, r\}$ contains probability masses that parametrize the sampling across subset of the entire domain. The problem of finding the best distribution of samples across strata, translates in a constrained minimization problem:

$$\text{find} \quad \boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha}} \, F = \arg\min_{\boldsymbol{\alpha}} \left[\sigma_{M,s}(\boldsymbol{\alpha}) + \lambda \left(\sum_{s=1}^{r} \alpha_s \right) \right] \tag{1.21}$$

By taking the first derivative of F, we have:

$$\frac{\partial F}{\partial \alpha_s} = -\frac{1}{M} p_s^2 \frac{\sigma_{M,s}^2}{\alpha_s^2} + \lambda \quad \rightarrow \quad \alpha_s^* = \frac{p_s \, \sigma_{M,s}}{\sqrt{M\lambda}} \tag{1.22}$$

We can also check that the optimal solution is a minimum, in fact:

$$\frac{\partial^2 F}{\partial \alpha_s^2} = \frac{2\, p_s^2 \, \sigma_{M,s}^2}{M \, \alpha_s^3} > 0 \tag{1.23}$$

Substitution in the constraint $\sum_{s=1}^{r} \alpha_s = 1$ gives:

$$\lambda = \frac{1}{M} \left(\sum_{s=1}^{r} p_s \sigma_{M,s} \right)^2 \quad \text{and} \quad \alpha_s = \frac{p_s \, \sigma_{M,s}}{\sum_{j=1}^{r} p_j \, \sigma_{M,j}} \tag{1.24}$$

In other words, the *optimal sampling size* should account not only for the probability to generate samples within a given Γ_s, but also on the variance associated to the same interval. More samples should be therefore located in areas characterized by a bigger variance.

1.7.2 Importance Sampling

Importance sampling is used when the probability density function f_y of the chosen r.v. y is not very well suited to represent the function g whose statistics are sought. This is the case, for example, when f_y and g have little overlap or when the product $g(y)f_y(y)$ is small. Samples associated to a pdf f_y might also not be able to capture important *features* of the function g. The main idea in importance sampling is to sample according to a modified pdf, i.e., $\tilde{f}_y(y)$ and to define a *preconditioner* as follows:

$$P(y) = \frac{\tilde{f}_y(y)}{f_y(y)}, \tag{1.25}$$

such that:

$$\tilde{\mathbb{E}}\left\{\frac{g(y)}{P(y)}\right\} = \int_\Omega g(y)\frac{f_y(y)}{\tilde{f}_y(y)}\tilde{f}_y(y)\,dy = \int_\Omega g(y)f_y(y)\,dy = \mathbb{E}\{g(y)\} = I \tag{1.26}$$

We can now generate an unbiased Importance Sampling estimator, I_M^p, using i.i.d. r.v.s $\tilde{y}_j$, $j = 1,\ldots,M$ defined over a different probability space $(\tilde{\Omega},\tilde{\mathcal{F}},\tilde{\mathbf{P}})$ endowed with the modified pdf $\tilde{f}_y(y)$. We have that

$$I_M^p = \frac{1}{M}\sum_{j=1}^{M}\frac{g(\tilde{y}_j)}{P(\tilde{y}_j)}. \tag{1.27}$$

The variance of the new estimator is:

$$\tilde{\sigma}^2\{I_M^p\} = \tilde{\mathbb{E}}\left\{\left(\frac{g(y)}{P(y)}\right)^2\right\} - \tilde{\mathbb{E}}^2\left\{\frac{g(y)}{P(y)}\right\} = \int_{\tilde{\Omega}} g^2(y)\frac{f_y^2(y)}{\tilde{f}_y(y)}\,dy - I^2. \tag{1.28}$$

Our purpose is to find the best possible measure $\tilde{f}_y$ such to minimize $\tilde{\sigma}^2\{I_n^p\}$. Assuming g positive, the best possible choice would be as follows:

$$\tilde{f}_y = \frac{g(y)f_y(y)}{\mathbb{E}\{g(y)\}}. \tag{1.29}$$

It is easy to see that, for this case, $\tilde{\sigma}^2\{I_n^p\} = 0$. In practice, equation (1.29) cannot be used as it is, due to the fact that the quantity $\mathbb{E}\{g(y)\}$ is not known in advance and actually is the very quantity we are trying to estimate. A simple alternative uses approximations of $\mathbb{E}\{g(y)\}$ based on the trapezoidal rule.

$$\tilde{f}_y = \frac{\sum_j g(y_j^*)f_y(y_j^*)\mathbb{I}_{(x_{j-1},x_j]}}{\sum_j g(y_j^*)f_y(y_j^*)(x_j - x_{j-1})}, \tag{1.30}$$

where $y_j^* \in (x_{j-1},x_j)$ and $\mathbb{I}_A$ is the indicator function for set A.

The ideal case expressed in (1.29) suggests that, when we have $|gf_y - I\tilde{f}_y| = 0$, i.e., $\tilde{f}_y$ proportional to gf_y, this results in the best possible choice for $\tilde{f}_y$. It can be noticed that this is not always true and, in practice, attention should be paid to the tail of the selected $\tilde{f}_y$ to have a reduction in variance respect to the Monte Carlo approach. If we set $r = gf_y - \mathbb{E}\{g\}\tilde{f}_y = gf_y - I\tilde{f}_y$, we have:

$$\tilde{\sigma}^2\{I_M^p\} = \int_{\tilde{\Omega}}\frac{(gf_y)^2}{\tilde{f}_y}\,d\tilde{y} - I^2 = \int_{\tilde{\Omega}}\frac{(r + I\tilde{f}_y)^2}{\tilde{f}_y}\,d\tilde{y} - I^2 = \int_{\tilde{\Omega}}\frac{r^2}{\tilde{f}_y}\,d\tilde{y} \tag{1.31}$$

If $\tilde{f}_y \ll r^2$ in some not degenerated subset of the domain, we could have a very big increase in $\tilde{\sigma}^2\{I_n^p\}$.

A remedy is to use a blend of the new density $\tilde{f}_y$ and some heavy tailed distribution as a margin against the explosion of $r^2/\tilde{f}_y$. We can, for example, have that:

$$\tilde{f}_y^{\alpha} = \alpha + (1-\alpha)\tilde{f}_y, \tag{1.32}$$

where α is the weight assigned to the uniform distribution.

So far, we have considered a modified measure $\tilde{f}_y$ which is constant throughout the simulation. We might point out that, by progressively sampling the unknown function g, we know more information about it as we proceed with the iterations. Therefore, it makes sense to slightly change our approach by refining our estimation of $\tilde{f}_y$ as part of the Monte Carlo iteration process. This defines an *adaptive* importance sampling methodology where a new measure is calculated as part of the estimation method itself.

Chapter 2

Spectral expansion methods for uncertainty propagation

2.1 From sampling based estimation to spectral expansion

In the previous chapter, we have shown how the variance of Monte Carlo-based estimators is related to the number of samples. In practice, while the speed of convergence for the Monte Carlo approach is proportional to $\sqrt{M}$ with M the total number of samples, stratified sampling with only 1 sample in every stratum (i.e., the so called Latin Hypercube Sampling) results in a convergence rate proportional to M.

The question in this case is: can we do better than this? To answer this question affirmatively, we start from a theorem formulated by Cameron and Martin in 1947 [20] which is based on Wiener's Homogeneous Chaos. Consider a complete probability space $(\Omega, \mathcal{F}, \mathbf{P})$ and denote by C the space of real functionals $g(y)$ of the r.v. y associated with the distribution function f_y which are continuous and have *finite variance*, i.e., $g(y) \in L_2(\Omega, \mathcal{F}, \mathbf{P})$ or,

$$\int_{\Omega} g^2(y)\, f_y\, dy < \infty \tag{2.1}$$

The following theorem can be stated:

Theorem 4 (Cameron and Martin)**.** The Fourier-Hermite series of any (real or complex) functional $g(\mathbf{y}) \in L_2(C)$ converges in the $L_2(C)$ sense to $g(\mathbf{y})$.

We can therefore use a Fourier-Hermite expansion with an infinite number of terms to describe every function $g(\mathbf{y})$ of random variables with finite variance as follows:

$$g(\mathbf{y}) = \sum_{i=1}^{\infty} a_i \Psi_i(\mathbf{y}) \tag{2.2}$$

If, for the sake of simplicity, we focus on the one-dimensional case where $\Psi_i(y) : \mathbb{R} \rightarrow \mathbb{R}$, the Hermite polynomial of order $i+1$ can be determined using the following recursive relation:

$$\Psi_{i+1}(y) = y\Psi_i(y) - i\Psi_{i-1}(y) \tag{2.3}$$

We should now consider the orthogonality properties of the Hermite polynomials respect to the inner product structure associated with the selected probability space. In practice, the product between two Hermite polynomials in a space $L_2(\Omega, \mathcal{F}, \mathbf{P})$ is

$$\langle \Psi_i(y), \Psi_j(y) \rangle_{L_2(\Omega,\mathcal{F},\mathbf{P})} = \int_\Omega \Psi_i(y)\Psi_j(y)\, f_y\, dy \tag{2.4}$$

The latter relationship assumes a very convenient form if the r.v. y is associated to the probability distribution function $f_y(y) = \frac{1}{\sqrt{2\pi}} e^{-y^2/2}$, i.e., the standard Gaussian distribution. In the latter case we have:

$$\int_\Omega \Psi_i(y)\Psi_j(y)\, f_y\, dy = \delta_{i,j} \tag{2.5}$$

as it is well know that the Hermite polynomials are orthogonal respect to the standard Gaussian probability measure.

An alternative to the Monte Carlo strategy for evaluating the expectation of $g(y)$ consists in expanding g using Hermite polynomials as follows:

$$g(y) \approx \sum_{i=1}^{P} a_i\, \Psi_i(y) \tag{2.6}$$

And then computing its expectation as:

$$\mathbb{E}\{g(y)\} = \int_\Omega \left(\sum_{i=1}^{P} a_i\, \Psi_i(y) \right) f_y\, dy = a_1. \tag{2.7}$$

The above expression is obtained as $\langle \Psi_i(y), 1 \rangle = 0$ for $i = 2, 3, \ldots$ but clearly $\langle \Psi_1(y), 1 \rangle = 1$ as $\Psi_1(y) = 1$. Moreover, higher moments of the function $g(y)$ can be computed with similar formulae from coefficients, i.e.,

$$\mathbb{E}\{g^2(y)\} = \sum_{i=1}^{P} a_i^2. \tag{2.8}$$

The expectation of the function $g(y)$ can be found by computing the coefficients of the Hermite expansion (2.6). Such coefficients are evaluated through simple integrals, using the orthogonality properties of the Hermite polynomials. First, we write the residual of (2.6) as follows:

$$R(y) = g(y) - \sum_{i=1}^{P} a_i\, \Psi_i(y) \tag{2.9}$$

and then we impose the orthogonality of the above residual respect to all the Hermite polynomials. As the latter family is a basis of $L_2(\Omega, \mathcal{F}, \mathbf{P})$ it follows that $\langle R(y), \Psi_j(y) \rangle = 0$ for $j = 1, 2, \ldots$ implies $R(y) = 0$. This leads to

$$\begin{aligned} \langle R(y), \Psi_j(y) \rangle &= \langle g(y) - \sum_{i=1}^{P} a_i\, \Psi_i(y), \Psi_j(y) \rangle = \langle g(y), \Psi_j(y) \rangle - \sum_{i=1}^{P} a_i \langle\, \Psi_i(y), \Psi_j(y) \rangle \\ &= \langle g(y), \Psi_j(y) \rangle - a_i\, \|\Psi_i(y)\|^2 = \langle g(y), \Psi_j(y) \rangle - a_i = 0. \end{aligned} \tag{2.10}$$

Therefore:

$$a_i = \langle g(y), \Psi_j(y) \rangle = \int_\Omega g(y)\Psi_j(y)\, f_y\, dy. \tag{2.11}$$

The evaluation of the Fourier-like coefficients amounts to computing integrals. A number of numerical integration formulae can be found in literature that can be used for this purpose. The choice of the best integration formula to use, typically results from a compromise between accuracy and prior knowledge of the function whose expectation is sought. Two popular choices in this regard are the Gauss and the Clenshaw-Curtis integration formulae. The first is usually preferred when evaluating expectations of smooth functions due to the fact that n_g Gauss points give exact quadrature of a polynomial with order up to $2n_g - 1$. On the other hand, the uniform convergence properties of the Chebyshev approximants allow comparable performance for the Clenshaw-Curtis quadrature for cases of non polynomial integrand. Moreover, the Clenshaw-Curtis formula can be implemented effortlessly by the FFT algorithm. A detailed discussion about the difference of the above quadrature rules is presented in [87].

The computation of $\mathbb{E}\{g(y)\}$ becomes:

$$\mathbb{E}\{g(y)\} = \int_{\Omega} g(y)\, f_y(y)\, dy \approx \sum_{i=1}^{n_g} g(y^{(i)})\, f_y(y^{(i)})\, w^i \tag{2.12}$$

where the *deterministic* samples $y^{(i)}$ are located at the n_g integration locations of the chosen formula and w^i are the associated weights.

2.2 Multidimensional case

We now consider the generalization of Hermite expansion or Gauss-Hermite integration of a function of several random variables, i.e., the vector $\mathbf{y} \in \Omega$, where $\mathbf{y} = [y_1, y_2, \ldots, y_d]$ and $\Omega = [-\infty, +\infty]^d$. The expectation of $g(\mathbf{y}) : \mathbb{R}^d \to \mathbb{R}$ now becomes:

$$\mathbb{E}\{g(\mathbf{y})\} = \int_{\Omega} g(\mathbf{y}) f_{y_1,y_2,\ldots,y_d}\, dy. \tag{2.13}$$

where $f_{y_1,y_2,\ldots,y_d}$ is the joint probability distribution function of the r.v.s $\{y_1, y_2, \ldots, y_d\}$. A straightforward generalization of the developments discussed above for the one-dimensional case results if we assume that $\mathbf{y}$ consists of a set of i.i.d. (independent, identically distributed) random variables. In this case the joint probability distribution can be written as:

$$f_{y_1,y_2,\ldots,y_d} = f_{y_1} f_{y_2} \ldots f_{y_d} \tag{2.14}$$

A tensor product of Hermite polynomials is used for multivariate spectral expansion. We give a simple example in the two-dimensional case, where we want to expand the function $g(y_1, y_2)$ using a maximum polynomial degree equal to 2 (linear approximation):

$$\begin{aligned} g(y_1, y_2) \approx\, & a_1\, (\Psi_1(y_1)\Psi_1(y_2)) + a_2\, (\Psi_1(y_1)\Psi_2(y_2)) + \\ & a_3\, (\Psi_2(y_1)\Psi_1(y_2)) + a_4\, (\Psi_2(y_1)\Psi_2(y_2)). \end{aligned} \tag{2.15}$$

Note that, by using a full tensorization, the maximum approximation order is now 4 instead of the second order specified for each y_i, $i = 1, \ldots, d$. In general, if m_i, $i = 1, 2, \ldots, d$ is the polynomial order in each dimension, the total number of terms in a full tensor expansion is $m_1\, m_2\, \ldots\, m_d$. Instead of a full tensorization, a reduced expansion can be performed such that the maximum polynomial order is preserved. In this case, and assuming the same polynomial order m for all dimensions, the number of terms in the expansion is:

$$g(\mathbf{y}) \approx \sum_{i=1}^{P} a_i\, \Psi_i(\mathbf{y}) \quad \text{where} \quad P = \frac{(d+m)!}{d!m!}. \tag{2.16}$$

The construction above allows to extend to more than one dimension the orthogonality properties of the Hermite system.

By looking at (2.16) it can be noticed the fast grow rate in the number of terms in the spectral expansion for an increasing number of input random variables. This effect is known as the *curse of dimensionality*, meaning that the number of terms needed for expanding a given functional is subject to a dramatic increase with the number of input random variables.

The numerical integration rules used in one-dimension can be extended to the multivariate case. In practice, the expectation of $g(\mathbf{y}) : \mathbb{R}^d \to \mathbb{R}$ becomes:

$$\begin{aligned} \mathbb{E}\{g(\mathbf{y})\} &= \int_\Omega g(\mathbf{y})\, f_{y_1,y_2,\ldots,y_d}(\mathbf{y})\, dy \\ &\approx \sum_{i=1}^{n_g} g(y_1^{(i)}, y_2^{(i)}, \ldots, y_d^{(i)})\, f_{y_1,y_2,\ldots,y_d}(y_1^{(i)}, y_2^{(i)}, \ldots, y_d^{(i)})\, w_{y_1}^i w_{y_2}^i \ldots w_{y_d}^i \end{aligned} \tag{2.17}$$

where $\mathbf{y}^{(i)} = (y_1^{(i)}, y_2^{(i)}, \ldots, y_d^{(i)})$, $i = 1, \ldots, n_g$ is the set of quadrature points and $w_{y_j}^i$ is the weight associated to the i-th integration location for variable y_j. It is clear at this point that the computational cost of evaluating the expectation of g is primarily related to evaluating the functional for the ensemble of multivariate quadrature points, i.e., evaluate $g(\mathbf{y}^{(i)}) = g(y_1^{(i)}, y_2^{(i)}, \ldots, y_d^{(i)})$. If full tensorization is used for quadrature locations $\mathbf{y}^{(i)}$, their number will also grow exponentially with the number of input dimensions. Two techniques are used to keep as limited as possible the computational cost of evaluating g for the multivariate case, i.e., using a nested quadrature rule and trying to mitigate the curse of dimensionality using sparse grids.

2.3 Nested univariate quadrature and multivariate Sparse Grids

Nested quadrature is designed to maximize sampling re-use. Assume an integration grid $I_l = \{\mathbf{y}^{(1)}, \mathbf{y}^{(2)}, \ldots, \mathbf{y}^{(n_l)}\}$ is built for the selected approximation order l. We would like to build a series of nested quadrature integration grids such that $I_{l-1} \subset I_l$ for all $l \geqslant 1$. This means that we could increase the accuracy of the expectation of g by using *all* the functional evaluations computed so far. This property relates on the one-dimensional quadrature rule adopted; for example, Newton-Cotes formulae have this property, as shown in Figure 2.1a as a result of their dichotomic subdivision strategy.

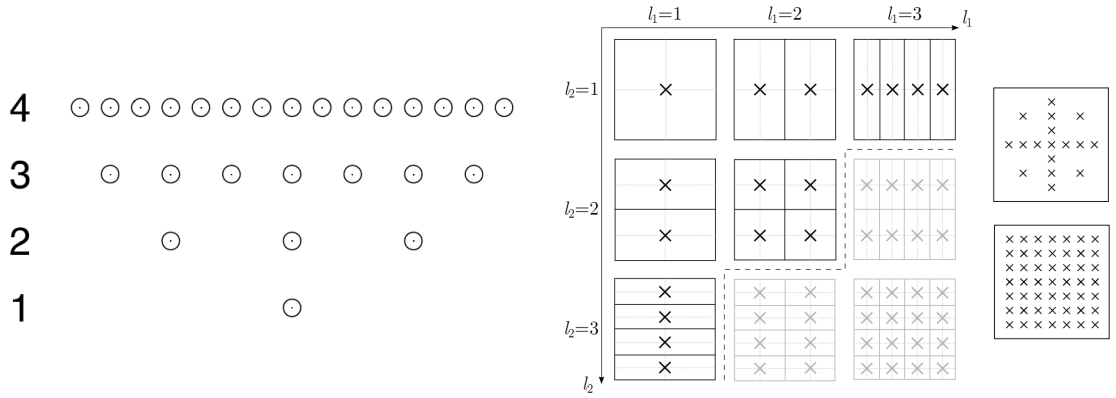
Clenshaw-Curtis quadrature points located at the zeros of Chebyshev polynomials can also be used to generate nested quadrature formulae. The location of the latter points is:

$$y^{(i)} = \cos\frac{i\,\pi}{l} \quad \text{where} \quad 0 \leqslant i \leqslant l, \tag{2.18}$$

If we introduce a new parameter s for the integration order, the above formula is modified as follows:

$$y^{(i)} = \cos\frac{i\,\pi}{2^s} \quad \text{where} \quad 0 \leqslant i \leqslant 2^s \quad \text{and} \quad s = 1, 2, 3, \ldots, \tag{2.19}$$

with the special case $y^{(i)} = 0$ for $s = 0$. It can be seen that we have $I_{s-1} \subset I_s$ is this case and the above formula is nested. Note that the number of sampling locations practically doubles going from s to $s + 1$.



(a) (b)

Figure 2.1: Nested quadrature formula for Newton-Cotes integration (a) and two-dimensional demonstration of partial tensorization of one-dimensional quadrature grids (b).

An other very useful idea is to exploit partial tensorization o one-dimensional integration formulae similarly to what discussed above for multivariate Hermite polynomials. In other words, the order of accuracy for full-tensor multivariate quadrature is not consistent to that of the one-dimensional integration formulae. As a starting point, we introduce the following notation in the context of one-dimensional integration:

$$\int_\Omega h(y) f_y \, dy = \int_\Omega g(y) \, dy \approx Q_l \, g = \sum_{i=1}^{n_l} g(y_l^{(i)}) \, w_l^i. \tag{2.20}$$

where, in this case, we use $g(y) = h(y) f_y$. If we also define $Q_0 \, g = 0$, then we can introduce the following *difference* formula:

$$\Delta_l \, g = (Q_l - Q_{l-1}) g \quad \text{where} \quad l = 1, 2, \ldots. \tag{2.21}$$

A d-dimensional difference formula can be derived by tensorization of one-dimensional formulae, by introducing the vector $\mathbf{l} = \{l_1, l_2, \ldots, l_d\}$, as follows:

$$\Delta_{\mathbf{l}} \, g = (\Delta_{l_1} \otimes \Delta_{l_2} \otimes \cdots \otimes \Delta_{l_d}) g \tag{2.22}$$

The approach suggested by Smolyak [82] for integration of g is therefore:

$$\int_\Omega h(\mathbf{y}) f_{\mathbf{y}} \, d\mathbf{y} = \int_\Omega g(\mathbf{y}) \, d\mathbf{y} \approx Q_l^d \, g = \sum_{|\mathbf{k}| \leqslant l+d-1} \Delta_{\mathbf{k}} \, g, \tag{2.23}$$

where $\mathbf{k} = \{k_1, k_2, \ldots, k_d\}$ and we define $|\mathbf{k}| = \sum_{i=1}^d k_i$. Note the the factor $l + d - 1$ depends linearly on d, the number of dimension over which the summation is evaluated. An example is provided in Figure 2.1b showing a comparison between full and partial tensor grids obtained by non-nested one-dimensional Newton-Cotes quadrature.

The Smolyak approach is characterized by the expression $|\mathbf{k}| \leqslant l + d - 1$ which is valid for an homogeneous approximation order over the various dimensions. A graphical representation of the Smolyak two-dimensional difference formulae and associated sparse grid can be found in Figure 2.2a and 2.2b. Figure 2.2c and 2.2d show a possible modification of the previous multivariate quadrature strategy which accounts for the possibility that a higher order is needed along one of the considered dimensions. In this case, the above constraint needs to be modified as follows:

$$|\mathbf{k}| \leqslant l(\mathbf{k}) + d - 1 \tag{2.24}$$

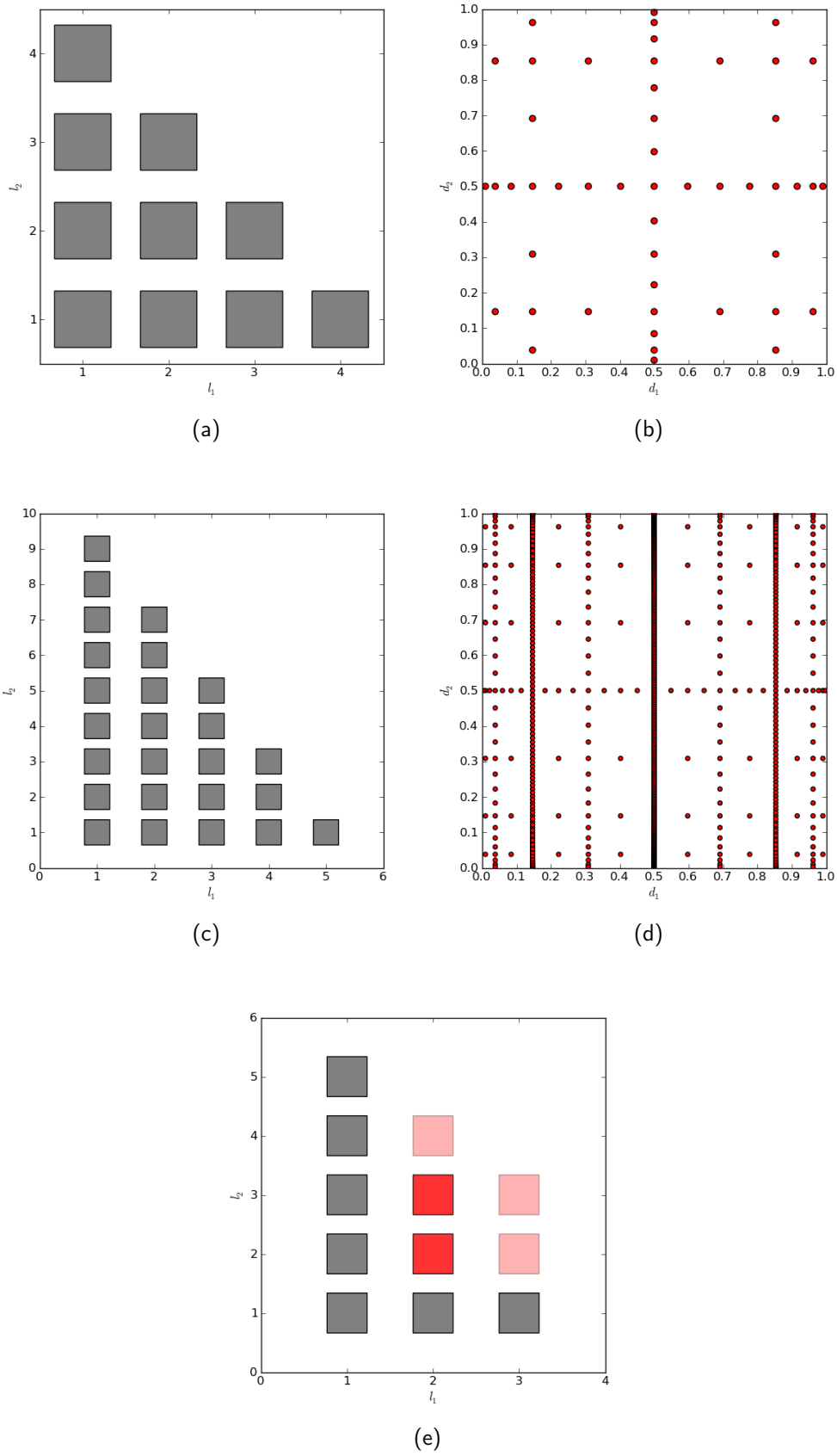


Figure 2.2: Graphical representation of the single two-dimensional quadrature formulae used for the Smolyak formula of order 4 (a) and associated quadrature points (b). Multivariate anisotropic sparse quadrature formulae (c) and associated grid (d). Concept of adaptive sparse grid (e).

Case	Probability Distribution	Askey Chaos	Support
Continuous	Gaussian	Hermite Chaos	(-∞,+∞)
	Gamma	Laguerre Chaos	[0,∞)
	Beta	Jacobi Chaos	[a, b]
	Uniform	Legendre Chaos	[a, b]
Discrete	Poisson	Charlier Chaos	{0, 1, 2,. . . }
	Binomial	Krawtchouk Chaos	{0, 1, 2,. . . ,N}
	Negative binomial	Meixner Chaos	{0, 1, 2,. . . }
	Hypergeometric	Hahn Chaos	{0, 1, 2,. . . ,N}

Table 2.1: Probabilistic measures associated to orthogonal polynomials within the Wiener-Askey Scheme.

and $l(\mathbf{k})$ is a multi-linear function in $\mathbf{k}$.

For cases where nested one-dimensional quadrature rules are used, it is also possible to generalize the above concepts within an adaptive framework where multivariate difference grids (in practice, they provide the building block of our quadrature formula) can be progressively evaluated based on the function evaluations already computed. A sketch is provided in Figure 2.2e showing this concept iteratively, where red difference formulae are evaluated at the current iteration, gray difference formulae have been previously evaluated and the next one to come are highlighted using a transparent effect.

2.4 Generalized polynomial chaos

Consider the case where both $g(\mathbf{y})$ and the i.i.d. r.v.s $\mathbf{y}$ have Gaussian distribution. Note that this is true if the function g is a linear map $g : \mathbb{R}^d \to \mathbb{R}$. In the latter case the statistics evaluated using a polynomial chaos expansion converge *exponentially* fast.

This is not true in general, meaning that the convergence rate to expectations of g with expansions in terms of Hermite polynomials, is largely dependent on g itself. However, it is important to highlight the following important similarity. Consider a family of polynomials $\Psi_i(x)$, $i = 1, 2, \dots$ with $x \in \mathbb{R}$. The usual product in $L_2(\mathbb{R})$ between two polynomials of the same family is extended by including a function $w(x) : \mathbb{R} \to \mathbb{R}$, as follows:

$$\langle \Psi_i(x)\, \Psi_j(x) \rangle_w = \int_{\mathbb{R}} \Psi_i(x)\, \Psi_j(x)\, w(x)\, dx \tag{2.25}$$

A family of orthogonal polynomials respect to the *measure* $w(x)$ is defined as:

$$\langle \Psi_i(x)\, \Psi_j(x) \rangle_w = 0 \quad \forall i \neq j \tag{2.26}$$

Many families of polynomials have been developed in literature which are orthogonal to known $w(x)$. Moreover, for many of the latter cases the weight functions w are formally identical to probability measures used in applications. Table 2.1 summarizes polynomial families of hypergeometric or basic hypergeometric type which can be organized into a hierarchy using the Askey scheme [10].

This leads to an exponential convergence of the error, when evaluating expectations using polynomials othogonal to the probability distribution of the quantities of interest. In practice, a precise statistical quantification of g is not known a priori as it is the result of the estimation process. Nevertheless, for i.i.d. input random variables whose probability density function can be assimilated to the ones shown in Table 2.1 using an appropriate family of orthogonal polynomials gives a starting point that leads to an exponential convergence of the error under linearity of g.

2.5 Spectral coefficient extraction as an algebraic problem

In the previous Sections, using the orthogonality of the selected family of polynomials, we used numerical integration to extract spectral expansion coefficients. In the present Section, we formulate the problem of finding the expansion coefficients as an algebraic problem and we discuss possible solution strategies. We start by suggesting a truncated expansion for the functional $g(\mathbf{y})$, $\mathbf{y} \in \mathbb{R}^d$

$$g(\mathbf{y}) \approx \sum_{i=1}^{P} \alpha_i \, \Psi_i(\mathbf{y}) \tag{2.27}$$

note that in this case the multivariate functions $\Psi_i(\mathbf{y})$ are tensor product of orthogonal polynomials. Assume g is sampled at M locations $\mathbf{y}^{(i)} = \{y_1^{(i)}, y_2^{(i)}, \ldots, y_d^{(i)}\}$, $i = 1, \ldots, M$. Expression (2.27) can be written for every sample location, resulting in the following linear system of equations:

$$\mathbf{g} = \mathbf{\Psi}\, \boldsymbol{\alpha} \tag{2.28}$$

where $\mathbf{g} = \left[g(\mathbf{y}^{(1)}), g(\mathbf{y}^{(2)}), \ldots, g(\mathbf{y}^{(M)})\right]^T$ is the sample vector, $\Psi = \left[\Psi_1, \Psi_2, \ldots, \Psi_P\right]$ is a matrix whose columns are the member of the orthogonal polynomial family evaluated at the sampling points, and $\boldsymbol{\alpha} = \left[\alpha_1, \alpha_2, \ldots, \alpha_P\right]$.

The system matrix $\mathbf{\Psi}$ is in general not symmetric, and various techniques can be used to solve for the coefficients, mainly related to the total number of samples M and the selected basis cardinality P. For example, when $M \gg P$, least squares techniques can be used to solve 2.28. For cases where $M = P$ and the matrix $\mathbf{\Psi}$ is reasonably well conditioned, direct and iterative unsymmetric system solvers can be used instead. Finally, we stress the fact that for our case a major computational effort could be devoted to evaluate the right hand side $\mathbf{g}$. Assume we are computing expectations of a quantity resulting from a complex system of non-linear differential equations. In this case, every sample evaluation would require the solution of a time consuming numerical simulation. This suggests that, in general, we would prefer to solve the case $M \ll P$ as computing the basis functions at the samples locations is significantly less expensive than computing $\mathbf{g}(\mathbf{y}^{(i)})$, $i = 1, \ldots, M$. A way to solve system (2.28) for the underdetermined case using techniques proper of the Compressive Sampling paradigm, will be explained in the following Chapters.

2.6 Application to Stochastic PDEs

In the previous Sections spectral expansion techniques are employed to compute expectations of functionals. This Section shows how to extend the latter methodology to compute statistics of solutions for partial differential equations.

Consider a probability space $(\Omega, \mathcal{F}, \mathbf{P})$ in which Ω is the set of elementary event, $\mathcal{F}$ is the σ-algebra of events, and $\mathbf{P}$ defines a probability measure on $\mathcal{F}$. A vector of independent and identically distributed random variables with joint probability density function $\rho(\mathbf{y}) : \mathbb{R}^d \to \mathbb{R}_{\geqslant 0}$ is indicated by $\mathbf{y} = (y_1, \ldots, y_d)$ with $y_i : \Omega \to \mathbb{R}$, $i = 1, \ldots, d$, and $d \in \mathbb{N}$.

We define a spatial domain $\mathcal{D} \subset \mathbb{R}^n$ with boundary $\partial\mathcal{D}$ and time $t \in [0, T]$.

We state our problem as follows: find a solution $\mathbf{u}(\mathbf{x}, t, \mathbf{y}) : \mathcal{D} \times [0, T] \times \Omega \to \mathbb{R}^q$, $q \in \mathbb{N}$, such that

$$\begin{aligned} \mathcal{L}(\mathbf{x}, t, \mathbf{y}, \mathbf{u}) &= \mathbf{f}(\mathbf{x}, t, \mathbf{y}) \quad \text{on} \quad \mathcal{D}, \\ \mathcal{B}(\mathbf{x}, t, \mathbf{y}, \mathbf{u}) &= \mathbf{u}_0(\mathbf{x}, t, \mathbf{y}) \quad \text{on} \quad \partial\mathcal{D} \end{aligned} \tag{2.29}$$

hold $\mathbf{P}$-a.s. in Ω. Here we assume the well-posedness (in $\mathbf{P}$-a.s. sense) of (2.29) with respect to the choices of the forcing and boundary functions $\mathbf{f}$ and $\mathbf{u}_0$, respectively.

We want to build a multiresolution representation of $\mathbf{u}(\mathbf{x}, t, \mathbf{y})$ at a fixed location in space $\mathbf{x}_a \in \mathcal{D}$ and time $t_a \in [0, T]$ by using samples $\{\mathbf{u}(\mathbf{x}_a, t_a, \mathbf{y}^{(k)}) : k = 1, \ldots, M\}$ corresponding to M realizations $\{\mathbf{y}^{(k)} : k = 1, \ldots, M\}$ of the random inputs $\mathbf{y}$. To simplify the notation and presentation, we henceforth drop the space and time variables $\mathbf{x}_a$, t_a and describe our approach for a scalar, multivariate solution $u(\mathbf{y})$, i.e., with $q = 1$ and arbitrary d.

Note that, in the context of a discretized solution of system (2.29), the set of locations (grid points) in space and time where $\mathbf{u}$ is available extends the concept of *sampling* to the variables $\mathbf{x}$ and t. Therefore, within the proposed non intrusive approach, space and time can be treated as further random variables associated to uniform probability measures where the pre-determined sampling set depends on the adopted numerical discretization.

Under the above assumptions, system 2.29 becomes:

$$\begin{aligned} \mathcal{L}(\mathbf{y}, u) &= \mathbf{f}(\mathbf{y}) \quad \text{on} \quad \mathcal{D}, \\ \mathcal{B}(\mathbf{y}, u) &= u_0(\mathbf{y}) \quad \text{on} \quad \partial\mathcal{D} \end{aligned} \tag{2.30}$$

The solution $\mathbf{u}$ is expanded as follows:

$$u(\mathbf{y}) = \sum_{i=1}^{P} a_i \, \Psi_i(\mathbf{y}) \tag{2.31}$$

The coefficients a_i are determined by projecting the residual on span$\{\Phi_i,\, i = 1, \ldots, P\}$. If we set aside the treatment of boundary conditions, we can formulate P equations, where the j-th reads:

$$\mathbb{E}\left\{ \left[\mathcal{L}\left(\mathbf{y}, \sum_{i=1}^{P} a_i \, \Psi_i(\mathbf{y}) \right) - \mathbf{f}(\mathbf{y}) \right] \Phi_j(\mathbf{y}) \right\} = 0 \quad \forall j = 1, \ldots, P. \tag{2.32}$$

The solution strategy greatly depends on the selected family Φ_i, $i = 1, \ldots, P$. Depending on this choice, two possible strategies, *intrusive* and *non intrusive*, are derived to solve stochastic partial differential equations. These two methodologies are described with more detail in the following Sections.

2.7 Intrusive Approach

A possible strategy to solve equation (2.32), which extends a well known approach for deterministic PDEs, uses $\Phi_i = \Psi_i$, $i = 1, \ldots, P$. This approaches is know as *Galerkin projection* and gives rise to the family of *intrusive methodologies*. To better understand how intrusive methodologies work, we focus on the 1D scalar transport equation with stochastic weave speed.

2.7.1 1D Scalar Trasport with stochastic wave speed

Assume $\mathcal{D} \equiv [-1, 1]$ is a one-dimensional interval and the time parameter is defined as $t \in [0, +\infty)$. Consider a r.v. $y \in (\Omega, \mathcal{F}, \mathbf{P})$ defined over a complete probability space, and associated with the

probability density function $\rho(y)$. We are interested in the solution of the following hyperbolic equation:

$$\frac{\partial u(x,t,y)}{\partial t} = c(y)\frac{\partial u(x,t,y)}{\partial x} \tag{2.33}$$

with initial conditions:

$$u(x,0,t) = u_0(x,y) \tag{2.34}$$

Due to the hyperbolic nature of equation (2.33), boundary conditions depend on the sign of the scalar wave speed:

$$\begin{cases} u(-1,t,y) = u_L(t,y) \quad \text{for} \quad c > 0 \\ u(1,t,y) = u_R(t,y) \quad \text{for} \quad c < 0 \end{cases} \tag{2.35}$$

2.7.2 Numerical solution

The convection operator is discretized using a 1D first order upwind finite volume approximation in space. The following approximation can be written with reference to a single finite volume, see Figure 2.3:

$$c\frac{\partial u}{\partial x} \approx \frac{(cu)_e - (cu)_w}{\Delta x} \tag{2.36}$$

If we insert $F = \frac{c}{\Delta x}$, the previous equation becomes:

$$c\frac{\partial u}{\partial x} \approx F_e u_e - F_w u_w \tag{2.37}$$

For $c > 0$, the convected variables at e, w are:

$$\begin{cases} u_e = u_P \\ u_w = u_W, \end{cases} \tag{2.38}$$

while for $c < 0$:

$$\begin{cases} u_e = u_E \\ u_w = u_P \end{cases} \tag{2.39}$$

Vector $\mathbf{u} = \{u_1, u_2, \ldots, u_n\}$ stores the main problem unknowns, located at the center of the n cells. The discretized one-dimensional first order upwind operator is:

$$c\frac{\partial u}{\partial x} \approx \mathbf{C}\mathbf{u} \tag{2.40}$$

The matrix $\mathbf{C}$ is, in general, an unsymmetric tri-diagonal matrix whose i-th row stores the following components:

$$\begin{cases} C_{i,i-1} = c_w = -max(F_w, 0) \\ C_{i,i} = c_e + c_w + (F_e - F_w) \\ C_{i,i+1} = c_e = -max(0, -F_e) \end{cases} \tag{2.41}$$

Please note that the above discretized operator C is non linear:

$$C(-c) \neq -C(c) \tag{2.42}$$

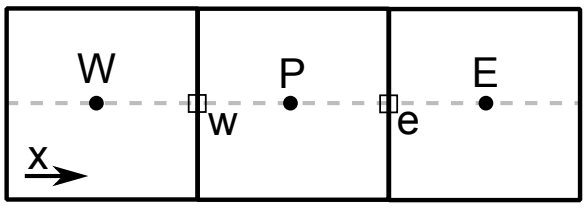


Figure 2.3: Subset of one-dimensional finite volume mesh.

2.7.3 Spectral expansion

Both the main solution $u(x,t,y)$ and the random wave speed $c(y)$ are expanded in probability space. The solution of (2.33) becomes:

$$u(x,t,y) = \sum_{i=0}^{+\infty} u_i(x,t)\psi_i(y) \approx \sum_{i=0}^{N} u_i(x,t)\psi_i(y) \tag{2.43}$$

The same approach can be used for the random wave speed:

$$c(y) = \sum_{i=0}^{+\infty} c_i\psi_i(y) \approx \sum_{i=0}^{N} c_i\psi_i(y) \tag{2.44}$$

Using the expressions above, expectation of c can be evaluated as follows:

$$\begin{aligned} \mathbb{E}\{c(y)\} = \mu_c &= \int_\Omega \left[\sum_{i=0}^{N} c_i\psi_i(y)\right]\rho(y)\,dy \\ &= \sum_{i=0}^{N} c_i\langle\psi_0,\psi_i(y)\rangle = \sum_{i=0}^{N} c_i\delta_{0,i} = c_0 \quad \text{since} \quad \psi_0 = 1 \end{aligned} \tag{2.45}$$

and similarly:

$$\sigma^2\{c(y)\} = \int_\Omega \left[\sum_{i=0}^{N} c_i\psi_i(y) - \mu_c\right]^2 \rho(y)\,dy = \sum_{i=0}^{N} c_i^2. \tag{2.46}$$

2.7.4 Stochastic Galerkin approach for the 1D trasport equation

First of all, we replace the correct solution u and the random wave speed c with their truncated approximations in probability space, $\hat{u}$ and $\hat{c}$, respectively. The truncation operation results in a non zero residual:

$$R(x,t,y) = \frac{\partial\hat{u}}{\partial t} - \hat{c}\frac{\partial\hat{u}}{\partial x} \tag{2.47}$$

A discrete upwind convection operator is applied:

$$\mathbf{R}(x,t,y) = \frac{\partial\hat{\mathbf{u}}}{\partial t} - \mathbf{C}(y)\hat{\mathbf{u}} \tag{2.48}$$

where bold quantities are used to represent vectors storing values of u at cells centers. The residual is made orthogonal to the same basis used to expand the main unknown and wave speed.

$$\begin{aligned}
&\mathbb{E}\{\mathbf{R}\,\psi_j\} = \langle \mathbf{R}, \psi_j \rangle = 0 \\
&\langle \sum_{i=0}^{N} \frac{\partial \mathbf{u_i}}{\partial t}\,\psi_i - \sum_{i=0}^{N} \mathbf{C}(y)\mathbf{u_i}\,\psi_i, \psi_j \rangle = 0 \\
&\sum_{i=0}^{N} \frac{\partial \mathbf{u_i}}{\partial t}\,\langle \psi_i, \psi_j \rangle - \sum_{i=0}^{N} \langle \mathbf{C}(y)\psi_i, \psi_j \rangle \mathbf{u_i} = 0 \\
&\mathbf{A}\,\frac{\partial \mathbf{v}}{\partial t} - \mathbf{D}\,\mathbf{v} = 0
\end{aligned} \tag{2.49}$$

Where $\mathbf{v}$ is a discretization in both space and probability of the unknown u. The matrix $\mathbf{A}$ is defined as:

$$A_{i,j} = \langle \psi_i, \psi_j \rangle, \tag{2.50}$$

while matrix $\mathbf{C}$ is commonly known as the *multiplication tensor*:

$$C_{i,j} = \sum_{k=0}^{N} c_k \langle \psi_k\,\psi_i, \psi_j \rangle \tag{2.51}$$

It can be easily seen that the matrix $\mathbf{C}$ is symmetric.

The scheme is completed by a discretization in time of $\frac{\partial \mathbf{v}}{\partial t}$. For example, a first-order explicit discretization can be accomplished by using the Euler method. Finally, we remark how the intrusive uncertainty quantification developments above require a substantial modification of the deterministic finite volume solver. A non-intrusive way to handle the propagation of uncertainty is presented next.

2.8 Stochastic collocation for non-intrusive UP

Instead of using a Galerkin approach in building a residual formulation, we use a Dirac delta function:

$$\Phi_j(\mathbf{y}) = \delta(y - y^{(j)}) \quad \text{where} \quad j = 1, \ldots, M. \tag{2.52}$$

In practice, this function is such that

$$\int_\Omega f(y)\delta(y - y^{(j)})\,d\Omega = f(y^{(j)}). \tag{2.53}$$

When applied to equation (2.32), we have that:

$$\mathcal{L}\left(\mathbf{y^{(j)}}, \sum_{i=1}^{P} a_i\,\Psi_i(\mathbf{y}^{(j)})\right) = \mathbf{f}(\mathbf{y}^{(j)}) \quad \forall j = 1, \ldots, M, \tag{2.54}$$

meaning that deterministic partial differential equations must hold for all M parameter realizations.

Moreover, note that this approach allows a complete decoupling between space-time discretization and computation of response statistics. Two possible strategies, affecting the locations $\mathbf{y}^{(j)}$, can be used in this context:

- *Cubature grid approaches*. Samples are drawn according to cubature grids, therefore evaluating expectations using numerical integration rules.

- *Regression approaches*. The stochastic response at the sampling points is first fitted using a system of basis functions. At a second stage, expectations are evaluated using the approximate response.

This study focuses on the last approach. An intuitive graphical representation of a regression methodology is schematically depicted in Figure 2.4, where the stochastic response in the whole sample space is approximated from that at the sampling locations.

A few interesting properties result from this non-intrusive approach. Unlike methodologies based on cubature grids, samples can be drawn at random if adaptivity is disregarded. This fact gives more flexibility for example, when existing libraries of responses are available. Moreover, a great advantage of non intrusive methodologies is that they don't require any change to be applied to existing deterministic solvers. This fact increases significantly the applicability of the presented uncertainty quantification framework to real engineering problems. However, it must be noted that accuracy of the deterministic solutions is assumed to be uniform in parameter space. In other words, the solver must be able to handle, *with comparable accuracy* the computation of the solution for all realizations of the parameters.

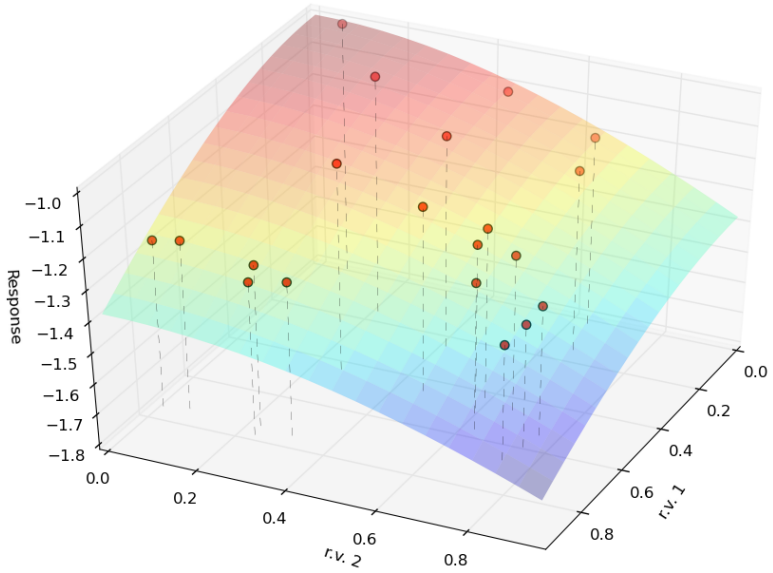


Figure 2.4: Schematic representation of a non-intrusive regression approach.

2.9 Multiresolution and Multiwavelets

Approximation in probability using orthogonal polynomial systems fails in providing sufficient accuracy for non-smooth stochastic responses. One on the challenges of modern uncertainty propagation methodologies is to develop approximation frameworks that work well both in the continuous and discontinuous cases. Our contribution follows this direction, using multiresolution approximations in the context of non-intrusive regression.

2.9.1 Multiresolution Analysis

A *multiresolution* approximation of $\mathbf{L}^2([0,1])$ is expressed by means of a nested sequence of closed subspaces $\mathbf{V}_0 \subset \mathbf{V}_1 \subset \cdots \subset \mathbf{V}_j \subset \cdots \subset \mathbf{L}^2([0,1])$, where each $\mathbf{V}_j = \text{span}\{\phi_{j,k}(y) : k =$

$0, \ldots, 2^j - 1\}$ and

$$\phi_{j,k}(y) = 2^{j/2}\phi(2^j y - k) \tag{2.55}$$

are generated by dilations and translations of a *scaling* function $\phi(y) : [0,1] \to \mathbb{R}$. The scaling function $\phi(y)$ is such that the closure of the union of $\mathbf{V}_j$, i.e., $\overline{\bigcup_{k=1}^{\infty} \mathbf{V}_k}$, is dense in $\mathbf{L}^2([0,1])$. Let the *wavelet* subspace $\mathbf{W}_j$ denote the orthogonal complement of $\mathbf{V}_j$ in $\mathbf{V}_{j+1}$, that is $\mathbf{V}_{j+1} = \mathbf{V}_j \oplus \mathbf{W}_j$ and $\mathbf{V}_j \perp \mathbf{W}_j$. It can be shown that $\mathbf{W}_j = \text{span}\{\varphi_{j,k}(y) : k = 0, \ldots, 2^j - 1\}$ where $\varphi_{j,k}(y)$ is generated from dilation and translation of a *mother* wavelet function $\varphi(y) : [0,1] \to \mathbb{R}$, i.e.,

$$\varphi_{j,k}(y) = 2^{j/2}\varphi(2^j y - k). \tag{2.56}$$

By the construction of the wavelet spaces $\mathbf{W}_j$, it is straightforward to see that $\mathbf{V}_j = \mathbf{V}_0 \oplus (\bigoplus_{k=0}^{j} \mathbf{W}_k)$, and consequently $\mathbf{V}_0 \oplus \left(\bigoplus_{k=0}^{\infty} \mathbf{W}_k\right) = \mathbf{L}^2([0,1])$. Therefore, any function $u(y) \in \mathbf{L}^2([0,1])$ admits an orthogonal decomposition of the form

$$u(y) = \tilde{\alpha}_{0,0}\phi_{0,0}(y) + \sum_{j=0}^{\infty}\sum_{k=0}^{2^j-1} \alpha_{j,k}\varphi_{j,k}(y), \tag{2.57}$$

where $\tilde{\alpha}_{0,0} = \langle u, \phi_{0,0}\rangle_{\mathbf{L}^2([0,1])}$ and $\alpha_{j,k} = \langle u, \varphi_{j,k}\rangle_{\mathbf{L}^2([0,1])}$. To simplify the notation, we rewrite (2.57) in the form

$$u(y) = \sum_{i=1}^{\infty} \alpha_i \psi_i(y), \tag{2.58}$$

in which we establish a one-to-one correspondence between elements of the basis sets $\{\psi_i : i = 0, \ldots, \infty\}$ and $\{\phi_{0,0}, \varphi_{j,k} : k = 0, \ldots, 2^j - 1, j = 0, \ldots, \infty\}$.

2.9.2 Multiwavelet Approximation

In the present study we adopt the slightly more complicated multiresolution of Alpert [8] where multiple scaling functions $\{\phi_i(y) : i = 0, \ldots, m-1\}$ are used to construct $\mathbf{V}_0$. Specifically, we choose $\phi_i(y)$ as the Legendre polynomial of degree i defined on the interval $[0,1]$. An orthonormal basis $\{\varphi_i(y) : i = 0, \ldots, m-1\}$ for $\mathbf{W}_0$ is also established. More precisely, let $\mathbf{U}_m = \{u(y) \in \mathbf{L}^2([0,1]) : \int_{[0,1]} u(y)\, y^m\, dy = 0\}$ represent the subspace of functions in $\mathbf{L}^2([0,1])$ with m vanishing moments. We then construct $\varphi_i \in \mathbf{U}_j$, $j = 0, \ldots, i + m - 1$, with the orthonormality constraint $\langle \varphi_i, \varphi_j\rangle_{\mathbf{L}^2[(0,1)]} = \delta_{ij}$, $i, j = 0, \ldots, m-1$, where δ_{ij} is the Kronecker delta. The *multiwavelet* basis functions $\varphi_{j,k}$ are then generated by dilations and translations of $\{\varphi_i(y) : i = 0, \ldots, m-1\}$.

The resulting basis is unique (up to the sign) and provides a generalization of Legendre and Haar representations. In particular, Legendre polynomials can be obtained by stopping the expansion at the resolution $j = 0$, while Haar wavelets are obtained for $m = 0$. If expanded in the Alpert multiwavelet basis, sparse representations are likely to be observed for piecewise smooth functions. Sharp gradients, bifurcations or discontinuities, for example in hyperbolic problems, motivate the use of such dictionaries as multiwavelet with the ability of capturing these local *features* for which global polynomials may not be adequate, see, e.g., [64]. In addition to several numerical advantages, the orthogonality property of Alpert multiwavelets is also desirable allowing first and second order statistics of u to be evaluated directly from the expansion coefficients. We refer the interested reader to [8] for an in-depth derivation of the Alpert multiwavelet basis.

In an effort to provide a self-contained exposition on the proposed methodology, the derivation of 1D Alpert Multiwavelet basis is reported in the appendix.

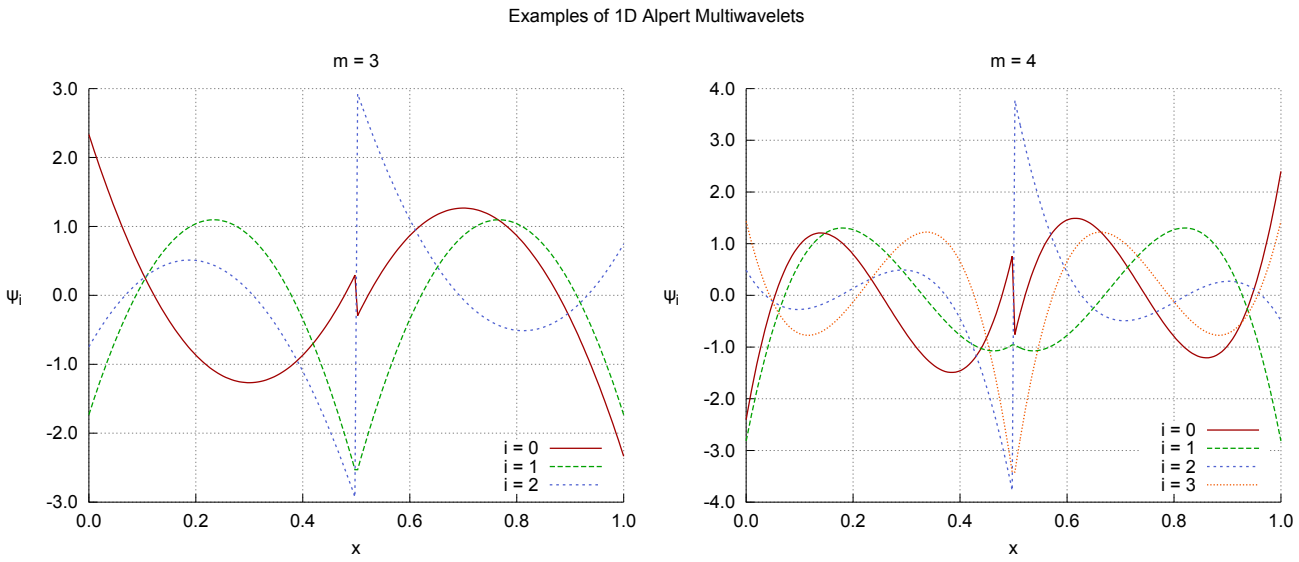


Figure 2.5: Examples of Alpert Multiwavelets with 3 and 4 vanishing moments, respectively.

Construction for arbitrary dimensionality

The construction of multiwavelet bases for $\mathbf{L}^2([0,1]^d)$ is presented in two successive stages. Some intuition is developed first for the two dimensional case, followed by the construction for arbitrary d.

A modified notation is introduced where $\mathbf{V}_0^m \subset \mathbf{V}_0$ is the one-dimensional subspace generated by $\{\phi_i(y) : i = 0, \ldots, m-1\}$. For multiple dimensions, the vector $\boldsymbol{m} = \{m_1, \ldots, m_d\}$ is introduced such that $\mathbf{V}_0^{\boldsymbol{m}} = \mathbf{V}_0^{m_1} \otimes \cdots \otimes \mathbf{V}_0^{m_d}$ is a product space spanned by tensorizations of one dimensional Legendre polynomials. Any function $u_1(y_1, y_2) \in \mathbf{V}_0^{\boldsymbol{m}}$, $\boldsymbol{m} = \{m_1, m_2\}$ can be therefore expressed as:

$$u(y_1, y_2) = \sum_{i=0}^{m_1-1} \sum_{j=0}^{m_2-1} \delta_{i,j}\, \phi_i(y_1)\, \phi_j(y_2) \tag{2.59}$$

Note that the full tensor product space is needed in (2.59), resulting in a total number of terms equal to $m = m_1\, m_2$. This relates to the fact that one dimensional Alpert Multiwavelets are piecewise polynomial functions of maximum degree m_i.

We now seek a space $\mathbf{W}_0^{\boldsymbol{m}} \perp \mathbf{V}_0^{\boldsymbol{m}}$ containing continuous polynomials defined on the four quadrants of $[0,1]^2$, respectively. Any function $u_2(y_1, y_2) \in \mathbf{W}_0^{\boldsymbol{m}}$ can be expressed as:

$$\begin{aligned} u_2(y_1, y_2) = &\sum_{i=0}^{m_1-1} \sum_{j=0}^{m_2-1} \alpha_{i,j}\, \phi_i(y_1)\, \varphi_j(y_2) + \sum_{i=0}^{m_1-1} \sum_{j=0}^{m_2-1} \beta_{i,j}\, \phi_i(y_1)\, \varphi_j(y_2) + \\ &\sum_{i=0}^{m_1-1} \sum_{j=0}^{m_2-1} \gamma_{i,j}\, \phi_i(y_1)\, \varphi_j(y_2). \end{aligned} \tag{2.60}$$

By the orthogonality of the Legendre polynomials (rescaled on $[0,1]$) respect to the uniform measure and for the properties of the multiwavelet basis, we have:

$$\int_{[0,1]^2} u_1(y_1, y_2)\, u_2(y_1, y_2)\, d\Omega = 0 \quad \forall u_1 \in \mathbf{V}_0^{\boldsymbol{m}} \quad \text{and} \quad \forall u_2 \in \mathbf{W}_0^{\boldsymbol{m}} \tag{2.61}$$

which implies orthogonality of the two spaces. We finally note that any function f of maximum degree $(m_1 - 1)(m_2 - 1)$, continuous on the four quadrants of the unitary square, can be uniquely

determined by $4\,m_1\,m_2$ constants. If we expand this function according to the following expression:

$$f(y_1,y_2) = \sum_{i=0}^{m_1-1}\sum_{j=0}^{m_2-1}\delta_{i,j}\phi_i(y_1)\phi_j(y_2) + \sum_{i=0}^{m_1-1}\sum_{j=0}^{m_2-1}\alpha_{i,j}\phi_i(y_1)\varphi_j(y_2) + \sum_{i=0}^{m_1-1}\sum_{j=0}^{m_2-1}\beta_{i,j}\varphi_i(y_1)\phi_j(y_2) + \sum_{i=0}^{m_1-1}\sum_{j=0}^{m_2-1}\gamma_{i,j}\varphi_i(y_1)\varphi_j(y_2). \tag{2.62}$$

We can see that exactly $m_1\,m_2$ constant are needed for each of the δ, α, β, γ families of coefficients. So every function in $\mathbf{V}_1^{\boldsymbol{m}}$ can be uniquely determined as a combination of $\mathbf{V}_0^{\boldsymbol{m}} = \text{span}\{\phi_i(y_1)\otimes\phi_j(y_2), i=0,\ldots,m_1-1,\, j=0,\ldots,m_2-1\}$ and $\mathbf{W}_0^{\boldsymbol{m}} = \text{span}\{\phi_i(y_1)\otimes\varphi_j(y_2)\oplus\varphi_i(y_1)\otimes\phi_j(y_2)\oplus\varphi_i(y_1)\otimes\varphi_j(y_2), i=0,\ldots,m_1-1,\, j=0,\ldots,m_2-1\}$. Basis of $\mathbf{W}_j^{\boldsymbol{m}}$ are obtained as usual, by scaling and shifting operations.

For arbitrary dimensionality d the above procedure is generalized, providing basis for $\mathbf{V}_0^{\boldsymbol{m}}$ and $\mathbf{W}_0^{\boldsymbol{m}}$, where $\boldsymbol{m} = \{m_1,\ldots,m_d\}$. We first introduce an index set $\boldsymbol{i} = \{i_1,i_2,\ldots,i_d : 0 \leqslant i_j < m_j,\, j = 1,\ldots,d\}$ and define a single d-dimensional scaling function as:

$$\phi_{\boldsymbol{i}}(y_1,y_2,\ldots,y_d) = \phi_{i_1}(y_1)\ldots\phi_{i_d}(y_d). \tag{2.63}$$

The complete set of scaling function $\mathcal{S}_0^d$ is given by:

$$\mathcal{S}_0^d = \bigcup_{\boldsymbol{i}\in\boldsymbol{I}}\phi_{\boldsymbol{i}}(y_1,y_2,\ldots,y_d), \tag{2.64}$$

where the set $\boldsymbol{I}$ contains all the $\prod_{j=1}^d m_j$ possible combinations of index sets $\boldsymbol{i}$. Before writing the expression for the wavelet function we introduce the following notation:

$$\psi_i^0 = \phi_i, \quad \psi_i^1 = \varphi_i, \quad \psi_{\boldsymbol{i}}^{\boldsymbol{k}}(y_1,y_2,\ldots,y_d) = \psi_{i_1}^{k_1}(y_1)\ldots\psi_{i_d}^{k_d}(y_d) \quad \text{where} \quad \boldsymbol{k} = \{k_1,\ldots,k_d\}. \tag{2.65}$$

We also define the set $\boldsymbol{k}^q$ as the binary representation of q with d digits. The set $\mathcal{W}_0^d$ of functions spanning $\mathbf{W}_0^{\boldsymbol{m}}$ is defined as follows:

$$\mathcal{W}_0^d = \bigcup_{\boldsymbol{i}\in\boldsymbol{I}}\bigcup_{q=1,\ldots,2^d}\psi_{\boldsymbol{i}}^{\boldsymbol{k}^q}(y_1,y_2,\ldots,y_d). \tag{2.66}$$

Similarly to the one-dimensional case, scaled and shifted analogues of the mother multiwavelets in $\mathcal{W}_0^d$ are used to generate approximation spaces of increasing resolutions. In particular, we use the notation $\mathcal{W}_0^d(\boldsymbol{i},q)$ to identify single basis in $\mathcal{W}_0^d$, as they are uniquely determined by the multi-index $\boldsymbol{i}$ and permutation index q. At resolutions $j>0$ we also need to identify which of the 2^{jd} uniform subdivisions of $[0,1]^d$ contains the given basis. To this aim, the *coordinate* vector $\boldsymbol{s} = \{s_1,\ldots,s_d\}$, $0 \leqslant s_1,\ldots,s_d < 2^j$, is used. It follows that:

$$\mathcal{W}_j^d(\boldsymbol{i},q,\boldsymbol{s}) = \psi_{\boldsymbol{i},\boldsymbol{s}}^{\boldsymbol{k}^q}(y_1,y_2,\ldots,y_d) = 2^{jd/2}\,\psi_{i_1}^{k_1}(2^j y_1 - s_1)\ldots\psi_{i_d}^{k_d}(2^j y_d - s_d). \tag{2.67}$$

To conclude the present section, a simple example of approximation of a 2D function is illustrated in figure 2.6, using order $\boldsymbol{m} = \{0,0\}$ (Haar) and order $\boldsymbol{m} = \{1,1\}$ multiwavelets with maximum resolution level set to $j=3$ (8x8 subdivisions of the unit square).

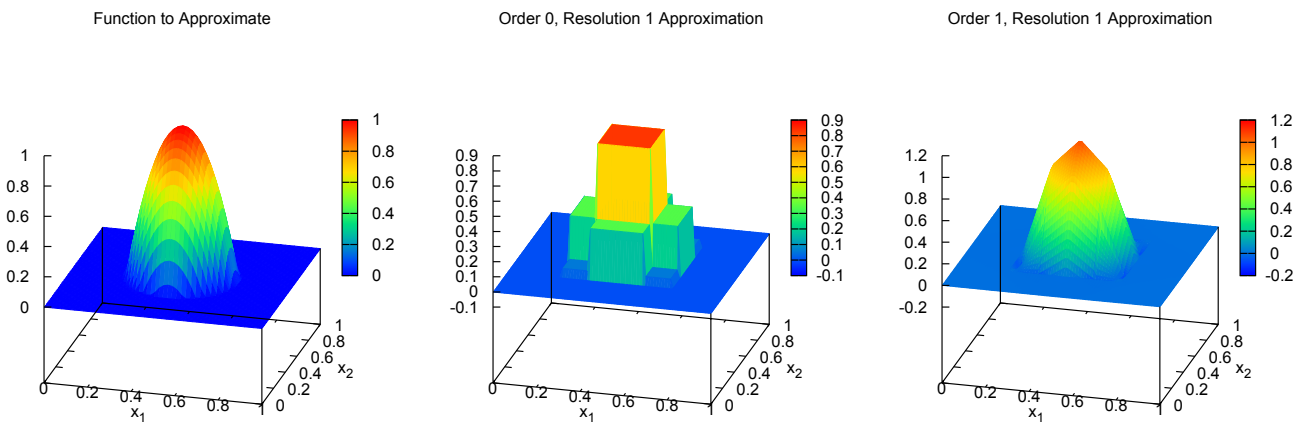


Figure 2.6: Examples of 2D Multiresolution approximation of a given function for $\boldsymbol{m} = \{0, 0\}$ and $\boldsymbol{m} = \{1, 1\}$.

Approximation Error in Multiwavelet Basis for deterministic sampling

Theoretical error bounds and decay rates for Multiwavelet approximation are also provided in [8]. For the one-dimensional case, consider a function $f : [0, 1] \to \mathbb{R}$ to be m-times differentiable, that is, $f(x) \in C^m([0, 1])$. We define $P_j^m f(x)$ as the approximation of the function f at resolution j, obtained using a multiwavelet dictionary with m vanishing moments. Interpolation with Chebyshev polynomials of order m, gives:

$$\|P_j^m f(x) - f(x)\| \leqslant 2^{-jm} \frac{2}{4^m\, m!} \sup_{x \in [0,1]} |f^{(m)}(x)| \tag{2.68}$$

The rate of convergence (see, e.g. [55]) is hence m for the one dimensional case and generalize to m/d for d dimensions. These are only a reference in our case as they assume all important scales are included at the selected finer resolution. Note also that sampling is random in the proposed framework as opposed to the Chebyshev points employed in the estimates above.

Remark 1 (Implementation details). Expression (2.67) can be used to derive software representations allowing efficient compression and reconstruction. It shows that five quantities are needed to identify single multi-dimensional multiwavelet basis, namely α_i (the expansion coefficient), j, $\boldsymbol{i}$, q, $\boldsymbol{s}$. In practice $\boldsymbol{i}$ is determined by constructing *multi-indexes*, thus defining a one-to-one map between the first $\prod_{i=1}^{d} m_i$ numbers in $\mathbb{N}$ and the permuations of d indexes i_k where $0 \leqslant i_k < m_k$. Similarly, $\boldsymbol{s}$ is identified associating the set $\{0, \ldots, 2^{jd} - 1\}$ to coordinate vectors in $\mathbb{R}^d$.

Remark 2 (Numerical evaluation of 1D Alpert Multiwavelet Basis). A carefully selected numerical integration scheme should be selected when constructing multiwavelet bases for $\mathbf{W}_0^m$; such functions are in fact discontinuous in $x = 1/2$. In our case, a separate numerical integration in $[0, 1/2)$ and $[1/2, 1]$, respectively, was implemented using a Clenshaw-Curtis rule. Numerical integration loops are thus retained but with *double* quadrature points and weights modified accordingly.

2.10 Construction of Alpert Multiwavelets

B. Alpert [9] introduced a *Multiwavelets* basis which lead to the sparse representation of smooth integral operators on a finite interval. We benefit from the Multiresolution analysis framework discussed above and define the following indexes:

- j is the resolution (or scale) index.
- k is the translation (or shift) index.
- m is the order index.

Consider $P_m \subset \mathbf{L}^2(\mathbb{R})$ as the space of polynomials with order less than k. Furthermore, we define the interval $S_{j,k}$ as

$$S_{j,k} = (2^j k, 2^j(k+1)) \tag{2.69}$$

We then define:

$$\mathbf{V}_j^m = \{f(x) : \text{if } g(x) \in P_m \Rightarrow f(x) = I(S_{j,k})g(x)\} \tag{2.70}$$

where $I(S_{j,k})$ is the indicator function for the interval $S_{j,k}$. In other words, $\mathbf{V}_j^m$ is the space of polynomials with order less than k, restricted to the interval $S_{j,k}$. A nested sequence of these spaces can be defined as follows:

$$\mathbf{V}_0^m \subset \mathbf{V}_1^m \subset \cdots \subset \mathbf{V}_{r_{max}}^m \subset \ldots$$

If we consider $f(x) \in V_j^m$ as the scaling functions of a multiresolution approximation, the *Alpert Multiwavelets* can be constructed as the basis of the associated detail space. In [8] a procedure is reported to build a basis for the space $\mathbf{O}_0^m$, the orthogonal complement of $\mathbf{V}_0^m$ in $\mathbf{V}_1^m$. A typical decomposition of $\mathbf{V}_j^m$, which will be used in the schemes developed in the following sections, is:

$$\mathbf{V}_j^m = \mathbf{V}_0^m \oplus \mathbf{O}_0^m \oplus \mathbf{O}_1^m \oplus \cdots \oplus \mathbf{O}_{2^{j-1}}^m \tag{2.71}$$

In other words, we employ the Legendre Basis only at the coarser level (the full $[0,1]$ support) expressing finer approximations in terms of Multiwavelet basis associated with detail spaces.

As a first step, the functions $f_1, \ldots, f_m$ are constructed with support in $[-1,1]$, with given properties. A basis $\psi_1, \ldots, \psi_m$ for $\mathbf{O}_0^m$, is then produced by squeezing the functions f_i on a unitary $[0,1]$ support.

2.10.1 Properties of the function set F

A family of functions f_i are build according to the following properties:

1. The set $F = \{f_i : i = 1, \ldots, m\}$ contains one dimensional real functions defined in $[-1,1]$.
2. The restriction of f_i to the interval $[0,1)$ is a polynomial of degree $m-1$.
3. The function f_i is extended to the interval $[-1,0]$ as an even or odd function according to the parity of $i+k-1$.
4. The functions in F satisfy the following orthonormality conditions:

$$\int_{-1}^{1} f_i(x) f_j(x)\, dx = \langle f_i, f_j \rangle = \delta_{i,j}, \quad i,j = 1, \ldots, m. \tag{2.72}$$

5. A function $f_j \in F$ has the following vanishing moments:

$$\int_{-1}^{1} f_j(x) x_i\, dx = 0, \quad i = 0, 1, \ldots, j+k-2. \tag{2.73}$$

The properties 2 and 3 define m^2 degrees of freedom in the choice of the set F, while 4 and 5 provide m^2 non trivial constraints, see [8].

2.10.2 Incremental Construction from Basis Properties

The original scheme proposed by Alpert, provide a methodology in four steps to build the Multiwavelet function vector $\boldsymbol{\psi} = \{\psi_i, i = 0, \ldots, m-1\}$. In the present paragraph we repeat that construction to provide a unitary description of our UP framework. The functions $f_1^1, f_2^1, f_3^1, \ldots, f_m^1$ are defined as:

$$f_i^1(x) = \begin{cases} x^{i-1}, & x \in (0,1], \\ -x^{i-1}, & x \in [-1,0], \\ 0 \quad \text{otherwise} \end{cases} \tag{2.74}$$

1. Gram-Schmidt orthogonalization is performed for functions f_i^1 with respect to $1, x, \ldots, x^{m-1}$, to generate the set $\{f_i^2, i = 1, \ldots, m\}$.

2. The next steps yield $m-1$ functions orthogonal to x^m, of which $m-2$ funtions are orthogonal to x^{m+1}, down to 1 function which is orthogonal to x^{2m-2}. First, if at least one of f_i^2 is not orthogonal to x^m, we reorder the functions such that it appears first, $\langle f_1^2, x^m \rangle \neq 0$. We then define $f_i^3 = f_i^2 - a_i f_0^2$ where a_j is chosen so $\langle f_i^3, x^m \rangle = 0$ for $i = 2, \ldots, m$, achieving orthogonality to x^m. Similarly, we orthogonalize to $x^{m+1}, \ldots, x^{2m-2}$, each in turn, to obtain $f_1^2, f_2^3, f_3^4, \ldots, f_m^{m+1}$, such that $\langle f_i^{i+1}, x^j \rangle = 0$ for $i \leqslant j + m - 2$.

3. Gram-Schmidt orthogonalization is performed on the functions $f_k^{k+1}, f_{k-1}^k, \ldots, f_1^2$.

4. A normalization operation yields to the family $f_m, f_m - 1, \ldots, f_1$.

A set of basis for $\mathbf{O}_0^m$, defined on $[0,1]$, is then obtained by the following expression:

$$\psi_i(x) = 2^{1/2} f_i(2x-1), \quad i = 1, \ldots, m. \tag{2.75}$$

Please note that Alpert multiwavelet basis are unique (up to sign). Similar methodologies could also be employed to build orthonormal bases resulting orthogonal to the monomials with degree less than m. For example, the procedure reported in [64] is similar to that suggested by Alpert, the main difference being step 5 in the definition of f_i (or step 2 above), where both indexes are defined as $i, j = 0, \ldots, k-1$. A graphical representation of the one dimensional Alper Multiwavelet basis for $\mathbf{O}_0^m$ is illustrated in figure 2.5 for $m = 3$ and $m = 4$ respectively.

Chapter 3

A Compressed Sensing Approach to Uncertainty Propagation

3.1 Rudiments of Compressive Sampling

Compressive Sampling (CS) is a new direction in signal processing that breaks the traditional limits of the Shannon-Nyquist sampling rate for reconstruction of sparse signals. Consider a vector of measurements $\mathbf{u} = (u(y^{(1)}), \dots, u(y^{(M)}))^T \in \mathbb{R}^M$ of $u \in \mathbf{L}^2([0,1])$. Assuming that u admits a multiwavelet expansion of the form (2.58) with some finite m and resolution j, $\mathbf{u}$ can be represented as $\mathbf{u} = \boldsymbol{\Psi}\,\boldsymbol{\alpha}$, where the so-called *measurement* matrix $\boldsymbol{\Psi} \in \mathbb{R}^{M\times P}$ contains the realization of the multiwavelet basis $\{\psi_i(y)\}$ corresponding to $\mathbf{u}$ and $\boldsymbol{\alpha} \in \mathbb{R}^P$ is the vector of unknown expansion coefficients. Here, P is the cardinality of the truncated multiwavelet basis. Then u has a sparse multiwavelet representation if $\|\boldsymbol{\alpha}\|_0 = \#\{\alpha_i : \alpha_i \neq 0\} \ll P$. For a *sufficiently* sparse u, CS recovers u exactly using some $M \ll P$ measurements by solving an optimization problem of the form

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^P} \|\boldsymbol{\alpha}\|_s \quad \text{subject to} \quad \mathbf{u} = \boldsymbol{\Psi}\,\boldsymbol{\alpha}. \tag{P_s}$$

The sparsest solution $\boldsymbol{\alpha}$ to (P_s) corresponds to $s = 0$, i.e., minimizing the ℓ_0 semi-norm $\|\boldsymbol{\alpha}\|_0$, which is generally NP-hard to compute. To break this complexity several heuristics based on greedy pursuit, e.g., Orthogonal Matching Pursuit (OMP), and convex relaxation via ℓ_1-minimization, i.e., $s = 1$, have been proposed, among other approaches. Moreover, several metrics such as the *mutual coherence*, [19], or the *restricted isometry property*, [21], have been introduced to provide guarantees on the uniqueness of the sparsest solution to (P_s) as well as the ability of the heuristic approaches in recovering the solution. In particular, the mutual coherence of $\boldsymbol{\Psi}$ (e.g., see [19]) is defined as

$$\mu(\boldsymbol{\Psi}) = \max_{i\neq j} \frac{|\boldsymbol{\psi}_i^T \boldsymbol{\psi}_j|}{\|\boldsymbol{\psi}_i\|_2 \, \|\boldsymbol{\psi}_j\|_2}, \tag{3.1}$$

where $\boldsymbol{\psi}_i \in \mathbb{R}^M$ is the i-th column of $\boldsymbol{\Psi}$. Note that $\mu(\boldsymbol{\Psi}) \in [0,1]$ in general, and that it is strictly positive for $M < P$. Depending on the sparsity level $\|\boldsymbol{\alpha}\|_0$,, the mutual coherence provides a sufficient condition on the number M of measurements for a successful recovery of $\boldsymbol{\alpha}$ from P_s, as shown in [19].

Finally, for cases of signals which are nearly sparse of affected by errors, a noise-tolerant version of (P_s) can be written as:

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^P} \|\boldsymbol{\alpha}\|_s \quad \text{subject to} \quad \|\mathbf{u} - \boldsymbol{\Psi}\,\boldsymbol{\alpha}\|_2 \leqslant \epsilon. \tag{$P_{s,\epsilon}$}$$

3.2 Sparse Reconstruction Algorithms

3.2.1 Semi-norm Relaxations and Greedy Pursuits

Sections 3.2.2 and 3.2.3 have been explicitly devoted to discuss the OMP and TOMP strategies, employed in our numerical investigations. As a large body of literature exists on methodologies providing linear representations of signals according to redundant dictionaries of functions (sometimes addressed as *waveforms* or *atoms*), we here discuss some of the main available alternatives.

The matching pursuit algorithm (MP) [67] naturally opens our discussion, as it provides ground for many signal recovery techniques. It provides a *sensing* mechanism where coefficients are progressively chosen based on their correlation with the current residual vector. An implementation is suggested in [67], where an initial dictionary is selected and new atoms are progressively added based on local maximum correlations. A recursive update procedure is also discussed for the residual which makes the algorithm appealing for large problems as demonstrated with examples adopting time-frequency, Gabor and wavepacket dictionaries. Note that the residual vector produced by MP is only orthogonal to the last selected waveform. This might lead to slow convergence even if the selected atoms form a basis for the underlying vector space.

CoSaMP [73] and StOMP [35] are greedy heuristics built on top of the OMP algorithm. Both approaches use the quantity $\mathbf{\Psi}^T \mathbf{r}_k$ (referred as *signal proxy* and *matched filter*, respectively) to select atoms associated with significant expansion coefficients. They accommodate fast matrix-vector products, both for extracting the above correlations and for iteratively solve of least squares problem. As a consequence of including multiple atoms in the support set, speed ups are obtained respect to a standard OMP implementation. Thresholding and pruning operations are implemented differently for the two algorithms. Convergence results are provided for uniform spherical, uniform random and Gaussian matrix ensembles or for matrices associated with bounded restricted isometry constants.

Reference sparsity-undersampling tradeoffs are obtained using l_1 relaxations of the l_0 seminorm. These can be obtained by linear programming techniques such as Simplex or Interior Point methods, see [25]. The computational cost associated to solving large scale problems can however be a concern for these approaches.

Iterative thresholding techniques solve linear inverse problems by minimizing l_1 regularized convex problems of the form:

$$\min_{\mathbf{x}\in\mathbb{R}^P}\{f(\boldsymbol{\alpha}) + \lambda\|\boldsymbol{\alpha}\|_1\} \quad \text{where} \quad \lambda \in \mathbb{R}_{\geqslant 0},\ f(\mathbf{x}) = \|\mathbf{\Psi}\boldsymbol{\alpha} - \mathbf{u}\|_2 \tag{3.2}$$

Due to the separability of the l_1 norm, a typical ISTA (Iterative Shrinkage-Thresholding Algorithm) iteration is expressed as:

$$\mathbf{x}_k = \mathcal{T}_{\lambda t_k}(G(\mathbf{x}_{k-1})) = \mathcal{T}_{\lambda t_k}(\mathbf{x}_{k-1} - t_k \nabla f(\mathbf{x}_{k-1})) = \mathcal{T}_{\lambda t_k}(\mathbf{x}_{k-1} - 2\, t_k \mathbf{\Psi}^T(\mathbf{\Psi}\, \mathbf{x}_k - \mathbf{u})) \tag{3.3}$$

where $\mathcal{T}_\alpha : \mathbb{R}^P \to \mathbb{R}^P$ is the thresholding operator, $t_k > 0$ is a suitable step parameter and G a gradient operator. Iteration (3.3) is computationally appealing. The main cost relates to multiplications with matrices $\mathbf{\Psi}^T$ and $\mathbf{\Psi}$ and can be efficiently implemented with fast matrix-vector operators available in many cases, while component-wise thresholding is relatively inexpensive. Unfortunately ISTA converges slowly only at a sublinear global rate. Successful attempts to improve this convergence rate to almost quadratic were explored in [14] resulting in the FISTA approach, with modified iteration of the form $\mathbf{x}_k = \mathcal{T}_{\lambda t_k}(G(\mathbf{y}_{k-1}))$. Even with improved convergence rates, FISTA still shows sparsity-undersampling tradeoffs which are significantly worse than obtained with

Linear Programming optimizations. Improved tradeoffs for sparse signal recovery are obtained with the Approximate Message Passing (AMP) algorithm [37].

An optimization approach is followed in the development of the spgl1 algorithm [90], tracing the Pareto curve between least squares fit and l_1 norm of the solution. This algorithm scales well to large problems, requiring only matrix-vector multiplications.

The use of Iteratively re-weighted least squares minimization techniques (IRLS) for sparse recovery is investigated in [32]. Minimum l_p, $p \in (0, 1]$ norm solutions are obtained by weighted surrogates of the l_2 norm. A possible implementation is discussed in [24].

Finally, we mention re-weighted l_1 minimization strategies, developed in [22]. In [41] it is shown that a careful choice of weighted l_1 norms might result in better reconstructions of a piecewise smooth signal respect to the TMP or TOMP heuristics.

Note that sparsity and *affordable* maximum resolution are two inter-related concepts. In theory, one can always obtain sparse representations of a piecewise smooth response with a sufficiently large dictionary. In practice, one can afford only dictionaries where a solution of the associated P_0 problem can be computed in a reasonable time. This means that, for a general response, computational constraints might lead us to violate the assumption of sparsity resulting in less favorable undersampling for accurate reconstruction. Nonetheless, OMP heuristics were developed before recent trends in CS and adopted to solve linear inverse problems with general waveforms. This motivates our preference for greedy MP-based heuristics.

3.2.2 OMP Algorithm

The orthogonal matching pursuit algorithm (OMP) is a widely used strategy for the solution of P_0. It improves on the matching pursuing heuristics by computing a residual vector which is orthogonal to all the atoms included in the index set. This can be seen considering $\mathbf{\Psi}_{\mathcal{I}_k}$, the restriction of the measurement matrix to the index set at the current iteration and writing:

$$\mathbf{\Psi}_{\mathcal{I}_k}^T \mathbf{r}_k = \mathbf{\Psi}_{\mathcal{I}_k}^T (\mathbf{u} - \mathbf{\Psi}_{\mathcal{I}_k} \boldsymbol{\alpha}_k) = \mathbf{\Psi}_{\mathcal{I}_k}^T \mathbf{u} - \mathbf{\Psi}_{\mathcal{I}_k}^T \mathbf{\Psi}_{\mathcal{I}_k} \boldsymbol{\alpha}_k. \tag{3.4}$$

If an approximate solution $\boldsymbol{\alpha}_k$ is evaluated using least squares, i.e. $\boldsymbol{\alpha}_k = (\mathbf{\Psi}_{\mathcal{I}_k}^T \mathbf{\Psi}_{\mathcal{I}_k})^{-1} \mathbf{\Psi}_{\mathcal{I}_k}^T \mathbf{u}$, then $\mathbf{\Psi}_{\mathcal{I}_k}^T \mathbf{r}_k = \mathbf{0}$, demonstrating the properties of the algorithm.

For each iteration, only one atom (normalized measurement columns are assumed here) is incrementally added to the support set, based on the *correlations* $|\langle \boldsymbol{\psi}_i, \boldsymbol{r}_k \rangle|$.

Given the localized nature of the employed multiresolution dictionary and as a result of undersampling, it can happen that $\boldsymbol{\psi}_{i^*} = \mathbf{0}$ for some $i^* \in \{1, \ldots, P\}$. A *degenerate* atom thus results. In this case, $\beta(i^*) = 0$ (see Algorithm 1) is set, avoiding i^* to be inserted in the index set $\mathcal{I}$. Once all columns with $\boldsymbol{\psi}_{i^*} \neq \mathbf{0}$ have been added to the support set, OMP must therefore terminate. Finally, we use the LSMR algorithm [44] to solve for least squares at every iteration.

3.2.3 TOMP Algorithm

The observation that piecewise smooth functions are characterized by a connected subtree representation in Wavelet space leads to modified implementations of the OMP algorithm, as described in [60, 59, 40]. Atoms are progressively inserted in the support set together with their ancestors, in an effort to perpetuate their connected subtree structure. As already discussed for the CoSaMP

Algorithm 1 Orthogonal Matching Pursuit Algorithm - OMP

Inputs:

Measurement Matrix $\mathbf{\Psi}$.
RHS Vector $\mathbf{u}$.
Maximum allowable iterations k_{max}.
Convergence Tolerance δ.

Outputs:

Solution Vector $\boldsymbol{\alpha}$.

Initialize:

Iteration Count $k \leftarrow 0$.
Initial Solution $\boldsymbol{\alpha}_0 \leftarrow \mathbf{0}$.
Initial Residual $\boldsymbol{r}_0 \leftarrow \mathbf{u}$.
Initial Support set $\mathcal{I} \leftarrow \{0\}$.
$\mathbf{W} \leftarrow diag(\|\boldsymbol{\psi}_i\|_2)$
Set to unitary columns $\tilde{\mathbf{\Psi}} \leftarrow \mathbf{\Psi}\,\mathbf{W}$.

while $\|\boldsymbol{r}_k\|_2 > \delta$ And $k < k_{max}$ **do**

(*Sweep*)
for all $i \notin \mathcal{I}_k$ **do**
$\beta(i) = |\tilde{\boldsymbol{\psi}}_i^T \boldsymbol{r}_k|$.
end for
$k \leftarrow k + 1$.
(*Update Support Set*) $\mathcal{I}_k = \mathcal{I}_{k-1} \bigcup \{\arg\max_i \beta(i)\}$.
(*Solve with LS*) $\tilde{\mathbf{\Psi}}_k = \{\tilde{\boldsymbol{\psi}}_1, \ldots, \tilde{\boldsymbol{\psi}}_k\}\, \forall\, i \in \mathcal{I}_k$. Solve $\tilde{\mathbf{\Psi}}_k^T \tilde{\mathbf{\Psi}}_k\, \boldsymbol{\alpha}_k = \tilde{\mathbf{\Psi}}_k^T \mathbf{u}$.
(*Update Residual*) $\boldsymbol{r}_k = \mathbf{u} - \tilde{\mathbf{\Psi}} \boldsymbol{\alpha}_k$.

end while

Return:

$\mathbf{W}^{-1}\, \boldsymbol{\alpha}_k$

Algorithm 2 Tree-based Orthogonal Matching Pursuit Algorithm - TOMP

Inputs:

The number of samples M, basis cardinality P.
Matrix $\mathbf{\Psi}$
Vector $\mathbf{u}$
Maximum allowable iterations k_{max}
Tolerance δ.
Coefficients γ, ρ

Outputs:

Solution Vector $\boldsymbol{\alpha}$.

Initialize:

$k \leftarrow 0$.
Initialize Residual $\boldsymbol{r}_0 \leftarrow \mathbf{u}$.
Initialize index set $\mathcal{I}_0 \leftarrow \{0\}$.
$\mathbf{W} \leftarrow diag(\|\boldsymbol{\psi}_i\|_2)$
Set to unitary columns $\tilde{\mathbf{\Psi}} \leftarrow \mathbf{\Psi}\,\mathbf{W}$.
Define $\mathcal{I}$ as the set of all columns of $\tilde{\mathbf{\Psi}}$.

while $\|\boldsymbol{r}_k\|_2 > \delta$ And $k < k_{max}$ And $card(\mathcal{I}_k) < \rho\, M$ **do**

$c_k^i = |\tilde{\boldsymbol{\psi}}_i^T \boldsymbol{r}_{k-1}|$ where $i \in \mathcal{I} \backslash \mathcal{I}_{k-1}$
$c_k^{i,Max} = \max_i \{c_k^i : i \in \mathcal{I} \backslash \mathcal{I}_{k-1}\}$
$S_k = \{i : c_k^i \geqslant \gamma\, c_k^{i,Max}\}$
for all $i \in S_k$ **do**
$F_i \leftarrow \text{Anc}(i)$.
Assemble $\tilde{\mathbf{\Psi}}_k$ with $\mathcal{I}_{k-1} \cup F_i$.
$\boldsymbol{\alpha}_i = (\tilde{\mathbf{\Psi}}_k^T \tilde{\mathbf{\Psi}}_k)^{-1} \tilde{\mathbf{\Psi}}_k^T \mathbf{u}$
$r(i) = \|\mathbf{u} - \tilde{\mathbf{\Psi}}_k\, \boldsymbol{\alpha}_i\|_2$
end for
$i_k = \arg\min_{i \in S_k} r(i)$
$\mathcal{I}_k = \mathcal{I}_{k-1} \cup F_{i_k}$.
$\boldsymbol{r}_k = \boldsymbol{r}_{i_k}$.

end while

Return:

$\mathbf{W}^{-1}\, \boldsymbol{\alpha}_k$

and StOMP algorithms, faster execution times result by *sensing* more than one index per iteration respect to OMP. Two slightly different approaches have been recently proposed in the literature.

We start by clarifying some common notation. Atoms in a multiresolution representation are naturally associated to a *scale* index j and *shift* index $k \in \{0, \dots, 2^j - 1\}$. As wavelets satisfy a two-scale refinement equation and as the number of atoms at successive resolution increase by a factor of two, the representation is isomorphic to a binary tree (T). In this Section, we denote atom i with scale and shift (i, j). An atom $(i, j) \in \mathcal{T}$ is the *father* of $(i + 1, 2j)$ and $(i + 1, 2j + 1)$; the *ancestors* of (i, j) are denoted by $Anc(i, j)$ and are obtained by recursively including fathers up to the root. Note that $(i, j) \in Anc(i, j)$. Similarly, the *descendants* of (i, j), $Desc(i, j)$ or $Desc(i, j)_b$ result from including all the children of (i, j) up to the maximum available scale index, or a specific limit $i = b$. A binary tree is *connected* if $(i, j) \in \mathcal{T}$ implies $Anc(i, j) \in \mathcal{T}$. An atom is a *leaf* for $\mathcal{T}$ if $Anc(i, j) \in \mathcal{T}$ while $Desc(i, j) \notin \mathcal{T}$.

In [40] a b-TMP procedure is proposed where two index sets are iteratively updated. At iteration k, S_k contains the indexes already in the support, while C_k contains the possible candidates. Once an index i with maximum correlation is found in $S_k \cup C_k$ then it is added to S_k, while C_k is updated with $Desc(i, j)_b$. In other words, b is used as a *look ahead* parameter which allows to progressively include atoms which conform to the connected tree structure.

In [60, 59], the sensing step leads to a set S_k of candidate *leaves* with cardinality $|S_k|$. In practice, all atoms with $c_k^i \geqslant \gamma\, c_k^{i,Max}$ are inserted in S_k. For all $i \in S_k$, a set F_i is formed containing node i together with $Anc(i, j)$. The node i generating the minimum residual from $|S_k|$ least squares problems is now added to the support set. For a given number of samples M, the iterations are stop whether $|S_k| > \rho M$. Optimal values of $\gamma = 0.975$ and $\rho = 2.0$ are suggested in [60, 59].

Our implementation follows this second approach which, in our opinion, gives somehow more flexibility in the sensing stage, though this may results in a limited computational overhead.

The above algorithms were developed for scalar tree representations; vector Multiwavelets trees are used in our case. While all the other steps remain practically unchanged, the selection of the ancestor set can be implemented in different flavors. For example, in case a leaf is selected whose basis has m vanishing moments, we might wonder how many vanishing moments should be allowed for the basis to be included in the ancestor set. In our implementation, we choose to include multiwavelet with all vanishing moments.

Finally, we remark that the efficiency of the signal reconstruction procedure is usually dependent on the noise parameter δ. An excessively small value of δ might result in missing the recovery of a sufficiently close sparse approximation of the response, while larger values of δ might tolerate excessive noise. In [39, 17] a procedure based on cross validation is proposed to find an optimal value for the noise tolerance. This technique is surely effective in reducing the number of parameters involved but requires multiple solutions of smaller-size problems, which may affect the overall reconstruction time, in particular for large measurements setups.

3.3 Recovery Performance of Multiwavelet Measurements

Phase diagrams [33] offer intuitive representations of sparsity-undersampling tradeoffs for CS recovery. In the present study, they are employed to compare between selected matrix ensembles. Gaussian, multiwavelets and measurements matrices assembled from preconditioned bounded orthonormal systems [79] are used in our examples. Parameters have been selected as shown in tables 3.1 and 3.2. An average mutual coherence is also computed from the repeated reconstructions for every point in the diagrams.

Parameter	Value
Steps along the δ axis	40
Steps along the ρ axis	40
Number of repetitions for every single reconstruction	100
Solution matching tolerance (relative l_2 error norm)	10^{-3}
Solver tolerance on the residual norm relative to the RHS	10^{-5}
Maximum number of iterations	3000

Table 3.1: Parameters for Phase Diagram generation

Case	Matrix Type	Sampling	Dim	m	r_{max}	P	Figure
1	Gaussian	-	-	-	-	200	Fig. 3.1
2	Rescaled Legendre	Uniform	5	3	-	252	Fig. 3.2
3	Rescaled Legendre	Chebyshev	5	3	-	252	Fig. 3.3
4	Multiwavelet	Uniform	3	2	0	216	Fig. 3.4
5	Multiwavelet	Chebyshev	3	2	0	216	Fig. 3.5
6	Multiwavelet	Uniform	2	2	1	144	Fig. 3.6

Table 3.2: Generated Phase Diagrams

Figure 3.1, generated using Gaussian random measurements produces the expected sharp transition. Tradeoffs for random Legendre ensembles confirm that Chebyshev sampling is optimal for such orthogonal system. Smaller mutual coherences and higher success rates are obtained using Chebyshev sampling (with associated preconditioner), as expected. Uniform and Chebyshev sampling have also been compared for random multiwavelet ensembles with lowest resolution atoms. Opposite trends can be observed compared to Legendre case. This suggests that optimality of the Chebyshev measure cannot be easily extended to multiresolution orthogonal systems. Furthermore, we focus our attention to random multiwavelet dictionaries with atoms at maximum resolutions 0 and 1, respectively. Results from uniform sampling show how the coherence of this basis set increases by including higher resolutions. This leads to less favorable transitions.

Finally, note that magnitudes of exact solution coefficients are drawn from a standard normal distribution in our tests, while the support is obtained by successive scrambling. This may alter our perception for performances associated to real stochastic responses.

3.4 Sampling Strategies for CS-MW

A natural way to obtain the measurements $\mathbf{u}$ is to generate random realizations of the input y and evaluate the corresponding solution $u(y)$. However, for situation where u exhibits, for instance, sharp gradients or discontinuities, such sampling strategy may not necessarily lead to accurate approximation, using CS with a limited number of realizations. This is because the higher resolution basis functions needed to capture the local structure of u may not be sampled enough to constitute a well-conditioned measurement matrix $\mathbf{\Psi}$.

In [79], for example, the optimality of Chebyshev sampling is discussed for a large class of orthogonal polynomial systems. We therefore start by a comparison between Chebyshev and Uniform sampling for multiresolution dictionaries. An Importance Sampling approach is discussed next, allowing local accumulation of samples.

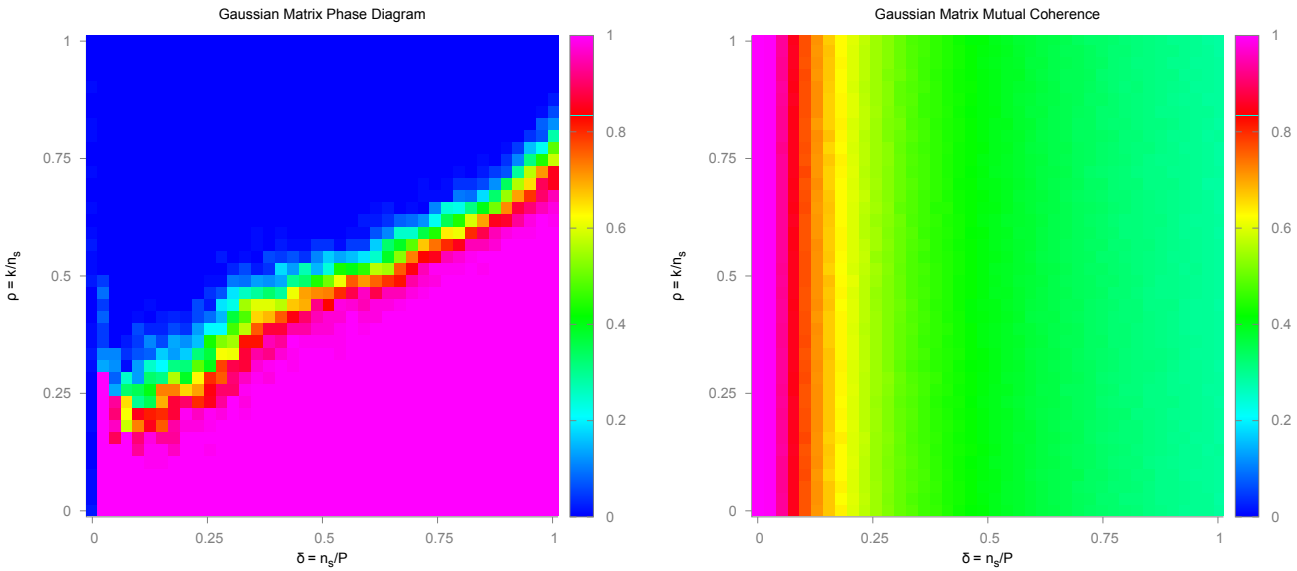


Figure 3.1: Phase Diagram for Gaussian Matrix

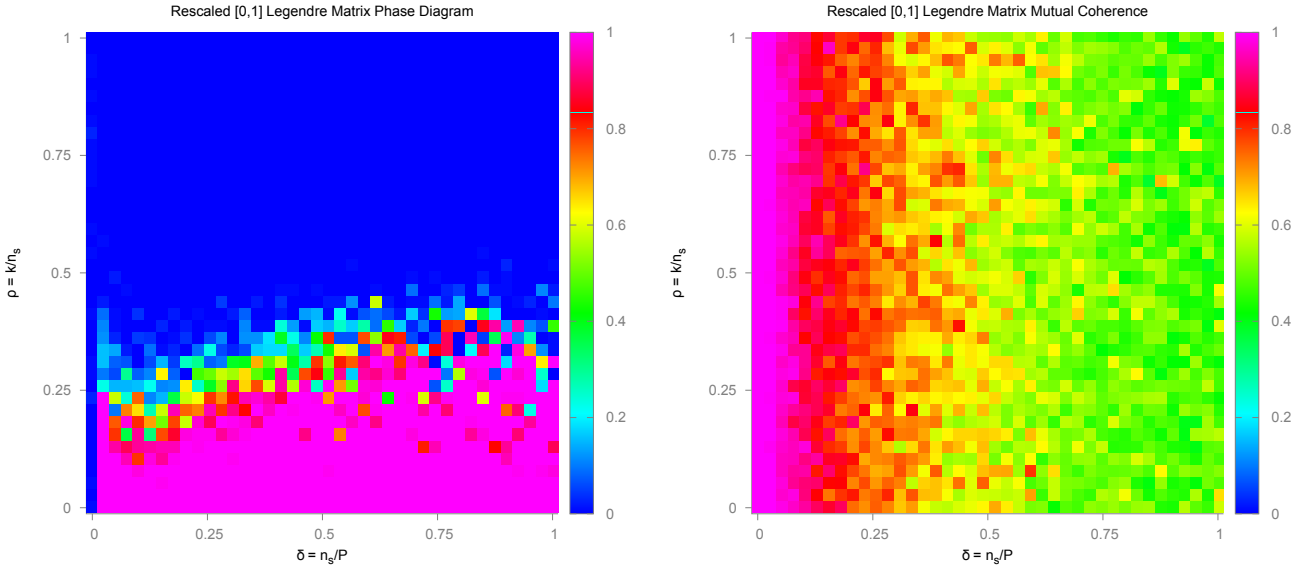


Figure 3.2: Phase Diagram for a Legendre Measurement Matrix - Uniform Sampling

3.4.1 Optimality of Chebyshev Sampling for MRA

The Chebyshev probability measure on $[0,1]$ is defined as $\rho_C(y) = 2.0/(\pi\sqrt{1-(2y-1)^2})$. A uniform random variable $y \in [0,1]$ can be mapped to a random variable $\hat{y} \in [0,1]$ with Chebyshev distribution using $\hat{y} = (sin[\pi(y-0.5)]+1)/2$. The family of rescaled Legendre polynomials on $[0,1]$ is denoted as $L_i(y) : \Omega \rightarrow \mathbb{R}$, where $y \in \Omega$ is a random variable associate with distribution $\rho(y)$ and $i \in \mathbb{N}$ is the degree of every member. In [79] it is observed that a signal which is s-sparse in a Legendre basis $L_i(y), i = \{0,\ldots,P-1\}$, not affected by noise, can be reconstructed exactly solving P_1 on a number of Chebyshev distributed random sampling points equal to $M \geqslant C\, s\, log^4(P)$, where C is a constant. The function $g(y) = \sqrt{\rho_C(y)}$ provides a square-integrable envelope function for $L_i(y)$ and therefore measurement matrices with atoms $L_i(y)/g(y)$ will exhibit bounded restricted isometry constants consistent with that of a uniformly bounded, orthogonal systems.

In an effort to extend the previous argument to MRA, we note that, at the coarsest resolution, $\phi_i(y) = L_i(y)$ and therefore the Chebyshev measure is optimal for the global scaling family. However, multiwavelet analogues $\varphi_i(y)$ are discontinuous with unbounded values at $\{0, 1/2, 1\}$, for increasing i.

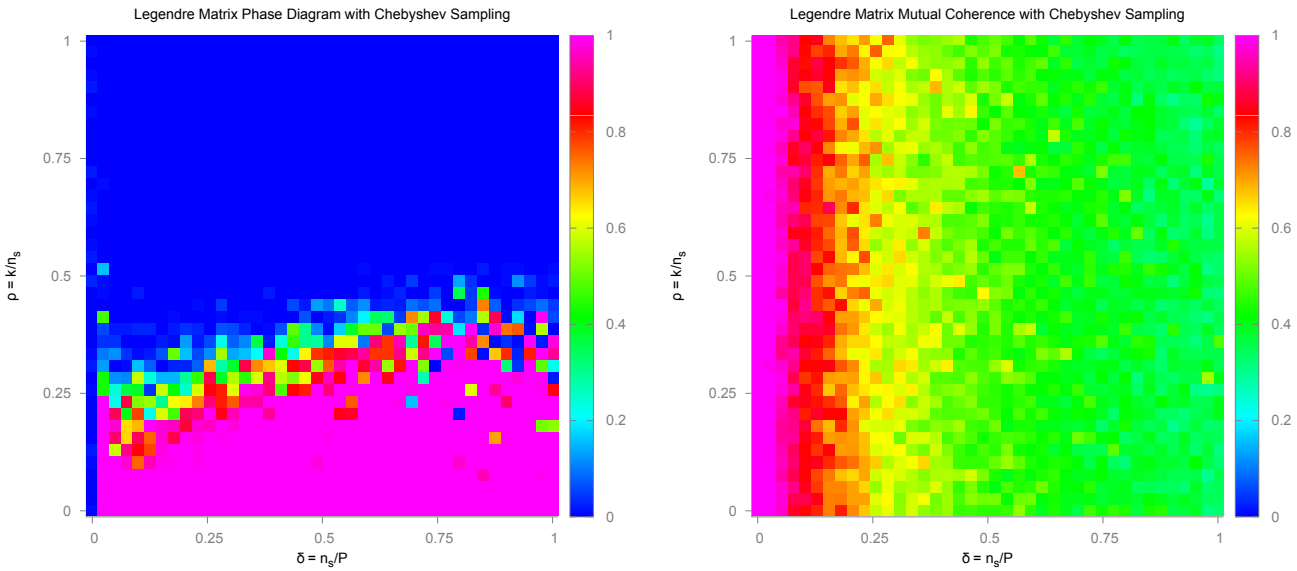


Figure 3.3: Phase Diagram for a Legendre Measurement Matrix - Chebyshev Sampling

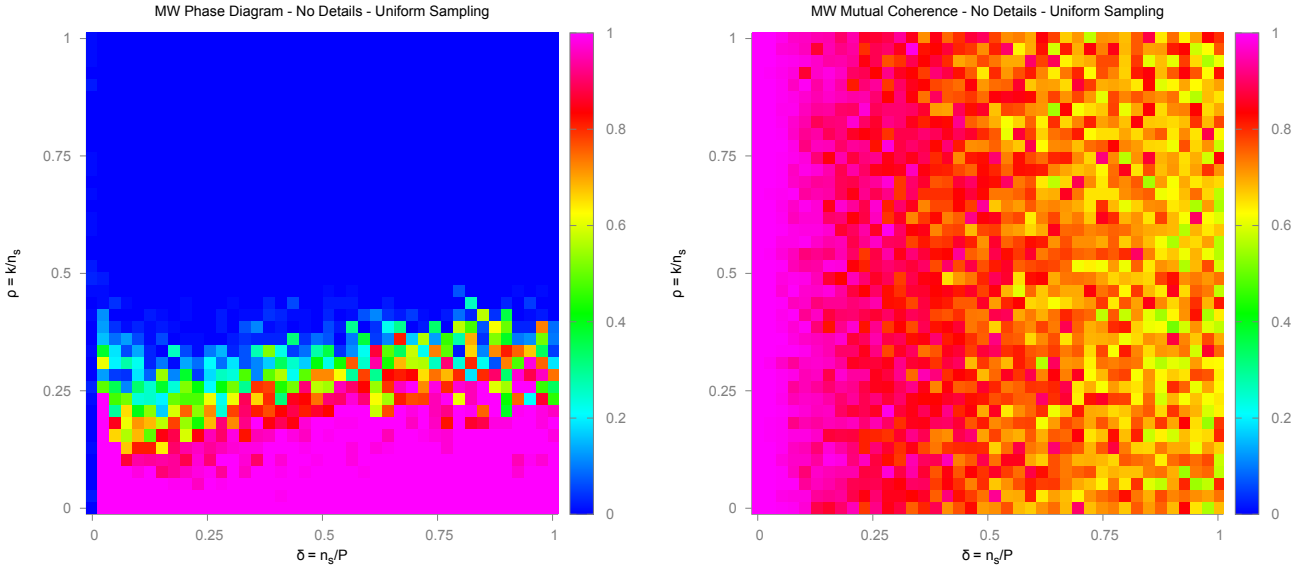


Figure 3.4: Phase Diagram for a Multiwavelet Measurement Matrix with no details - Uniform Sampling

As $g(1/2) = 1$, it follows that $\varphi_i(1/2)/g(1/2)$ is not bounded for increasing i. This is also suggested by the fact that $\mathbf{V}_j \oplus \mathbf{W}_j = \mathbf{V}_{j+1}$, and what happens in $\{0, 1\}$ with $L_i(y)$ simply translates to $\bigcup_{k=0,\ldots,2^j-1}\{2^{-j}\, k, 2^{-j}\,(k+1)\}$ for arbitrary j. While a simple extension applies *multiple* Chebyshev measures within every partition at the finer resolution, we won't explore further this possibility in the present work.

Our arguments are also supported by numerical experiments. A subset of the phase diagram results for cases 2,3,4,5 (table 3.2) are extracted to facilitate an immediate comparison. Figure 4.3 contains the results obtained for the Legendre and Multiwavelet ensembles. Cumulative distributions of mutual coherence are determined for fixed sparsity-undersampling ratios. The best results are observed for the Chebyshev-Legendre and Uniform-multiwavelet matches.

Next, a different sampling strategy is proposed which might further improve on uniform sampling.

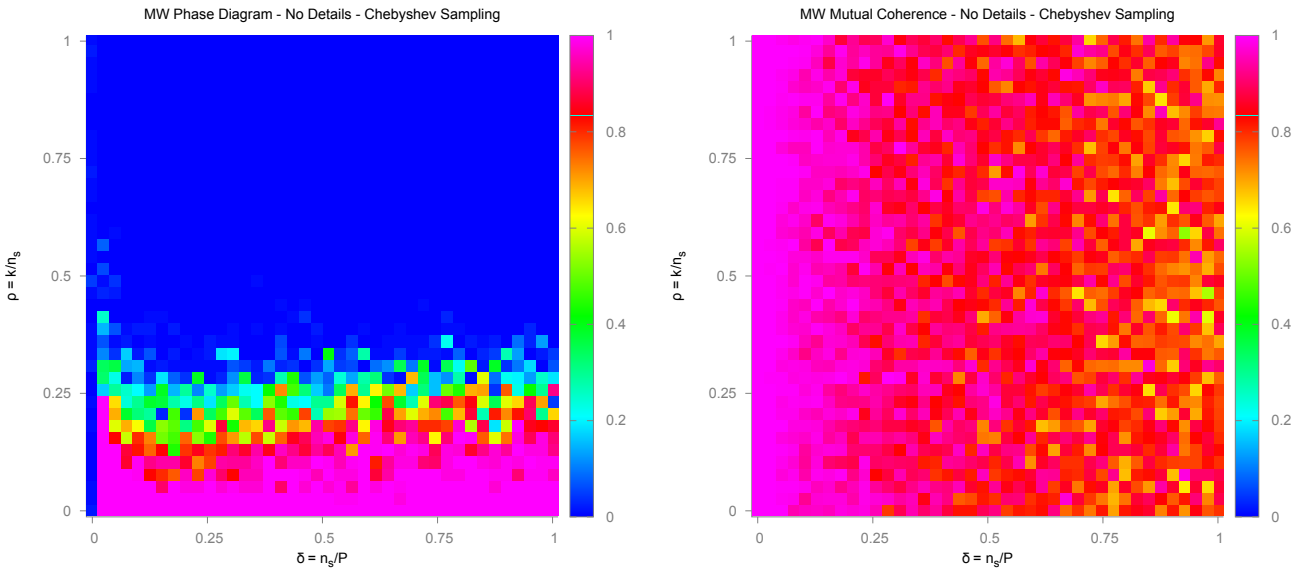


Figure 3.5: Phase Diagram for a Multiwavelet Measurement Matrix with no details - Chebyshev Sampling

3.4.2 Importance Sampling

Importance Sampling is a well known variance reduction methodology in Monte Carlo estimations. Sampling is performed according to a modified distribution, which promotes the important regions of the input variables and the quantity of interest whose expectation is sought.

An insight on the typical wavelet structure of piecewise smooth functions is given in [40]. In particular, the wavelet coefficients of piecewise smooth functions tend to form connected subtrees within wavelet trees. Additionally, a large wavelet coefficient (in magnitude) generally indicates the presence of a local singularity or sharp gradient. The above considerations form the basis of our sampling strategy. The idea is to concentrate samples at locations where large multiwavelet coefficients are observed while preconditioning the basis to maintain orthogonality. The proposed importance sampling consists of a number of steps that are applied iteratively:

1. A multiwavelet approximation up to a given m and resolution j is obtained by solving (P_0).

2. The coefficients α_i are sorted in decreasing order, based on the quantity $|\alpha_i|/|\mathrm{supp}(\psi_i)|$, where $|\mathrm{supp}(\psi_i)|$ is the size of the support of ψ_i.

3. A sample is drawn in $\mathrm{supp}(\alpha_i)$ according to a uniform distribution only if $|\alpha_i| > \alpha_{tol}$ ($\alpha_{tol} = 1.0 \times 10^{-3}$ is used in the present study).

3.4.3 Preconditioning

Assuming y is uniformly distributed on $[0,1]$, i.e., $\rho(y) : [0,1] \rightarrow 1$, the direct application of the above modified sampling leads to measurement matrices $\mathbf{\Psi}$ with large mutual coherence $\mu(\mathbf{\Psi})$. This is because the multiwavelets are orthogonal with respect to the measure $\rho(y)$, i.e. $\int_0^1 \psi_i(y)\,\psi_j(y)\,dy = \delta_{ij}$. A correction, therefore, is needed to retain orthogonality for sufficiently large M. Let $\gamma(y) : [0,1] \rightarrow \mathbb{R}_{\geqslant 0}$ denote the density function according to which the (independent) modified samples $y^{(k)}$, $k = 1, \ldots, M$, are distributed and $\hat{\psi}_i(y) = \psi_i(y)/\sqrt{\gamma(y)}$ be the scaled

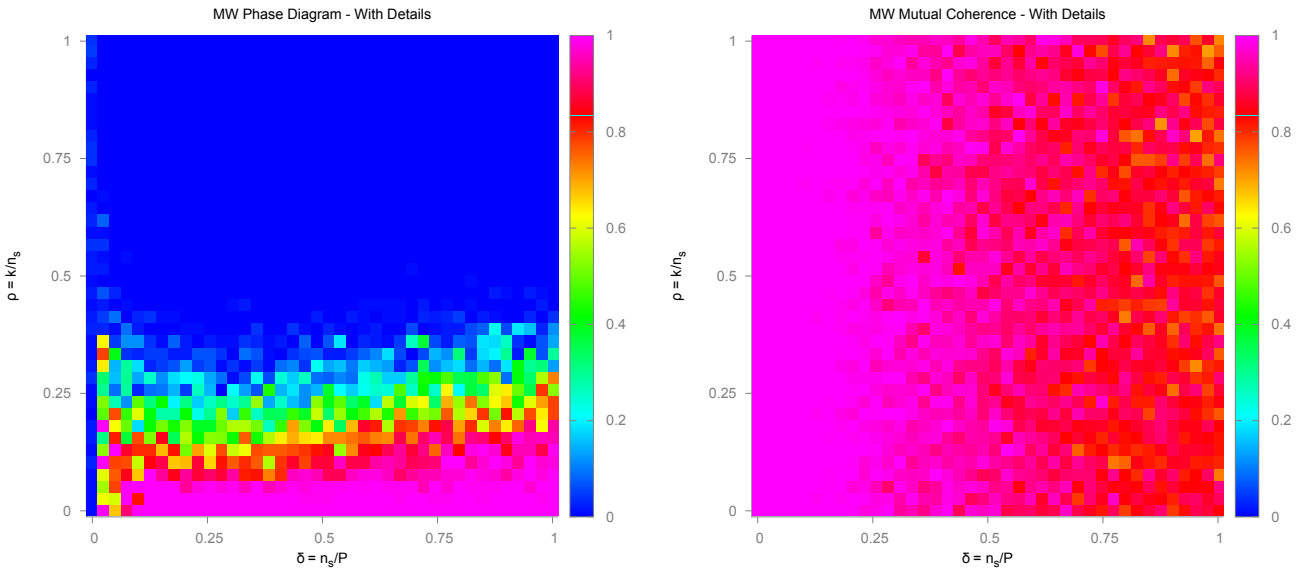
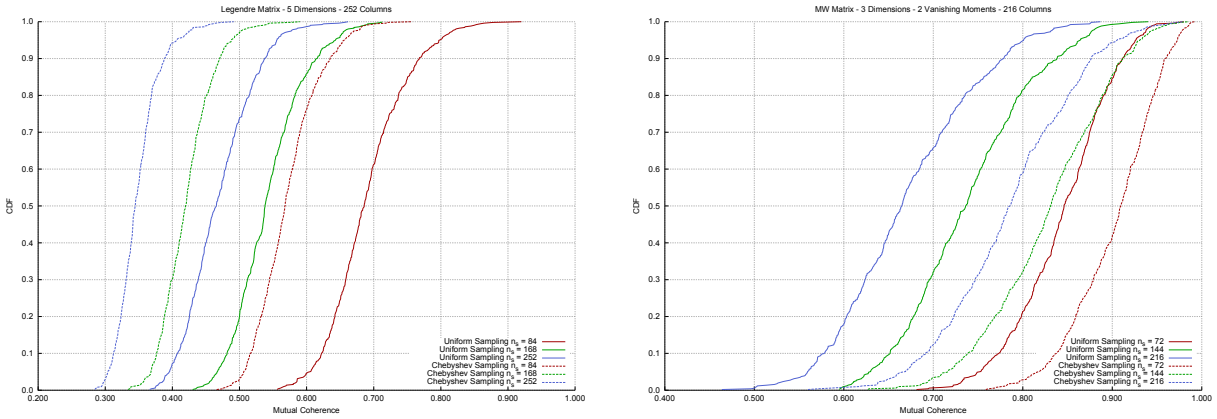


Figure 3.6: Phase Diagram for a Multiwavelet Measurement Matrix where first order details have been included - Uniform Sampling



(a) (b)

Figure 3.7: Mutual coherence distribution for random Legendre (a) and Multiwavelets (b) matrix ensambles

multiwavelet basis. Then,

$$\frac{1}{M}\sum_{k=1}^{M}\hat{\psi}_i(y^{(k)})\,\hat{\psi}_j(y^{(k)}) \xrightarrow{\text{a.s.}} \int_0^1 \frac{\psi_i(y)}{\sqrt{\gamma(y)}}\frac{\psi_j(y)}{\sqrt{\gamma(y)}}\,\gamma(y)\,dy = \delta_{ij}, \tag{3.5}$$

as a result of the strong law of large numbers.

In the CS framework, this translates in sampling according to $\gamma(y)$ and using a modified measurement matrix $\hat{\mathbf{\Psi}} = \mathbf{W}\mathbf{\Psi}$ and the data $\hat{\mathbf{u}} = \mathbf{W}\mathbf{u}$ with the *preconditioner* matrix $\mathbf{W} = \text{diag}(1/\sqrt{\gamma(y^{(i)})})$, $i = 1, \ldots, M$.

We now discuss how a piecewise constant measure $\gamma(y)$ can be defined on partitions of $[0, 1]$ associated with a (truncated) multiwavelet representation. We focus on establishing a one-to-one relationship between a scalar wavelet tree and a partition of $[0, 1]$. Vector trees (whose vertices are arrays of numbers) are used to store multiwavelet representations while scalar trees are usually adopted for wavelets. A partition of $[0, 1]$ is build by first forming a scalar connected subtree $\mathcal{T}$, obtained by pruning all vertices with coefficients α_i with $|\alpha_i| < \alpha_{tol}$. The *leaves* ($L$ in total) of $\mathcal{T}$

are identified, their supports form a set of disjoint intervals $\{\mathcal{B}_i : i = 1, \ldots, L\}$ which result in the desired partition of $[0, 1]$.

Using the coefficient-driven sampling discussed above, $\gamma(y)$ is defined as a piecewise constant distribution. In particular, a set of probability masses $p_i = 2^{-j}(M_i/M)$ for every box $\mathcal{B}_i$ is considered in which M_i is the number of samples within the interval $\mathcal{B}_i$ (with $|\mathcal{B}_i| = 2^{-j}$), j is the resolution level of the associated leaf, and M is the total number of available samples.

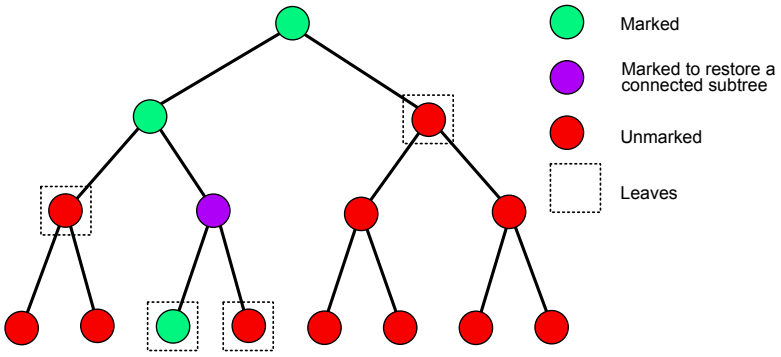


Figure 3.8: Identification of leaves on Multiwavelet tree

3.4.4 Numerical tests

To conclude this Chapter, we present two numerical examples. In the first example we consider two multiwavelet atoms with nested support and show the effect of importance sampling and preconditioning on their inner product. The second example, a sparse piecewise smooth signal of known tree representation is approximated by solving (P_0) with OMP, TOMP, uniform and Importance Sampling.

Multiwavelet Basis Product

In the present section, $\phi^{i_1}_{j_1,k_1}$ and $\phi^{i_2}_{j_2,k_2}$ are used to indicate two multiwavelet atoms with resolution j_1, j_2, translated by k_1, k_2 and with i_1, i_2 vanishing moments. In practice, we choose $j_1 = 0$, $k_1 = 0$, $i_1 = 2$ and $j_2 = 3$, $k_2 = 7$, $i_2 = 2$ respectively. A scalar tree with two leaves is associated to the resulting dictionary; therefore, $\mathcal{B}_1 = [0.0, 0.875)$ while $\mathcal{B}_2 = [0.875, 1.0]$. The samples are drawn proportionally to the two interval of the refinement, that is:

$$r_1 = |\mathcal{B}_2|/(|\mathcal{B}_1| + |\mathcal{B}_2|),\ r_2 = |\mathcal{B}_1|/(|\mathcal{B}_1| + |\mathcal{B}_2|),\ M_1 = \lfloor M,\ r_1 \rfloor,\ M_2 = M - M_1. \tag{3.6}$$

A preconditioner is derived by p_1 and p_2 which determine a piecewise constant measure over $[0, 1]$, as follows

$$|\mathcal{B}_1|\, p_1 + |\mathcal{B}_2|\, p_2 = 1,\ p_1/p_2 = M_1/M_2 = c,\ p_1 = c\, p_2,\ p_2 = 1/(|\mathcal{B}_1|\, c + |\mathcal{B}_2|). \tag{3.7}$$

Figure 3.9 shows the results in terms of inner product vs. number of samples. The basis product convergence faster to zero adopting importance sampling together with dictionary preconditioning. Note that preconditioning is essential to restore asymptotic orthogonality.

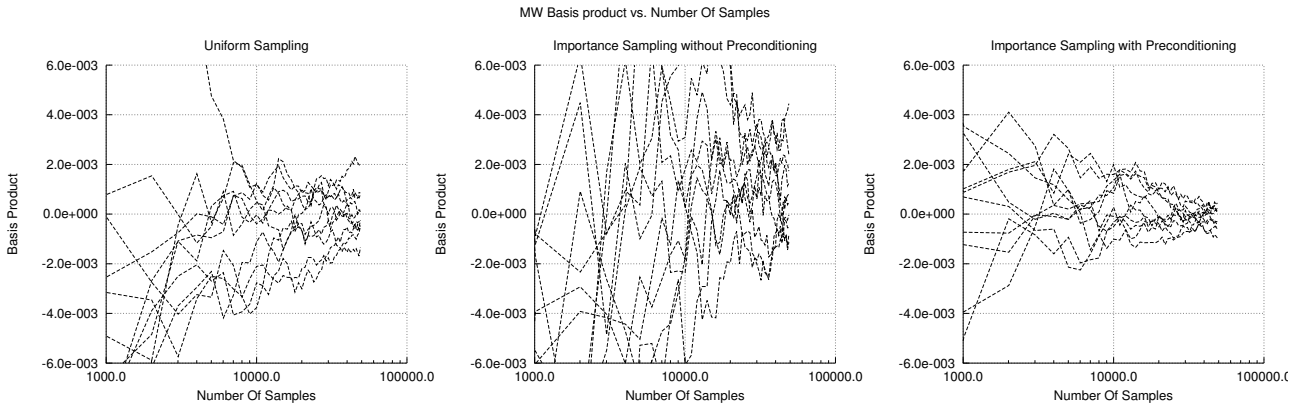


Figure 3.9: Basis product vs. Number of samples for Uniform and Importance Sampling

Reconstruction of sparse signals with known tree representation

As a second numerical experiment, we study the successful recovery rates for two piecewise smooth sparse signals as a result of adopting different sampling strategies and greedy heuristics (i.e. OMP, TOMP).

Before discussing our numerical experiments, we define the ℓ_p, ℓ_∞ norms we use to quantify the distance of discretely sampled signals.

$$\ell_p = \left(\frac{1}{M} \sum_{i=1}^{M} |u_{approx}(y^{(i)}) - u_{ex}(y^{(i)})|^p \right)^{1/p}, \quad \ell_\infty = \max_{i=0,\ldots,M} |u_{approx}(y^{(i)}) - u_{ex}(y^{(i)})| \tag{3.8}$$

We choose two functions, f_1 and f_2, as follows:

$$f_1 = \begin{cases} \sin(40y) & \text{if} \quad y < 0.25 \\ 10\sin(15y) & \text{otherwise} \end{cases} \qquad f_2 = \begin{cases} 1 & \text{if} \quad y < 0.75 \\ 10\sin(15y) & \text{otherwise} \end{cases} \tag{3.9}$$

A one dimensional multiwavelet basis with $j \in \{0, \ldots, 7\}$ and $m = 2$ (2 vanishing moments) is also selected, resulting in a dictionary with cardinality equal to 768.

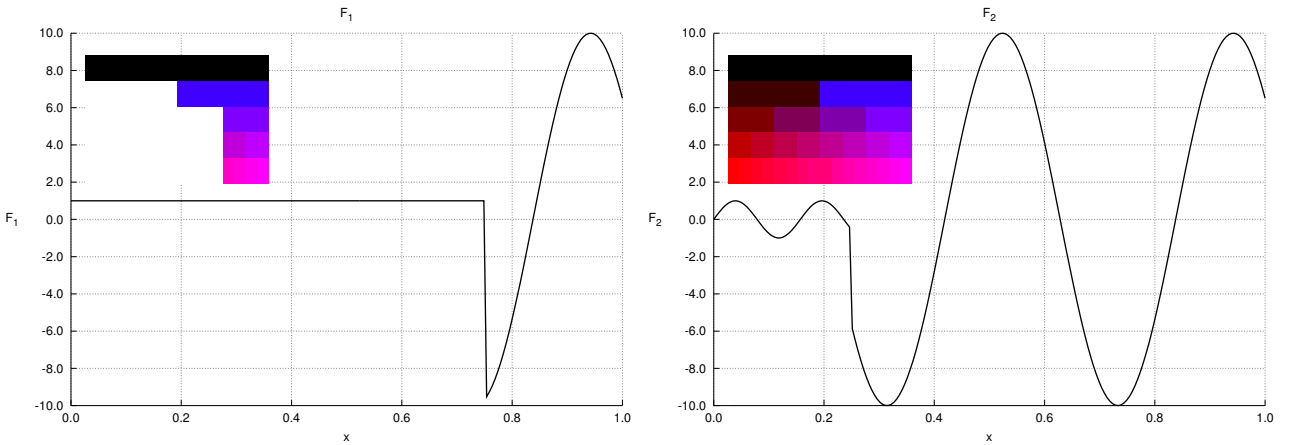


Figure 3.10: Adopted Piecewise smooth signal for reconstruction tests

Figure 3.10 shows the selected signals together with the associated envelope scalar trees. Note that both signals exhibit a connected scalar tree representation.

Tests are performed as follows: an increasing number of samples is drawn from 10 to 694 with increments of 36 samples. They are placed uniformly at random or using an Importance Sampling approach as highlighted in Section . Reconstructions are performed using OMP and TOMP to solve $(P_{0,\epsilon})$ 500 times. The relative ℓ_2 errors on 1000 randomly samples points is plotted in Figure 3.11 against the number of samples.

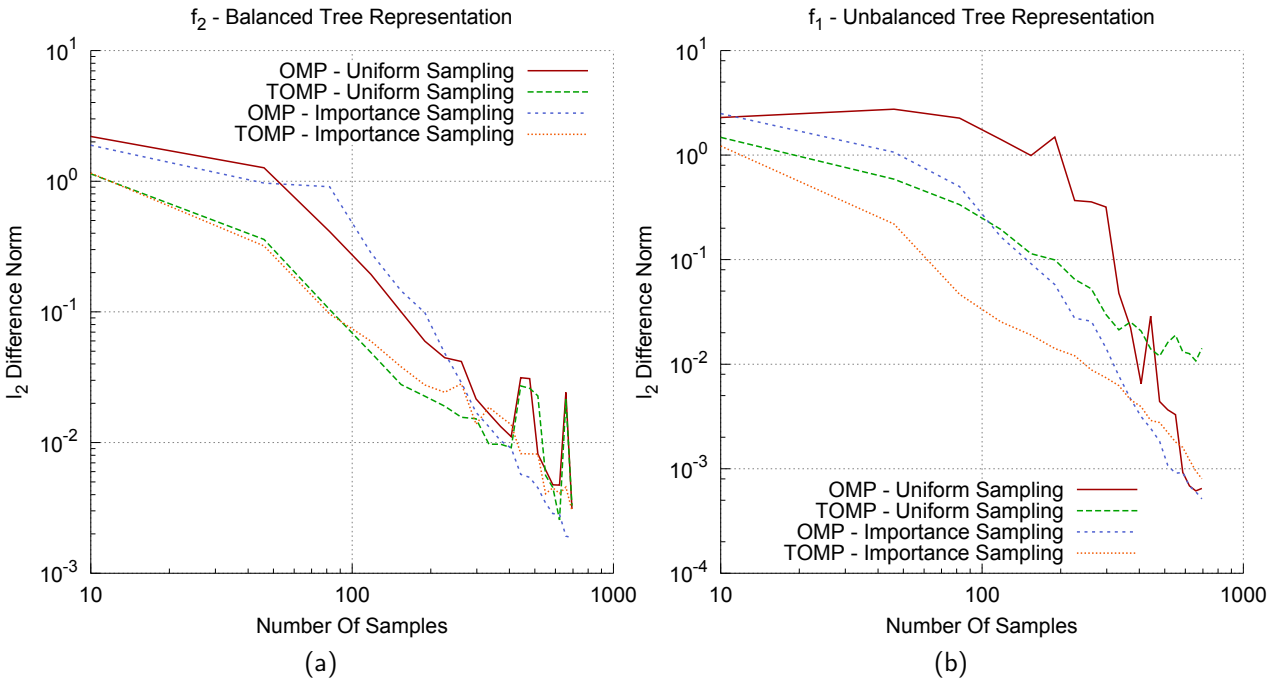


Figure 3.11: Relative reconstruction errors vs. number of samples at 1000 random locations.

Different choices of ϵ are necessary to compare optimal performances of OMP and TOMP. Figure 3.12 shows the ℓ_2 residual norm versus index set cardinality produced by the OMP and TOMP algorithm, respectively. A fixed tolerance on the residual norm clearly translates in two different cardinalities. OMP performs a careful selection of one support location per iteration while a larger connected subtree index set is used by TOMP.

To establish a fair comparison between the two algorithms, we employ cross validation, i.e., we run every reconstruction using 11 noise levels from 10^{-1} to 10^{-5} and using the results which produce the minimum residual over a subset of the total samples. In practice, we maintain a training/testing ratio of $3/4$ throughout the simulation.

From Figure 3.11, it can be seen how TOMP and Importance Sampling result in the best performance in particular when a limited number of samples are used. For an increasing number of samples OMP eventually provides the best relative reconstruction errors.

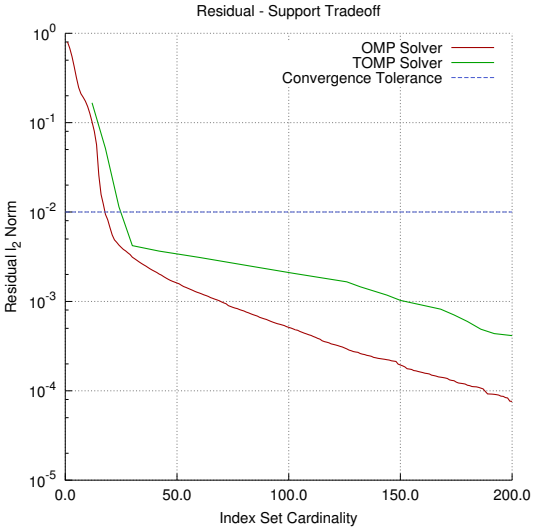


Figure 3.12: Residual vs. index set cardinality for successive iterations of OMP and TOMP, respectively.

Chapter 4

Benchmarks and Applications

4.1 Remarks on implementations of CS-MW UQ

In the present Section, all previous discussions on approximation dictionary, sparse reconstruction algorithms and sampling strategies are assembled into a framework for uncertainty propagation. As

Algorithm 3 CSMW algorithm

Inputs:
The number vanishing moments m.
The maximum resolution level j_{max}.
The maximum number of samples together with their increment $M, \Delta M$.
Algorithm:
InitMWTree() ▷ Initialize Multiwavelet tree structure.
for all $l = 1 \rightarrow M$ **do**
  AddSamples() ▷ Add samples uniformly or based on Importance Sampling.
  EvalModelResponse() ▷ Perform deterministic simulations.
  BuildMWMatrix() ▷ Build MW Measurement Matrix.
  if Importance Sampling **then**
    AssembleCompactCoeffVector() ▷ Shrink Coefficient Vector to envelope scalar wavelet.
    RestoreCompactTreeConnectivity() ▷ Restore the connectivity in the envelope scalar tree.
    BuildHyperCubeRefinement() ▷ Build Hypercube Refinement.
    AddSamplesToRefinement() ▷ Add Samples To refinements.
    BuildSamplePreconditioner() ▷ Build Preconditioner.
    ApplyPreconditioner() ▷ Apply preconditioner to Matrix and RHS.
  end if
  SolveSparseRecovery() ▷ Solve with OMP, TOMP etc.
  EvalStatistics() ▷ Evaluate statistics from expansion coefficients.
  CheckConvergence() ▷ Evaluate convergence to global statistics.
end for

presented in algorithm 3, adaptivity is considered only from a sampling perspective. In other words, it is used as a strategy to place samples at locations where important features are expected for the response at increasingly finer scales. Alternatively, approximants might also be parameterized by the number of vanishing moments, affecting the cardinality of the Legendre *father* wavelet family. Moreover, an adaptive procedure based on local resolution increments could also be also conceived.

For practical applications it is also important to provide an error estimation methodology, allowing the iterative procedure to be stopped when a sufficiently close approximation of the stochastic response is reached. Methodologies based on cross validation look promising in this regard, where reconstructions are performed over a subset of the available samples while the remaining ones are used for error evaluation.

Remark 3 (Multi-Element Generalization)**.** A straightforward generalization of the proposed framework can be obtained by subdividing the unitary hypercube in smaller elements, using the CS-MW

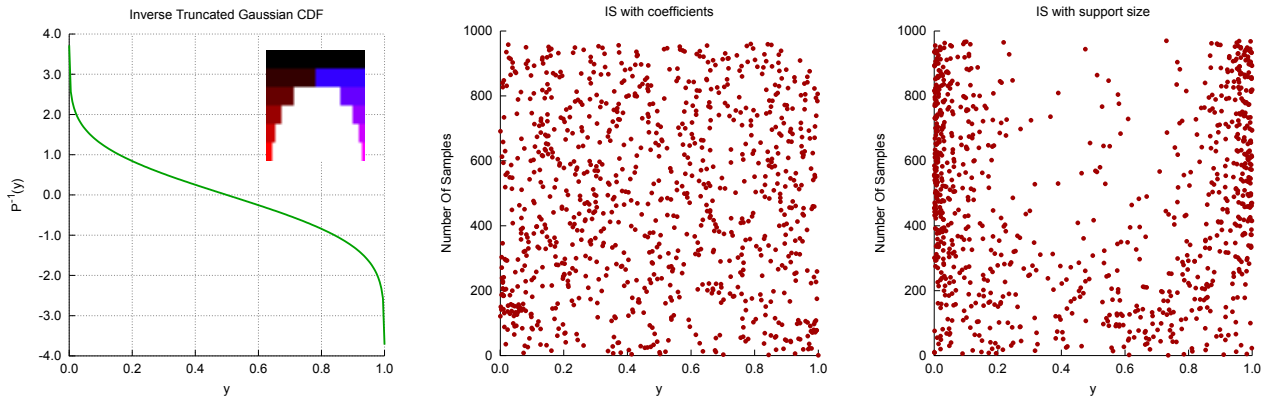
methodology within each subdomain. Two significant advantages will immediately results: an uncoupled propagation step will result for all subdomains and local approximations will promote sparsity. As a consequence, parallel implementations could produce significant speed ups, and recovery procedures will be more effective in producing accurate representations.

4.2 Transformation to Gaussian measure

We use CSMW to evaluate the statistics of a r.v. $\hat{y}$ associated to standard normal distribution. As multiwavelets form an orthonormal dictionary in $\mathbf{L}^2(\mathbb{R})$ respect to the uniform measure, we need to transform $\hat{y}$ to a variable which is $\mathcal{U}([0,1])$. We therefore project it onto the standard normal cumulative distribution. If $\hat{y} \in (-\infty, +\infty)$, then its cumulative distribution $P(\hat{y} \leqslant \hat{y}_i)$ is defined as $P(\hat{y}) : \mathbb{R} \to [0,1]$ which provides the sought transformation. We have:

$$\hat{y} = P^{-1}(y), \quad \mu(\hat{y}) = \int_0^1 \hat{y}\, 1\, dy = \int_0^1 P^{-1}(y)\, 1\, dy. \tag{4.1}$$

A graphical representation of the function to integrate is illustrated in Figure 4.1a. As a result of the transformation from an infinite to a finite support, we have that $P^{-1}(y) \to -\infty$ for $y \to 0$ and $P^{-1}(y) \to +\infty$ for $y \to 1$. Asymptotes in $\{0,1\}$ are difficult to interpolate for modest resolutions and convergence to the final statistics is consequently slowed down.



(a) (b) (c)

Figure 4.1: Inverse cumulative mapping with tree represetation (a). The distribution of samples is also shown, generated by Importance Sampling using only coefficient magnitudes (b) and divided by the support size (c)

A straightforward remedy which is naturally applied in practical applications, is to perform a truncation of the standard normal distribution. A truncation level is selected in our case which corresponds to a total probability of 2.0×10^{-4}.

Convergence profiles are shown in figure 4.2; faster convergence is obtained for the CSMW approach respect to the MCS and LHMCS methods. It is also shown that convergence can be further improved by increasing the maximum resolution level of the multiwavelet approximant.

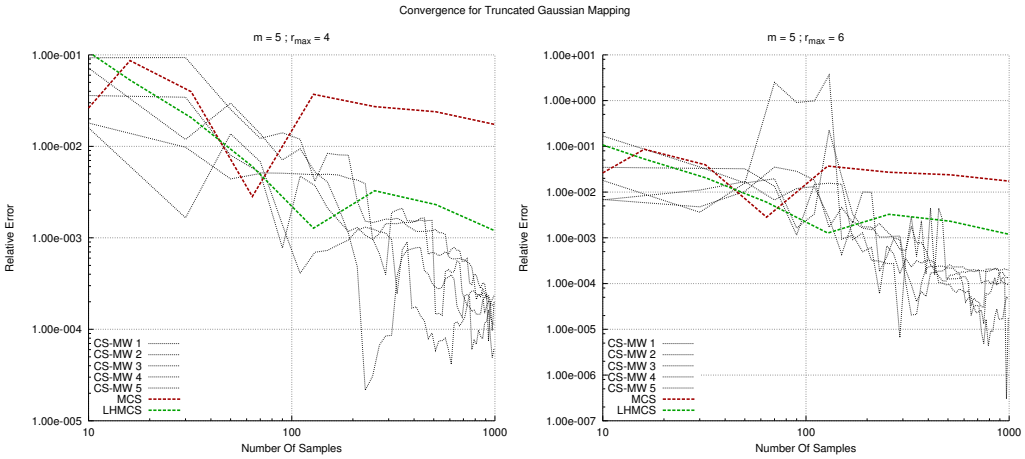


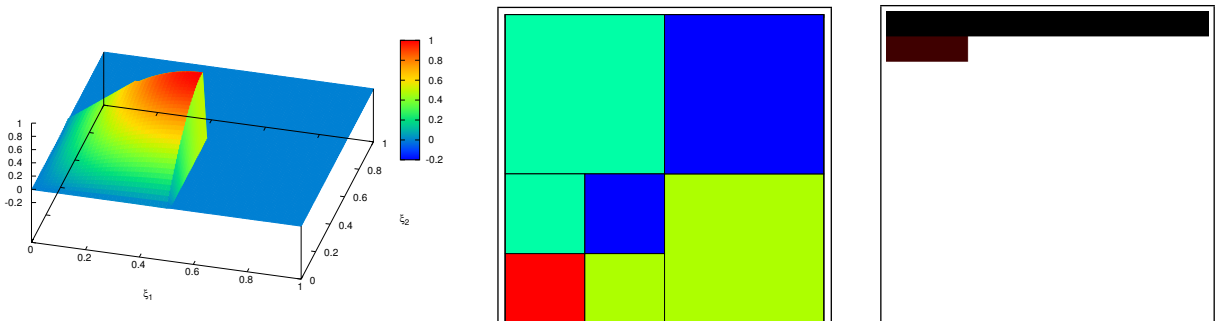
Figure 4.2: Convergence for Truncated Gaussian Mapping

4.3 Non smooth approximation from Agarwal et al.

A non smooth function is prosed in [2] as follows:

$$u_{ex}(\xi_1, \xi_2) = \begin{cases} 0, & \text{if} \quad \xi_1 > \alpha_1, \xi_2 > \alpha_2 \\ sin(\pi\xi_1)sin(\pi\xi_2), & \text{otherwise} \end{cases} \tag{4.2}$$

with $\alpha_1 = 0.5$ $\alpha_2 = 0.5$. A graphical representation of the function is shown in figure 4.3a. Its support is only defined in the third quadrant of the unitary plane $[0, 1]^2$. As a result, global approximation methods offer poor approximations due to the discontinuities located at $y_1 = 1/2$ and $y_2 = 1/2$. Note that it is instead particularly well suited for our multiresolution framework giving rise to a sparse expansion in the employed dictionary, as shown by the envelope scalar multiwavelet tree in figure 4.3c. A 2D representation of the scalar wavelet tree leaves and associated coefficients is depicted in figure 4.3b.



(a) (b) (c)

Figure 4.3: Probability of successful reconstruction (a) and mutual coherence distribution (b) for random Legendre matrix

Figure 4.4 shows the convergence profiles in terms of ℓ_1, ℓ_2, ℓ_∞ norms based on 10^3 random locations. Convergence rates are compared for two independent runs of the OMP and TOMP solvers with Importance Sampling. It is clear how TOMP performs better than OMP in this case requiring nearly half of the samples.

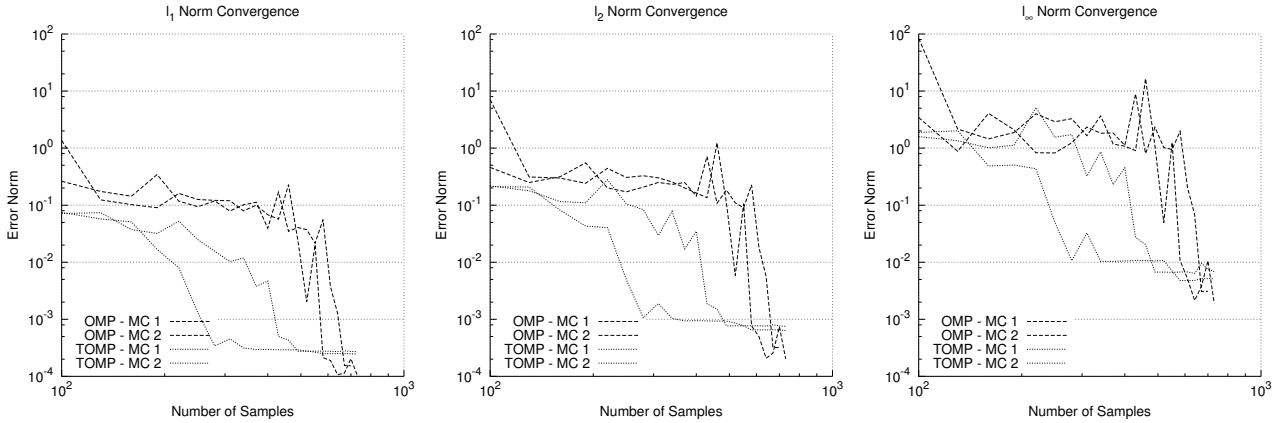


Figure 4.4: Convergence ℓ_1, ℓ_2, ℓ_∞ norms for non smooth function from Agarwal et al.

4.4 Kraichnan-Orszag (K-O) Problem

The Kraichnan-Orszag (KO) problem is derived from simplified inviscid Navier-Stokes equations [58], and is expressed as a coupled system of non-linear ODEs. We here adopt a rotated version of the original KO problem

$$\frac{du_1}{dt} = u_1\, u_3, \quad \frac{du_2}{dt} = -u_2\, u_3, \quad \frac{du_3}{dt} = -u_1^2 + u_2^2, \tag{4.3}$$

with initial conditions specified below.

In [93], the KO problem is used as a benchmark and analytical solutions are provided in terms of Jacobi's elliptic functions. If the set of initial conditions is chosen such that the bifurcation point $(u_1, u_2, u_3) = (\sqrt{2}, 0, 1)$ is consistently crossed, it is shown that the accuracy of the global polynomial approximations (at the stochastic level) deteriorates rapidly with time.

4.4.1 Results for 1D KO Problem.

Initial conditions for (4.3) are assumed to be uncertain and specified as

$$u_1(t=0) = 1 \quad ; \quad u_2(t=0) = 0.2\, y - 0.1 \quad ; \quad u_3(t=0) = 0, \tag{4.4}$$

where y is uniformly distributed on $[0, 1]$. The stochastic response is evaluated at $t = 20s$ and $t = 30s$ using a multiwavelet dictionary with $m = 3$ and a resolution up to $j = 7$. The OMP solver was used with a relative tolerance $\epsilon = 1.0 \times 10^{-4}$. The time history of the standard deviation for variable u_1 together with the reconstructed response at $t = 30s$ and convergence graphs are illustrated in Figure ??. The error metric $\epsilon_{rel} = |\sigma_{CSMW} - \hat{\sigma}|/\hat{\sigma}$ is also evaluated where σ_{CSMW} is the estimate for the standard deviation calculated with the CS-based multiwavelet expansion and $\hat{\sigma}$ the corresponding exact value.

4.4.2 Results for 2D KO Problem at $t = 10s$.

The initial conditions of the Kraichnan-Orszag problem are again assumed to be uncertain but this time are functions of two random variables

$$u_1(t=0) = 1 \quad ; \quad u_2(t=0) = 0.2\, y_1 - 0.1 \quad ; \quad u_3(t=0) = 2\, y_2 - 1, \tag{4.5}$$

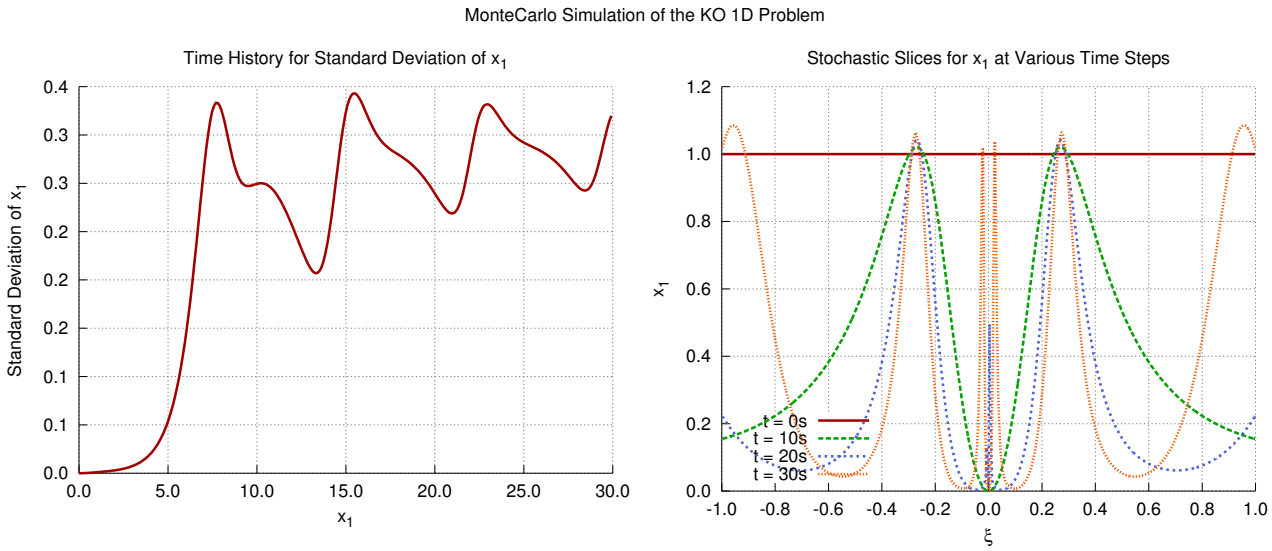


Figure 4.5: Time history for $\sigma_{\hat{x}_1}$ and stochastic slices of $\hat{x}_1$ at different simulation times

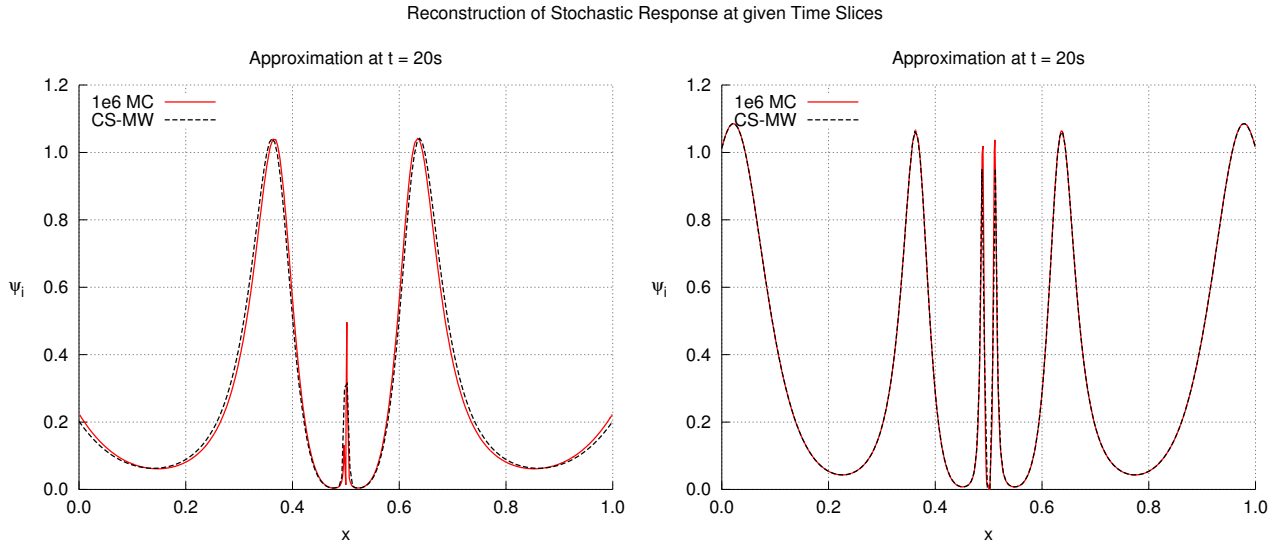


Figure 4.6: Stochastic Response Reconstruction for the 1D KO Problem

where y_1 and y_2 are independent and uniformly distributed on $[0, 1]$. A two dimensional multiwavelet measurement matrix is generated with $m = 2$ and resolution up to $j = 4$, resulting in a basis of cardinality $P = 9216$. Figure 4.8 shows results in terms of sampling distribution and multiwavelet coefficients. Convergence to standard deviation of the system's response is also shown in Figure 4.10. The expansion coefficients produced by the proposed strategy with about $M = 2400$ samples is comparable to those obtained using a multiwavelet least squares approximation (where coefficients are evaluated as $\boldsymbol{\alpha}_{LS} = (\boldsymbol{\Psi}^T\boldsymbol{\Psi})^{-1}\boldsymbol{\Psi}^T\mathbf{u}$ with 9×10^4 samples, demonstrating the efficiency of the CS-based reconstruction.

4.5 Application: Passive vibration control under uncertainty using TMD devices

Vibrations produced by harmonic or stochastic excitations may produce excessive acceleration levels in structures with significant impact on serviceability. Periodic vibrations might be produced in a

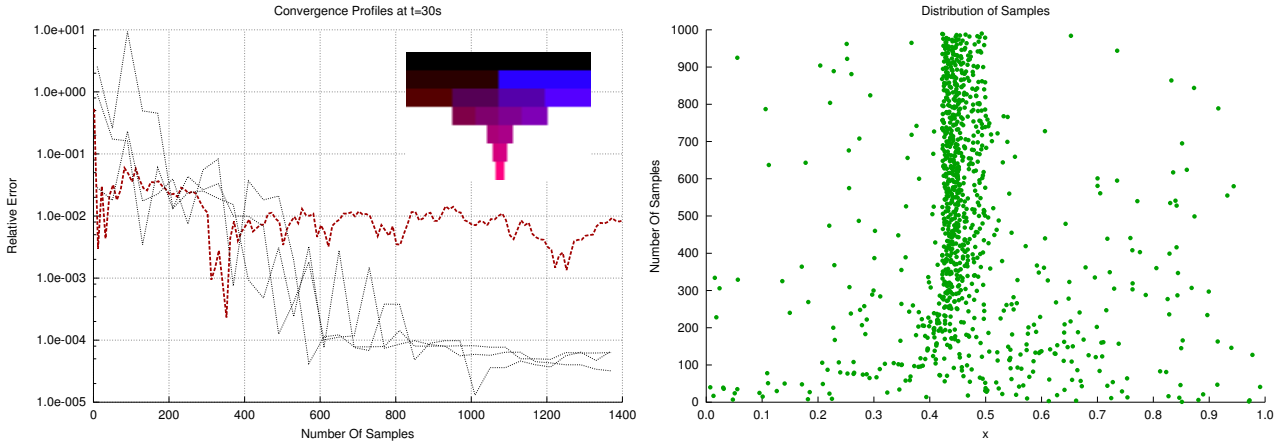


Figure 4.7: MW-CS Convergence and Sampling Set

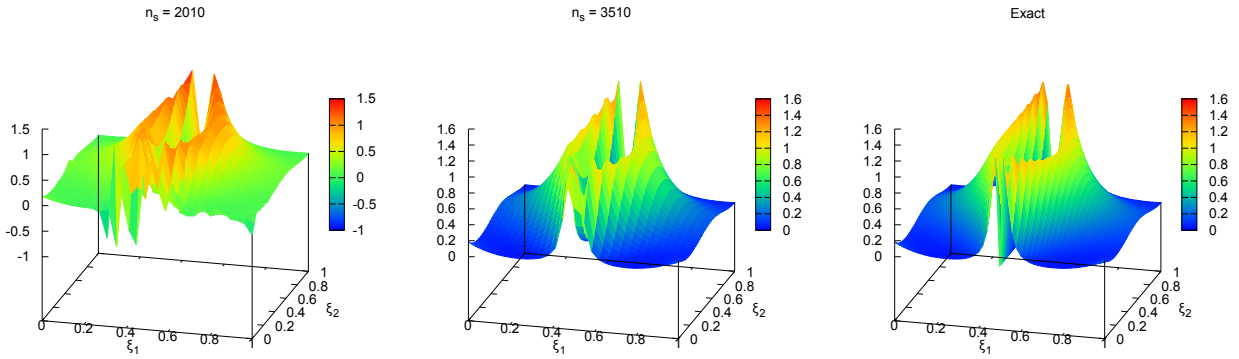


Figure 4.8: Stochastic response reconstruction for progressively increasing number of samples.

floor structure by walking of the occupants while the action of the wind on high rise buildings offers a typical example of stochastic excitation. If resonance occur, the effects of the latter forces might be significantly amplified; the available system damping plays, in this case, a crucial role for the response of the system. Excessive accelerations are perceived by humans as loss of comfort; levels higher than 0.5% of g (the gravitational acceleration) might be perceived by the occupants of a given structural system, while 5% of g can be considered an upper bound for serviceability related to human perception. Passive vibration control may provide a cost effective remedy against excessive vibration levels in structures mainly due to the absence of expensive active control systems and the minimal required maintenance. Tuned Mass Dampers (TMD) devices are one of the typical choices for vibration reduction. Their introduction follows from a relatively simple observation applied to a 2 degrees of freedom (d.o.f.) spring-mass system: the steady state undamped response of the principal mass subject to an harmonic excitation can be minimized by applying a TMD device tuned to the same frequency (see Section 4.5.1). The efficiency of the installed TMD can be defined, in this case, based on the reduction achieved in the peak acceleration response. Perfect (infinite) efficiency is achieved in the theoretical case. However, practical efficiency of TMDs is limited by the following factors:

- Real loads can be characterized by broad frequency spectra and multiple spatial components.
- External excitations might have limited time duration such that a steady state might be difficult to reach in practice.
- Real structures as well as TMD devices always exhibit damping which might be difficult to measure in practice. It is in fact well known that natural frequencies might be affected by

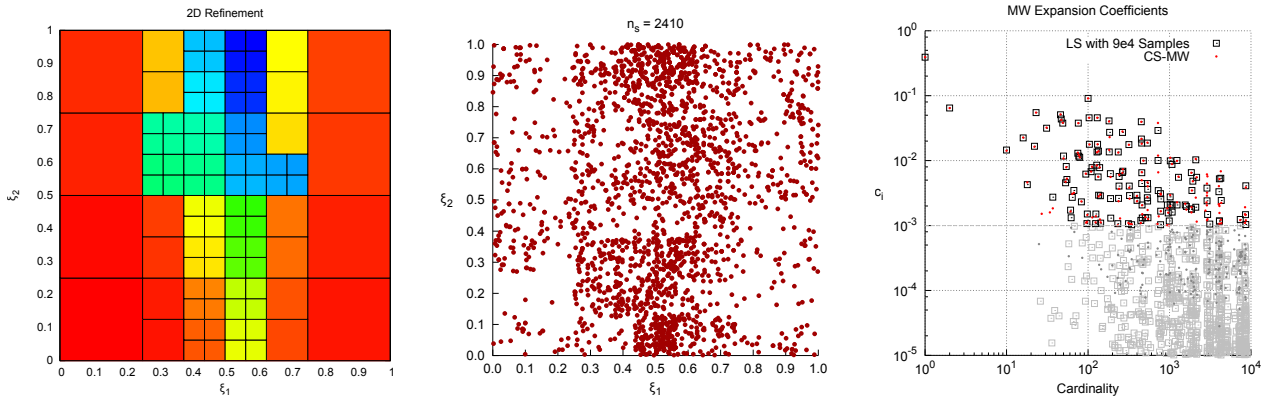


Figure 4.9: Refinements (a), Samples (b) and MW expansion coefficients for K-O 2D.

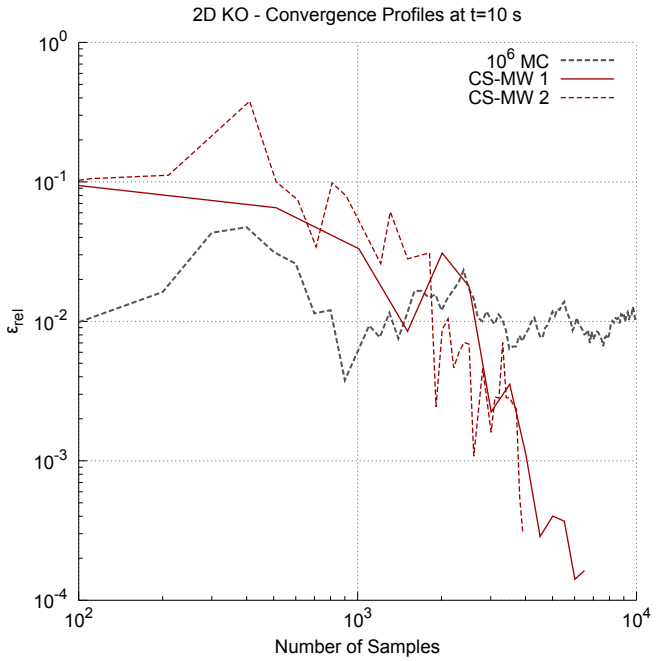


Figure 4.10: Convergence profiles to second order statistics.

damping, even if modest differences are usually observed for typical elastic damping. Furthermore, suspended or cable stayed structure might experience a modification of their stiffness as a result of relaxation phenomena. As a result, tuning of TMD devices might be difficult to achieve or may deteriorate with time.

- Finally, real structures are continuous systems characterized by an infinite frequency content. Therefore, a TMD device designed for a particular frequency could be, in general, ineffective to prevent vibration for a number of different modes.

Various factors affect the performance of TMD devices. In the present study, we deal with these factors as *uncertainties* and consider a parametrization in probability. Our objective is to use uncertainty propagation methodologies to efficiently and systematically derive statistical surrogates for better assessing TMD efficiency.

4.5.1 Two dof systems with passive vibration control

A simple 2 dof system is represented in Figure 4.11.

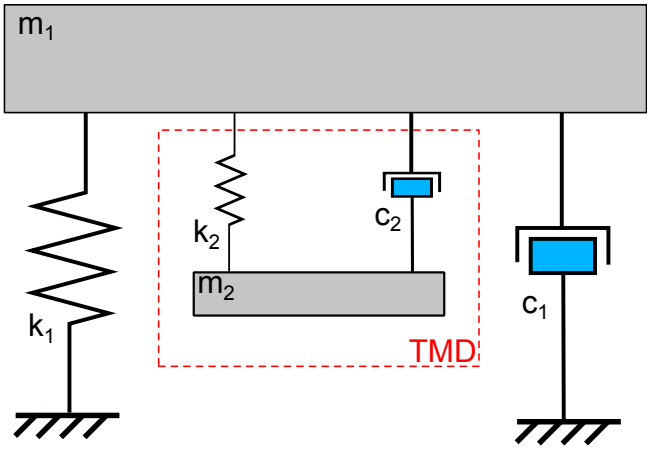


Figure 4.11: Schematic representation of a two dof dynamical system characterized by a principal system ("1") and an attached TMD device ("2").

The motion of the system can be completely characterized by the two triplets $(\ddot{x}_1(t), \dot{x}_1(t), x_1(t))$ and $(\ddot{x}_2(t), \dot{x}_2(t), x_2(t))$ providing the evolution in time of the principal and TMD mass, respectively. Assuming a linear elastic material and that small oscillations are produced in the system, a linear system of ODEs can be written, as follows:

$$\boldsymbol{M}\ddot{\boldsymbol{x}} + \boldsymbol{C}\dot{\boldsymbol{x}} + \boldsymbol{K}\boldsymbol{x} = \boldsymbol{F}(t) \tag{4.6}$$

where:

$$\ddot{\boldsymbol{x}}^T = [\ddot{x}_1, \ddot{x}_2], \quad \dot{\boldsymbol{x}}^T = [\dot{x}_1, \dot{x}_2], \quad \boldsymbol{x}^T = [x_1, x_2], \quad \boldsymbol{F}(t)^T = [F_1(t), F_2(t)] \tag{4.7}$$

and:

$$\boldsymbol{M} = \begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix}, \boldsymbol{C} = \begin{bmatrix} c_1 + c_2 & -c_2 \\ -c_2 & c_2 \end{bmatrix}, \boldsymbol{K} = \begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix}. \tag{4.8}$$

A characterization of damping in terms of *damping ratios*, ξ_1 and ξ_2, is usually more convenient. The following expressions can therefore be used:

$$c_1 = \xi_1 c_{1,cr} = \xi_1 \, 2\sqrt{m_1 \, k_1} \quad c_2 = \xi_2 c_{2,cr} = \xi_2 \, 2\sqrt{m_2 \, k_2} \tag{4.9}$$

where typical values of the structural system damping are $\xi_1 = 0.01 \div 0.05$, depending on construction material, type of beam-column and beam-beam connection as well as resistance mechanism.

4.5.2 Numerical solution for the 2 dof system with TMD

The *Newmark beta method* is employed to numerically solve system (4.6). If we consider a generic time interval $\mathcal{I}_n = [n\Delta t, (n+1)\Delta t]$ with $n \in \mathbb{N}, n \geqslant 0$ and Δt a selected time step, it assumes a linear variation of the acceleration in $\mathcal{I}_n$, as follows:

$$\ddot{\boldsymbol{x}}_\beta^{(n,n+1)} = (1-\beta)\,\ddot{\boldsymbol{x}}^{(n)} + \beta\,\ddot{\boldsymbol{x}}^{(n+1)}. \tag{4.10}$$

The following expression are derived for the velocity and displacement vectors at instant $(n+1)$:

$$\begin{aligned} \dot{\boldsymbol{x}}^{(n+1)} &= \dot{\boldsymbol{x}}^{(n)} + \Delta t\,\ddot{\boldsymbol{x}}_\beta^{(n,n+1)} = \dot{\boldsymbol{x}}^{(n)} + \Delta t\,(1-\beta)\,\ddot{\boldsymbol{x}}^{(n)} + \Delta t\,\beta\,\ddot{\boldsymbol{x}}^{(n+1)} \\ \boldsymbol{x}^{(n+1)} &= \boldsymbol{x}^{(n)} + \Delta t\,\dot{\boldsymbol{x}}^{(n)} + \Delta t^2\left[\left(\frac{1}{2} - \alpha\right)\ddot{\boldsymbol{x}}^{(n)} + \alpha\,\ddot{\boldsymbol{x}}^{(n+1)}\right]. \end{aligned} \tag{4.11}$$

The values of acceleration and velocity at $(n+1)$ can also be obtained in terms of quantities evaluated at instant (n) and only the displacement at $(n+1)$ as follows:

$$\begin{aligned}\ddot{\boldsymbol{x}}^{(n+1)} &= \frac{1}{\alpha\,\Delta t^2}\,\boldsymbol{x}^{(n+1)} - \frac{1}{\alpha\,\Delta t^2}\,\boldsymbol{x}^{(n)} - \frac{1}{\alpha\,\Delta t}\,\dot{\boldsymbol{x}}^{(n)} - \left(\frac{1}{2\,\alpha} - 1\right)\ddot{\boldsymbol{x}}^{(n)} \\ \dot{\boldsymbol{x}}^{(n+1)} &= \frac{\beta}{\alpha\,\Delta t}\,\boldsymbol{x}^{(n+1)} - \frac{\beta}{\alpha\,\Delta t}\,\boldsymbol{x}^{(n)} + \left(1 - \frac{\beta}{\alpha}\right)\dot{\boldsymbol{x}}^{(n)} + \Delta t\left(1 - \frac{\beta}{2\,\alpha}\right)\ddot{\boldsymbol{x}}^{(n)}.\end{aligned} \tag{4.12}$$

Finally, the equilibrium equation is enforced at instant $(n+1)$ (*implicit* time integration):

$$\boldsymbol{M}\ddot{\boldsymbol{x}}^{(n+1)} + \boldsymbol{C}\dot{\boldsymbol{x}}^{(n+1)} + \boldsymbol{K}\boldsymbol{x}^{(n+1)} = \boldsymbol{F}^{(n+1)}. \tag{4.13}$$

A dynamic right-hand side is defined with all the terms at instant (n):

$$\begin{aligned}\hat{\boldsymbol{F}}^{(n+1)} = \boldsymbol{F}^{(n+1)} &+ \boldsymbol{M}\left[\frac{1}{\alpha\,\Delta t^2}\,\boldsymbol{x}^{(n)} + \frac{1}{\alpha\,\Delta t}\,\dot{\boldsymbol{x}}^{(n)} + \left(\frac{1}{2\,\alpha} - 1\right)\ddot{\boldsymbol{x}}^{(n)}\right] + \\ &+ \boldsymbol{C}\left[\frac{\beta}{\alpha\,\Delta t}\,\boldsymbol{x}^{(n)} - \left(1 - \frac{\beta}{\alpha}\right)\dot{\boldsymbol{x}}^{(n)} - \Delta t\left(1 - \frac{\beta}{2\,\alpha}\right)\ddot{\boldsymbol{x}}^{(n)}\right],\end{aligned} \tag{4.14}$$

leading to the final algebraic system:

$$\left[\frac{1}{\alpha\,\Delta t^2}\,\boldsymbol{M} + \frac{\beta}{\alpha\,\Delta t}\,\boldsymbol{C} + \boldsymbol{K}\right]\boldsymbol{x}^{(n+1)} = \hat{\boldsymbol{F}}^{(n+1)} \tag{4.15}$$

Typical values for (α, β) are $(0.25, 0.5)$; they minimize numerical damping for the Newmark scheme.

A preliminary numerical simulation is performed to show how a TMD usually produces an attenuation in the response of the host system. Note that the international unit system (SI) is used throughout. The following parameter set is adopted:

$$\begin{aligned}&m_1 = 20.0\,kg, \quad \xi_1 = 1\,\%, \quad k_1 = 19739.2\,N/m, \quad f_1 = 5.0\,Hz, \\ &m_2 = 1.0\,kg, \quad \xi_2 = 5\,\%, \quad k_2 = 986.96\,N/m, \quad f_2 = 5.0\,Hz, \\ &\Delta t = 1.0 \times 10^{-3}\,s, \quad T_{tot} = 4.0\,s,\end{aligned} \tag{4.16}$$

with the following initial conditions:

$$\boldsymbol{x}(t=0)^T = [0,0], \quad \dot{\boldsymbol{x}}(t=0)^T = [0,0], \quad \ddot{\boldsymbol{x}}(t=0)^T = [0,0]. \tag{4.17}$$

A step-shaped load is applied as follows:

$$\begin{cases}\boldsymbol{F}(t)^T = [0,0], & \text{if} \quad t < t^* \\ \boldsymbol{F}(t)^T = [F^*,0], & \text{if} \quad t \geqslant t^*\end{cases} \tag{4.18}$$

with $t^* = 0.5\,s$ and $F^* = 100\,N$

The results in terms of displacements, velocities and accelerations of both principal and TMD systems are illustrated in Figure 4.13. A reduction is observed in the response of the system with the TMD device installed, as expected.

4.5.3 Typical efficiency of TMD passive vibration control devices

A better understanding of the attenuation mechanism of TMD devices can be captured by looking at the frequency domain. Consider the same dynamical system highlighted in the previous Section

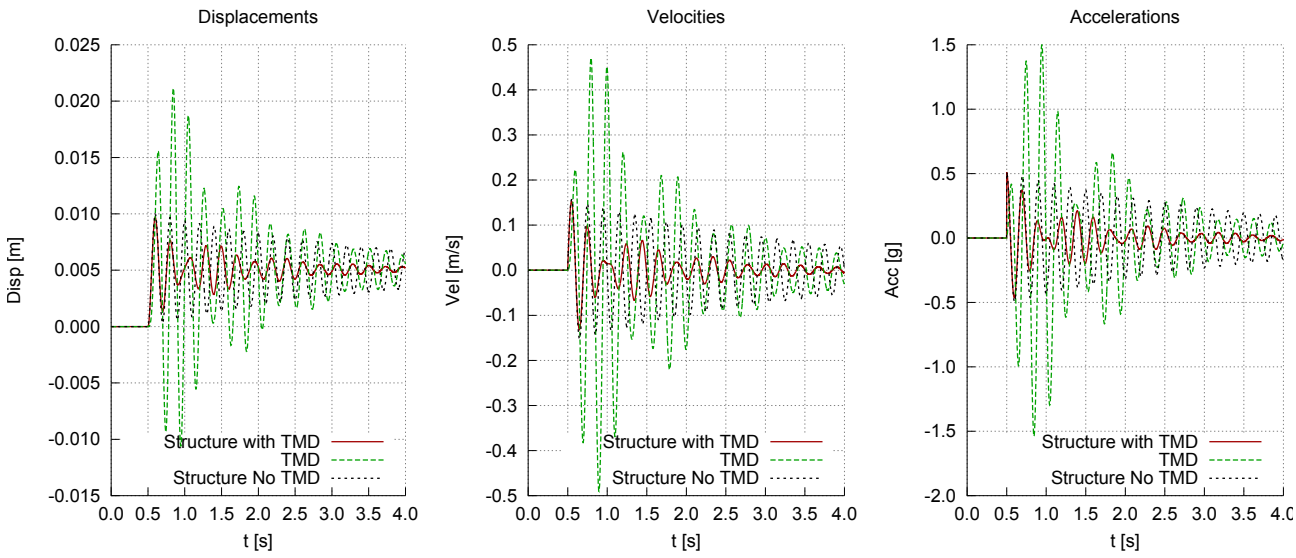


Figure 4.12: Results of a transient dynamic simulation showing reduction in the principal system response after installation of a TMD device.

where the integration in time has been extended to $T_{tot} = 10.0\,s$. A family of harmonic external excitations is considered here, as a function of frequency f.

$$\boldsymbol{F}(t)^T = [F^* sin(2\pi f), 0], \quad f = 4.0 \div 0.6\,Hz \tag{4.19}$$

A graph of the maximum acceleration in the principal system versus the external excitation frequency is depicted in figure 4.13 for the following configurations:

- Undamped principal system with no TMD device installed.
- Undamped principal system with undamped TMD device installed.
- 1% damped principal system with undamped TMD device installed.
- 1% damped principal system with 10% TMD device installed.

A single-peak infinite acceleration response typical of a resonant sdof system (with linear amplification in time) is replaced by two nearby peaks of lower magnitude after installing the TMD device. Note that maximum values of acceleration shown in Figure 4.13 are obtained from transient responses integrated over a limited time duration ($10.0\,s$). The effect of a 1% damping in the principal system also results in significant reduction in the acceleration response relative to the new peaks, as expected for resonant conditions. An increased dissipation (10% damping ratio) in the TMD device has the effect of further reducing the peak acceleration response.

4.5.4 Uncertainty Quantification of passive damping efficiency

When designing a TMD device for a given structural system, some of the quantities maybe difficult to estimate or the actual structure may be under construction leaving the engineer only with possible *ranges* of design variables. Some of the sources of uncertainty can be eliminated by *adjustable* TMD devices where stiffness, frequency or damping can be tuned on site and even modified at later stages. Others, like frequencies of external forces, are inherently random and should be accounted as such within the whole design process. We provide an example where uncertainties are injected directly

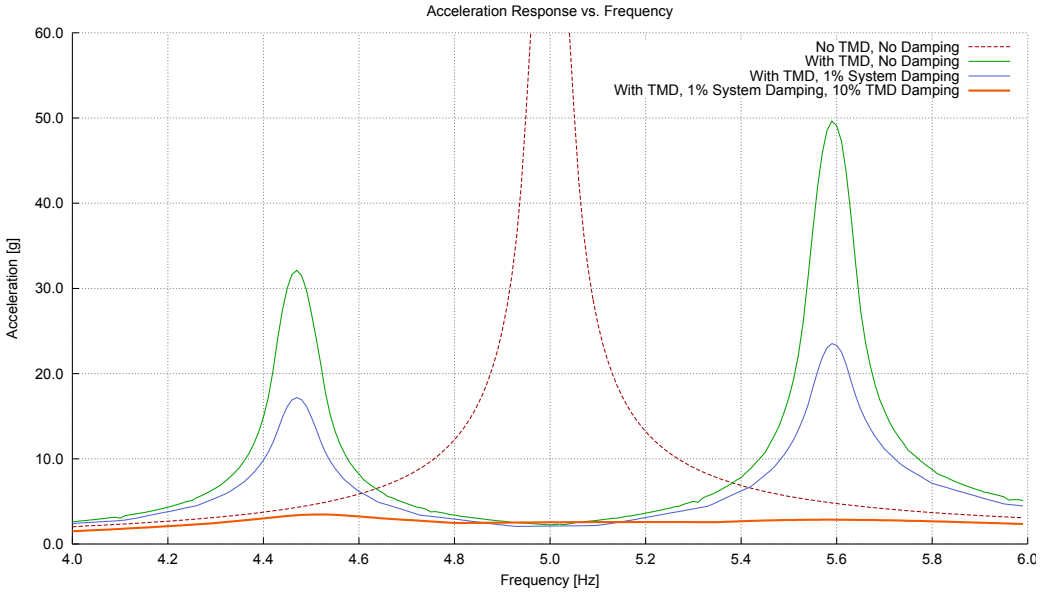


Figure 4.13: Acceleration response of 2 dofs dynamic system with and without the TMD device installed, for a range of forcing frequencies. The effect of variations in the damping raio of the principal system and TMD device are also explored.

in system (4.6) not only as far as the forcing term is concerned, but also for some of the system parameters. As a result, (4.6) becomes a system of stochastic ODEs; collocation in probability is used resulting in a non-intrusive approach leading to the characterization of the system's output based on repeated deterministic simulations. Even if a simple 2 dof system is analyzed here, the proposed procedure is general and can be applied to multiple degrees of freedom systems (mdof) with an arbitrary number of TMD devices. Note that for such configurations, the transient dynamic deterministic solution of the full system can be a computationally intensive task. Therefore, a propagation methodology aiming to be efficient for designers should keep to a minimum the required number of deterministic solutions.

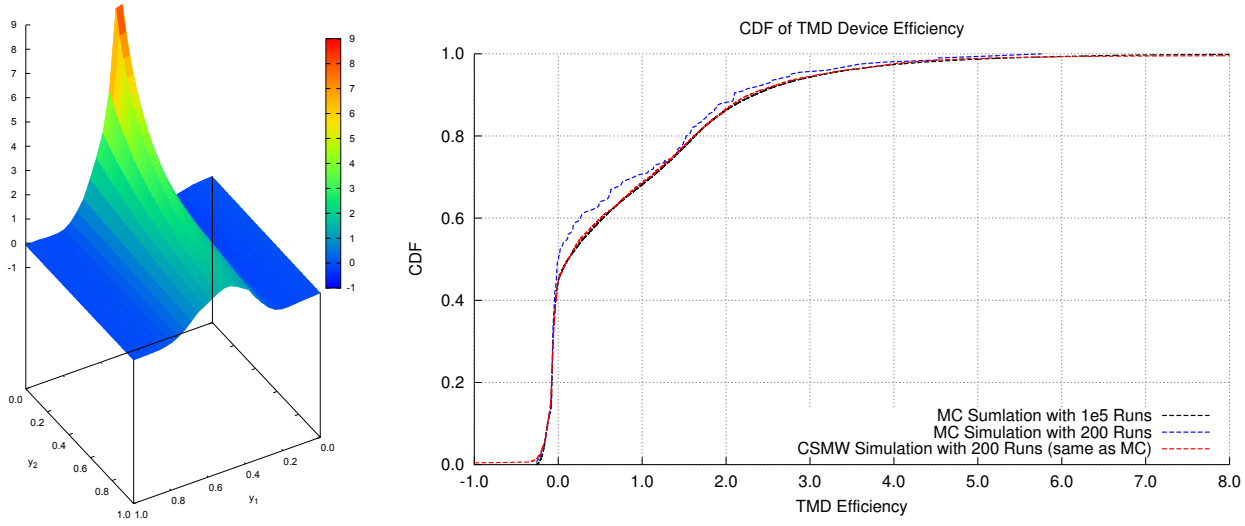
For our numerical experiment two sources of uncertainty are injected into system (4.6), namely the forcing frequency and the damping ratio of the principal system ξ_1. Randomness in the forcing frequency might results from environmental or anthropic actions; damping of the main structure could be difficult to measure in practice or the device could be designed to actually account for a range of possible damping ratios. The first is parametrized as $f = 4.0 + 2.0\,y_1$ by a uniformly distributed random variable $y_1 \in \mathcal{U}([0,1])$; the second as $\xi_1 = 0.01 + 0.05\,y_2$ where again $y_2 \in \mathcal{U}([0,1])$. For every couple of parameters (y_1,y_2), the response of the system is evaluated in terms of *efficiency* e, as follows:

$$e = \ddot{x}_{1,max}/\tilde{\ddot{x}}_{1,max} - 1, \tag{4.20}$$

where $\ddot{x}_{1,max}$ is the maximum acceleration of the principal system in the time interval $[0, 10\,s]$ without any vibration control device, while $\tilde{\ddot{x}}_{1,max}$ is the corresponding value with the TMD device installed. Note that when $e = 0$ no reduction in the maximum acceleration is provided by the device, while positive values are desirable. The cumulative distribution function for TMD efficiency is computed as a result of the stochastic problem formulated above. This curves show the probability of occurrence for efficiencies that are lower than a selected value.

Two methodologies are compared. First of all, 10^5 Monte Carlo simulations are performed to compute a reference CDF of e. A reasonable number of runs (200) is then selected which can normally be afforded even for transient simulations of full dynamical systems (e.g. a complete multi-storey building or mechanical assembly). Finally, the efficiency CDF curves computed with the same 200 runs are compared for the Monte Carlo method and the proposed CS-MW approach and

illustrated in Figure 4.14. A surface plot of the TMD efficiency for $(y_1, y_2) \in [0, 1]^2$ is illustrated in Figure 4.14a. Note that areas of steep gradients close to resonance can be observed, as expected. Finally, it can be seen how the CDF curve computed by the CS-MW methodology is practically superimposed to the one requiring 10^5 Monte Carlo runs and offers a fair accuracy for design purposes. Furthermore, Monte Carlo estimation with 200 samples would result in higher probabilities for lower values of TMD efficiency, thus underestimating the capabilities of the passive vibration control device.



(a) (b)

Figure 4.14: (a) Representation of the stochastic response in term of efficiency for the 2 dofs system. (b) Resulting efficiency CDFs computed with the Monte Carlo and CS-MW approaches.

4.6 Application of non-intrusive UP: robust design of windmill airfoil sections

Recently, wind power generation is drawing attentions as a way of making use of natural energy. It is the process to generate electric power by conversion of wind energy into propeller rotation. Crucial to the technology is the ability to achieve high conversion efficiency under constantly changing wind conditions. The velocity triangle of a windmill airfoil section is illustrated in Figure 4.15. It can be seen how the change in wind velocity V_a results in changes in the angle of attack and inflow velocity relative to the airfoil section. Under such conditions, optimization should guarantee a stable performance while maximizing the rotation thrust or, in other words, should be formulated as a robust process. However, due to the significant computational cost associated to traditional Monte Carlo-like strategies used in conjunction with CFD simulations, design optimization is often carried out for specific environmental conditions. The objective of this study is two-fold:

1. to investigate and show the increase in computational cost involved in the transition from traditional to robust optimization for windmill airfoil section profiles and

2. to explore efficient optimization methods.

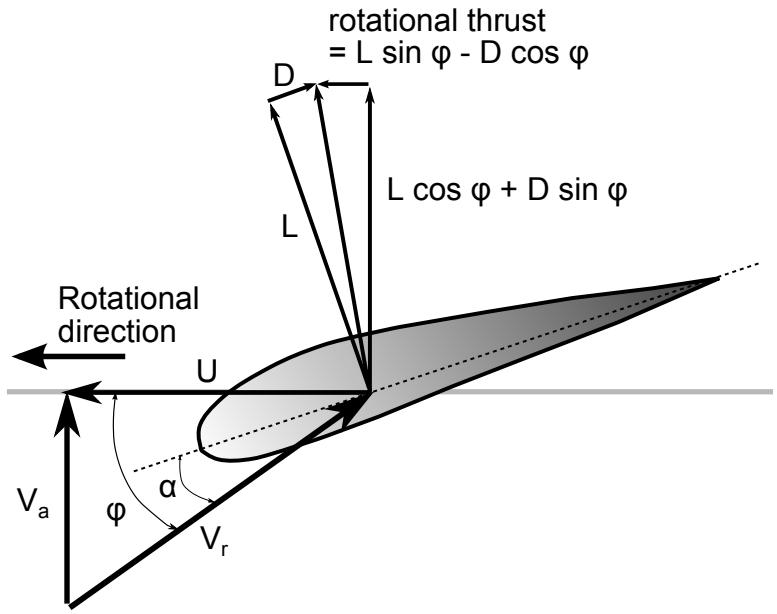


Figure 4.15: Schematic representation of velocity triangle for a windmill airfoil section.

4.6.1 Problem formulation

Two optimization problems are analyzed in the present study. A *traditional* optimization strategy is firstly approached, where no variation of inflow wind conditions is considered. It is formulated as a single objective optimization, maximizing a measure of aerodynamic efficiency, i.e. the lift to drag coefficient ratio C_L/C_D.

Robust optimization is successively investigated, where the disturbances in the inflow angle and Mach number are associated to uniformly distributed random variables α and M, respectively. The average performance is thus maximized over a range of wind conditions, while minimizing the associated variance, i.e. the sensitivity to the stochastic environmental changes.

Incompressible flow conditions are assumed throughout; the flow field is computed using a RANS solver at constant Reynolds number, once the airfoil geometry and wind conditions are determined by realizations of the parameters. Assume the airfoil cross section geometry is determined by a vector $\boldsymbol{\xi} \in \mathcal{D}$ of parameters, where $\xi = \{\xi_1, \ldots, \xi_{n_p}\}$, $\xi_i^l \leqslant \xi_i \leqslant \xi_i^u$ and $\mathcal{D}$ is the compact space of feasible airfoil configurations; ξ_i^l, ξ_i^u are the lower and upper bound respectively for the generic parameter ξ_i. Moreover, consider a probability space $(\Omega, \mathcal{F}, \mathcal{P})$ in which Ω is the set of elementary event, $\mathcal{F}$ is the σ-algebra of events, and $\mathcal{P}$ defines a probability measure on $\mathcal{F}$. A vector of two independent and identically distributed random variables with joint probability density function $\rho(\mathbf{y}) = \rho_\alpha(\alpha)\rho_M(M) : \mathbb{R}^2 \to \mathbb{R}_{\geqslant 0}$ is indicated by $\mathbf{y} = (\alpha, M)$ with $\alpha, M : \Omega \to \mathbb{R}$. Realization of the stochastic vector $\mathbf{y}$ are denoted by $\mathbf{y}^{(i)} = (\alpha^{(i)}, M^{(i)})$, $i = 1, \ldots, n_l$. Consider $f : \mathcal{D} \times \Omega \to \mathbb{R}$ a function mapping design and stochastic parameters into lift to drag coefficient ratios. We also stress that every evaluation of f requires prior sampling of design parameters $\boldsymbol{\xi}$ and environmental variables $\mathbf{y}^{(i)} = (\alpha^{(i)}, M^{(i)})$ followed by a solution of a RANS simulation. Design optimization can be formulated using two different approaches:

1. Without including the uncertainty in wind conditions (traditional optimization). Find $\boldsymbol{\xi}^* \in \mathcal{D}$

such that:

$$\boldsymbol{\xi}^* = \arg\max_{\boldsymbol{\xi}\in\mathcal{D}} f(\boldsymbol{\xi}, \mathbf{y}^{(i)}) \tag{4.21}$$

2. Including the environmental uncertainty (robust approach). Find $\boldsymbol{\xi}^* \in \mathcal{D}$ such that:

$$\mathcal{M}\left(\mathbb{E}\{f(\boldsymbol{\xi}^*, \mathbf{y})\},\, \sigma\{f(\boldsymbol{\xi}^*, \mathbf{y})\}\right) > \mathcal{M}\left(\mathbb{E}\{f(\boldsymbol{\xi}, \mathbf{y})\},\, \sigma\{f(\boldsymbol{\xi}, \mathbf{y})\}\right) \quad \forall \boldsymbol{\xi} \in \mathcal{D} \tag{4.22}$$

where $\mathbb{E}\{\cdot\}$ and $\sigma\{\cdot\}$ denote the expectation and standard deviation operator, expressed as:

$$\mathbb{E}\{f(\boldsymbol{\xi}, \mathbf{y})\} = \int_{M_l}^{M_u} \int_{\alpha_l}^{\alpha_u} f(\boldsymbol{\xi}, \mathbf{y}) \rho_\alpha(\alpha) \rho_M(M)\, d\alpha\, dM \tag{4.23}$$

$$\sigma\{f(\boldsymbol{\xi}, \mathbf{y})\} = \left(\int_{M_l}^{M_u} \int_{\alpha_l}^{\alpha_u} \left(f^2(\boldsymbol{\xi}, \mathbf{y}) - \left(\mathbb{E}\{f(\boldsymbol{\xi}, \mathbf{y})\}\right)^2 \right) \rho_\alpha(\alpha) \rho_M(M)\, d\alpha\, dM \right)^{1/2}, \tag{4.24}$$

and $\mathcal{M} : \mathbb{R}^2 \to \mathbb{R}$ is a multi-objective decision criteria. In other words, the solution of the robust optimization problem maximizes the mean lift-drag ratio across variables environmental conditions while minimizing the associated standard deviation, i.e. the sensitivity of the airfoil efficiency to alterations in the angle of attack and Mach number. Here $[\alpha_l, \alpha_u]$ and $[M_l, M_u]$ are elementary event intervals associated to the inflow angle α and Mach number M, respectively. In practice, we adopt:

$$[\alpha_l, \alpha_u] = [\alpha_0 - 2, \alpha_0 + 2] \quad \text{with} \quad \alpha_0 = 10 \tag{4.25}$$

$$[M_l, M_u] = [M_0 - 0.05, M_0 + 0.05] \quad \text{with} \quad M_0 = 0.25. \tag{4.26}$$

As formulated in equation (4.22), the problem is essentially a multi-objective optimization whose solution is generated with a trade-off between the two objective functions, i.e. the average performance and sensitivity to environmental conditions. Note that different choices of $\mathcal{M}$ lead to different optimal solutions. For example, given two objectives $o_1, o_2 \in \mathbb{R}$, $\mathcal{M}(o_1, o_2) = o_1$ generates the solution with the maximum average efficiency, $\mathcal{M}(o_1, o_2) = -o_2$ that with minimum sensitivity, while $\mathcal{M}(o_1, o_2) = \sqrt{o_1^2 + o_2^2}$ might be chosen as a compromise.

4.6.2 Airfoil representation

The PARSEC [83] representation is used to define a parametric airfoil profile. In this representation, the upper and the lower section curves are expressed by polynomials of the following form:

$$z = \sum_{i=1}^{6} a_i\, x^{(i-1/2)}, \tag{4.27}$$

where the section is defined in the (x,z) plane. The coefficients a_i are determined from the geometric parameters $\boldsymbol{\xi} \in \mathcal{D}$ (design variables), illustrated in Figure 4.16. Eleven design variables are selected: leading edge radius, upper and lower crest locations and curvatures, trailing edge coordinates (at x=1), thickness, direction angle and wedge angle. Design variables and corresponding ranges are represented in Table 4.1. As PARSEC tries to minimize the number of parameters needed to generate wing profiles of practical interest in applications, it is useful to consider the possible interaction between some of these parameters. An example is illustrated in Figure 4.17, where different values are applied to the parameters z_{TE} and $z_{lo} + z_{up}$, repsectively.

ID	Descr.	ξ_l	ξ_u	ID	Descr.	ξ_l	ξ_u
r_{le}	Radius of LE	0.005	0.02	$z_{xx,lo}$	Curv. of lower surf.	0.3	0.9
x_{up}	X at crest of upper surf.	0.3	0.7	z_{TE}	Y at TE	-0.01	0.05
z_{up}	Y at crest of upper surf.	0.12	0.18	α_{TE}	Camber gradient at TE	-13.0	-3.0
$z_{xx,up}$	Curv. of upper surf.	-0.4	0.0	ΔZ_{TE}	Thickness at TE	0.0	0.0
x_{lo}	X at crest of lower surf.	0.2	0.6	β_{TE}	Wedge angle at TE	4.0	8.0
z_{lo}	Y at crest of lower surf.	-0.07	0.02	M	Mach number	0.2	0.3
α	Angle of attack	8.0	12.0				

Table 4.1: List of parameters used in PARSEC [83]

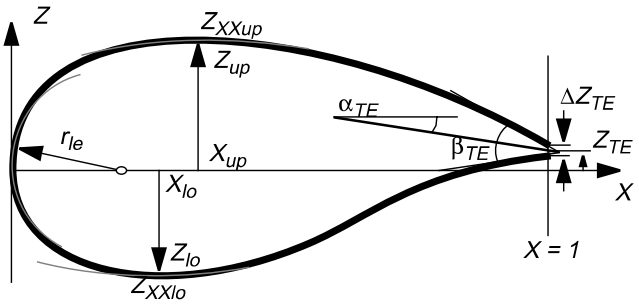


Figure 4.16: Graphical representation of the input parameters for PARSEC [83].

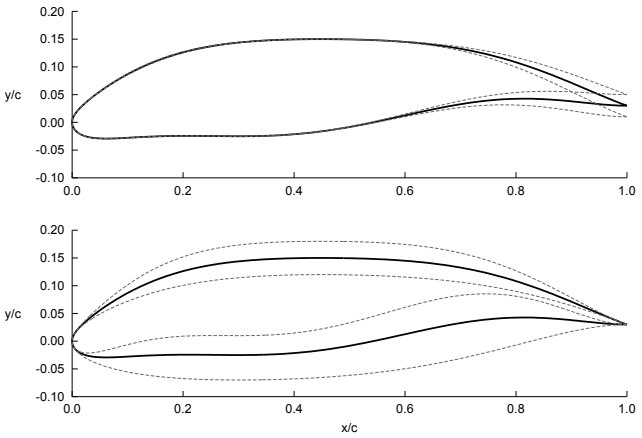


Figure 4.17: Change in airfoil configuration as a result of adjusting z_{TE} (top) or $z_{lo} + z_{up}$ (bottom).

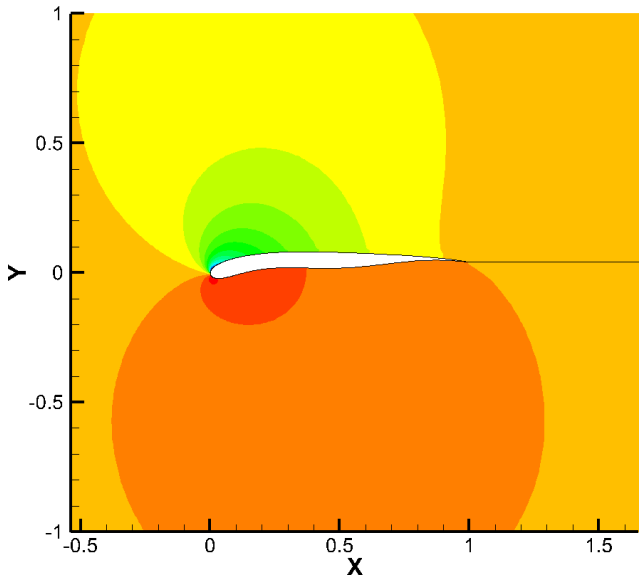


Figure 4.18: Unconstrained optimal airfoil design for certain wind conditions.

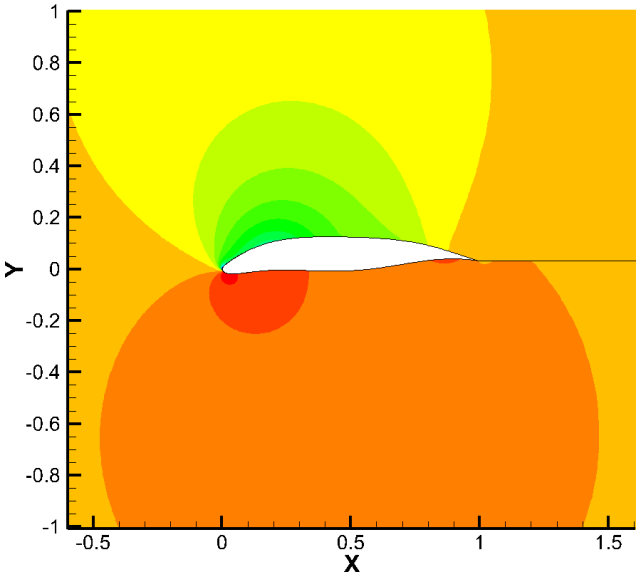


Figure 4.19: Constrained optimal airfoil design for certain wind conditions.

4.6.3 Computation of lift and drag, preliminary optimizations

Once the airfoil profile is determined by the parameter set $\boldsymbol{\xi}$, the solution of the two-dimensional incompressible Navier-Stokes equation is sought using a fixed Reynolds number equal to 10^5. An automatic meshing procedure is first carried out starting from the profile generated by PARSEC. The CFD solver, developed internally by Honda, uses a delta form implicit finite difference method. A 3rd order Chakravarthy-Osher TVD limiter is also used for advection together with Menter's k-ω SST fully transitional turbulence model.

As a first, preliminary step, the windmill profile was optimized under known wind conditions. The resultant shape is shown in Figure 4.18. A very thin airfoil is generated, as expected, exhibiting a pronounced curvature of the leading edge in the direction of the selected angle of attack. This solution, although being characterized by very high levels of aerodynamic efficiency, shows an insufficient section modulus thus suffering from a lack of structural strength. An alteration to the initial design parameter space $\mathcal{D}$ is thus required to fulfill both aerodynamic and structural feasibility. The modified *constrained* optimal solution is represented in Figure 4.19.

4.6.4 Robust Optimization methods

Shape optimization of windmill airfoils is performed by means of Genetic Algorithms (GA). This family of methodologies seeks optimal solutions by selectively creating successive generations of individuals. Each individual represents a design configuration with *chromosomes* $\boldsymbol{\xi}$, i.e. the 11 section design parameters. Each individual is also associated to a set of environmental conditions, that is, realizations of angle of attach and Mach number $(\alpha^{(i)}, M^{(i)})$, $i = 0, \ldots, n_l$. These locations are carefully selected to facilitate the computation of statistics via multivariate quadrature. This differentiate a robust approach from traditional optimization where a single evaluation of f is sufficient for every individual in the population.

Now we assume that $f(\boldsymbol{\xi}^*, \mathbf{y})$ takes the form:

$$f(\boldsymbol{\xi}^*, \mathbf{y}) \approx \sum_{i=0}^{P} \alpha_i(\boldsymbol{\xi}^*)\phi_i(\mathbf{y}) \tag{4.28}$$

as a finite linear combination of tensor product orthogonal polynomials of the random vector $\mathbf{y}$ (polynomial chaos expansion). The quantities $E\{f(\boldsymbol{\xi}^*, \mathbf{y})\}$ and $\sigma\{f(\boldsymbol{\xi}^*, \mathbf{y})\}$ can be computed by numerical integration as follows:

$$\mathbb{E}\{f(\boldsymbol{\xi}^*, \mathbf{y})\} \approx \sum_{j=1}^{n_l} w_j \, f\left(\boldsymbol{\xi}^*, \mathbf{y}^{(j)}\right) \tag{4.29}$$

$$\sigma\{f(\boldsymbol{\xi}^*, \mathbf{y})\} \approx \sum_{j=1}^{n_l} w_j \left[f\left(\boldsymbol{\xi}^*, \mathbf{y}^{(j)}\right)^2 - \mathbb{E}\left\{ f\left(\boldsymbol{\xi}^*, \mathbf{y}^{(j)}\right)\right\}^2 \right] \tag{4.30}$$

where $\sum_{j=1}^{n_l} w_j = 1$. Here the n_l quadrature locations are the zeros of the selected tensor product polynomial family $\phi_i(\mathbf{y})$. Is is well known [99], that optimal convergence to the statistics of sufficiently smooth stochastic responses is obtained by employing polynomial families orthogonal to the input probability measures. Our choice of adopting Clenshaw-Curtis quadrature locations translates in expanding f using Chebyshev polynomials. For applications where f is not known in advance and arbitrary input probability measures could be specified, we feel that the good convergence properties of Chebyshev approximants give us a good compromise to be implemented in a general framework. For every computed generation, the steps performed by GA are highlighted in algorithm 4.

Algorithm 4 Genetic Algorithm

Step 1 ▷ Make the initial population of individuals at random.
Select PARSEC parameters $\boldsymbol{\xi}^*$ uniformly within the design space.
Select quadrature locations for Angle of attach and Mach number.
Step 2 ▷ Evaluate the fitness of each individual in that population.
Calculate $\mathbb{E}\{f(\boldsymbol{\xi}^*, \mathbf{y})\}$, $\sigma\{f(\boldsymbol{\xi}^*, \mathbf{y})\}$ for each individual.
Step 3 ▷ Repeat on this generation until termination.
Select the best-fit individuals for reproduction.
Breed new individuals through crossover and mutation, to give birth to offspring.
Evaluate the individual fitness of new individuals.
Replace least-fit population with new individuals.

It is a known fact that the computational cost of evaluating multivariate integrals, like $E\{f(\boldsymbol{\xi}, \mathbf{y})\}$, with given accuracy leads to a dramatic increase of polynomial terms (or numerical integration points) for a corresponding increase in the number of stochastic input variables. This fact is generally addressed as the *curse of dimensionality*; a typical trend is shown in Table 4.2. We also stress that every evaluation of $f(\boldsymbol{\xi}^*, \mathbf{y}^{(i)})$ requires the complete solution of a RANS fluid dynamic simulation. It is therefore clear how the number of deterministic realizations must be kept to a minimum if robust optimization is to be performed in a reasonable time. In this study, we adopt a Smolyak sparse grid approach together with nested Clenshaw-Curtis quadrature as a possible mitigation of this phenomenon. A two-dimensional Smolyak sparse grid with 13 points is used in the present study. Approximation order 0, 1 and 2 are thus recovered with 1, 5 and 13 quadrature points, respectively, allowing rough estimates of convergence and accuracy to be computed. Figure 4.20 shows the location of the selected quadrature locations in the α-M plane. Finally, we note that this study employs a non-intrusive approach which uses an unmodified deterministic CFD solver even if stochasticity has been injected into the equations. This approach is justified by the fact that the accuracy of the RANS solution is constant for the selected parameter ranges.

Dimension	1	2	3	4	5	d
Full Grid	5	25	125	625	3125	5^d
Smolyak Sparse Grid	5	13	33	89	253	$3^d + 2d$

Table 4.2: Increase in multivariate quadrature points with dimensionality for fixed one-dimensional polynomial accuracy

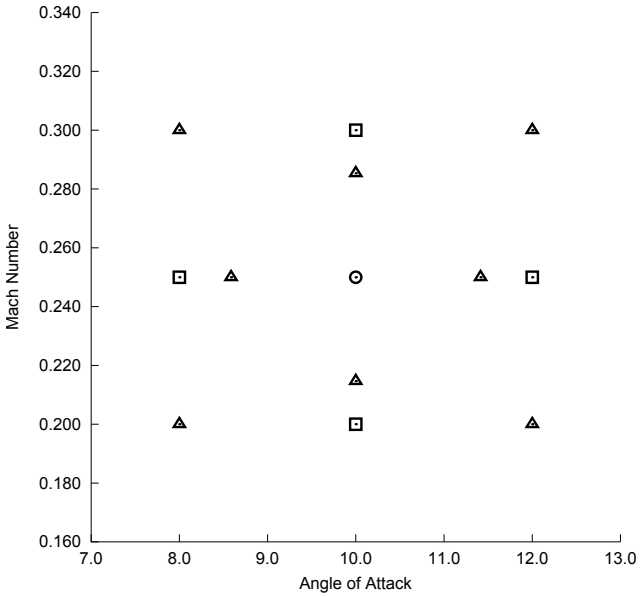


Figure 4.20: Two dimensional Smolyak sparse tensor quadrature grid up to order 2 accuracy. Level 0 (◯), Level 1 (□) and Level 2 (△) incremental grids are show.

4.6.5 Optimal windmill airfoils

At a first stage, a sensitivity analysis is performed to assess how the optimal design is influenced by some of the GA parameters, i.e. population size, cross probability, flip probability. This is performed for fixed environmental conditions $(\alpha, M) = (10, 0.25)$; results are illustrated in Table 4.3. For a sufficient population size, the sensitivity to the GA parameters has a limited effect on the optimal solution.

Two optimization tasks are then carried out denominated *traditional* and *robust*, generating optimal designs $\boldsymbol{\xi}_T^*$ and $\boldsymbol{\xi}_R^*$, respectively. The location of the traditional optimal design in the $(C_D, 1/C_L)$ plane is highlighted in Figure 4.21. After the set of parameters $\boldsymbol{\xi}_T^*$ is found giving the maximum C_L/C_D ratio for fixed environmental conditions $(\alpha, M) = (10, 0.25)$, an uncertainty propagation analysis is carried out leading to $\mathbb{E}\{f(\boldsymbol{\xi}_T^*, \mathbf{y})\} = 30.46$ and $\sigma\{f(\boldsymbol{\xi}_T^*, \mathbf{y})\} = 2.53$ (Table 4.4).

Some of the generations produced by robust optimization are also shown in Figure 4.22. Optimal designs have been reported which correspond to various metrics $\mathcal{M}$, i.e. maximum expected efficiency, minimum standard deviation and best compromise. Values of $\mathbb{E}\{f(\boldsymbol{\xi}_R^*, \mathbf{y})\}$ and $\sigma\{f(\boldsymbol{\xi}_R^*, \mathbf{y})\}$ for designs selected according to the above metrics are also reported in Table 4.4.

Note that first, second and also third order statistics have been included in the analysis. Moreover, a negative skewness is observed for most reported designs, showing an asymmetry of the pdf of f with a longer tail towards values of smaller efficiency. This provides an even stronger motivation for including variance minimization as a further optimization objective.

Population size	20	50	100	Optimal C_L/C_D	38.20	37.70	39.22
Cross probability	0.2	0.4	0.6	Optimal C_L/C_D	37.94	39.23	39.22
Flip probability	0.002	0.004	0.02	Optimal C_L/C_D	39.22	39.58	38.22

Table 4.3: Sensitivity of optimal design to GA parameters

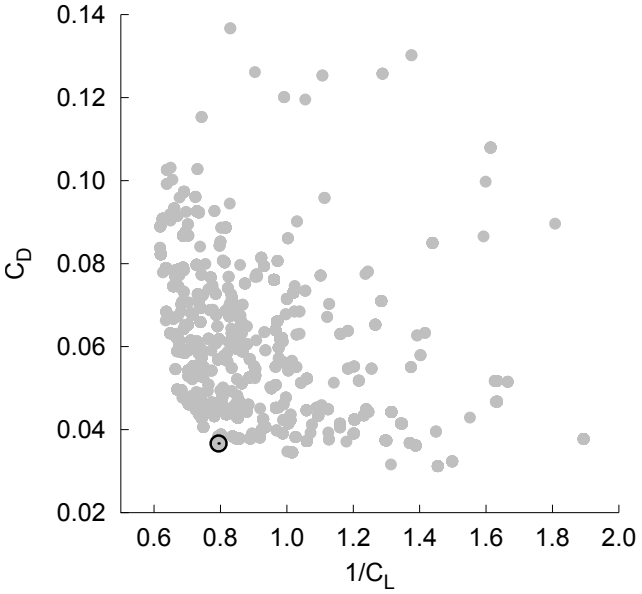


Figure 4.21: Individuals generated by GA for single-objective optimization not accounting for variability in wind conditions. The optimal constrained design (◯) is highlighted.

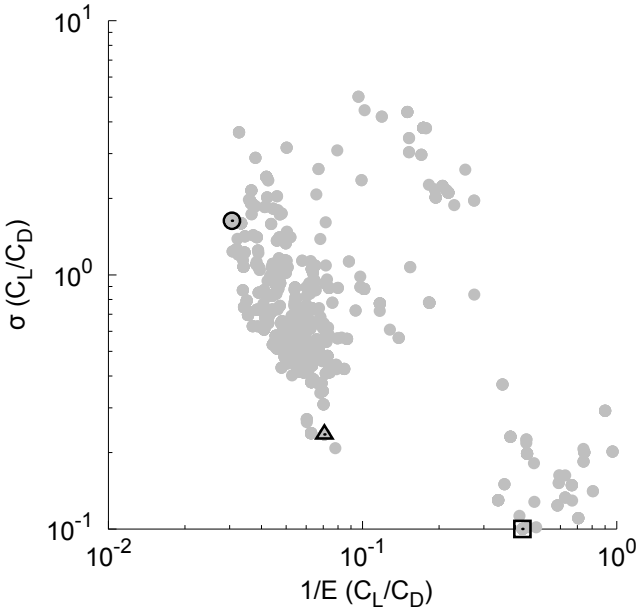


Figure 4.22: Efficiency-Sensitivity tradeoff resulting from robust optimization. Designs with maximum average efficiency (◯), minimum combined metric (△) and minimum sensitivity (□) are highlighted.

Robust optimal designs with minimum standard deviation and best compromise metrics are much less sensible to the environmental conditions than traditional optima. However, the average performance is also significantly smaller in this case.

The robust design maximizing the expected efficiency metric results, as expected, in better average performance across environmental conditions than the traditional optimal solution. As a result of the explicit inclusion of the variance minimization in the optimization task, the robust optimal is also less sensitive to changes of wind conditions.

In concluding, the proposed robust optimization framework has proven successful in improving the efficiency of a traditional optimum across a spectrum of uncertain wind conditions.

Design	Statistic	Value	Design	Statistic	Value
Max C_L/C_D	Mean	30.46	Min Distance	Mean	15.96
	SD	2.53		SD	0.24
	SK	−0.45		SK	−0.04
Max Expected Value	Mean	32.68	Min Standard Deviation	Mean	2.35
	SD	1.64		SD	0.10
	SK	−0.40		SK	0.12

Table 4.4: Comparison of results for traditional and robust optimizations.

4.6.6 Conclusion

A robust optimization framework has been assembled using Genetic Algorithms with uncertainty propagation techniques and applied to maximize the efficiency of a windmill airfoil over a spectrum of environmental scenarios.

A Smolyak sparse tensor grid of nested Clenshaw-Curtis one-dimensional quadrature rules is used to mitigate the curse of dimensionality, of special interest for the presented application, where airfoil efficiency is evaluated by solving a complete RANS simulation for any realization in the parameter space and given wind conditions.

Robust optimization has proven successful in providing designs performing better than traditional optima over a range of uncertain wind conditions, thus leading to savings in manufacturing resources and increasing the generated power.

This technique is particularly appealing in the development of industrial products which perform under variable environmental conditions. If quantities of interest exhibit sufficiently smooth variations in response to parameter changes, then sparse grid approaches can be used to minimize the number of deterministic solutions needed, thus reducing the overall computation cost.

Moreover, robust optimization provides a systematic and theoretically sound way to account for aleatoric uncertainty in engineering design.

Chapter 5

Velocity correction

5.1 Introduction

A Galerkin projection onto the space of linear iso-parametric simplicial elements is a common approach to find numerical solutions for the steady state diffusion equation. Piecewise constant *velocities* (product of scalar diffusivity and gradient) computed with this approach are not continuous across neighbor elements. Moreover, velocities with non zero components orthogonal to zero flux boundaries are also observed. A streamline representation of the product between diffusivity and gradient helps in highlighting the above inaccuracies. Various schemes are available in literature to compute streamlines of a vector field. For example, the Euler or second order Runge-Kutta methods have been widely implemented in commercial fluid dynamics visualization tools. In this article, a Euler approach is used with constant element velocities.

Figure 5.1 shows a typical streamline representation resulting from P_1 Galerkin, which highlights the inaccuracies discussed above. Blue streamlines correctly flow through the domain from Dirichlet to Dirichlet boundaries; red streamlines violate zero flux boundaries while green ones terminate over edges where convergent velocities occur for neighbor elements.

To improve accuracy in the computed gradients, approaches are available in literature which conserve fluxes across elements. In particular, edge gradients are elevated to main problem unknowns for Mixed-Hybrid (MH in the following) finite elements (see, i.e., [18]). Although both the theory and implementation of MH introduce complications respect to P_1 Galerkin, it produces the best results in terms of streamlines and will be considered a reference throughout. Therefore, we investigate post-processing strategies to improve on P_1 Galerkin fluxes in order to match MH results.

These strategies are inspired by research developed during the 80's and 90's in the context of error estimation and adaptivity for the finite element method. The first contribution can be found in [56] in the context of finding a self-equilibrated configuration of the residual for second order elliptic problems. It forms the basis for a complementary *a posteriori* energy overestimation of the finite element discretization error. Generalizations were provided in [4, 3, 5] where local problems on element *stars* (clusters of elements sharing the same node) were addressed for the first time. The proposed approach is suitable for arbitrary dimensionality and can be used on both conforming and non conforming meshes.

In [61] the above ideas are applied to restore an element-wise conservative flux from P_1 Galerkin velocities. Both a global (with piecewise constant egde corrections) and a parallel (with piecewise linear corrections) local star-based algorithms have been presented, together with numerical tests assessing existence, uniqueness and convergence. Note that, in the latter tests, a uniform diffusivity

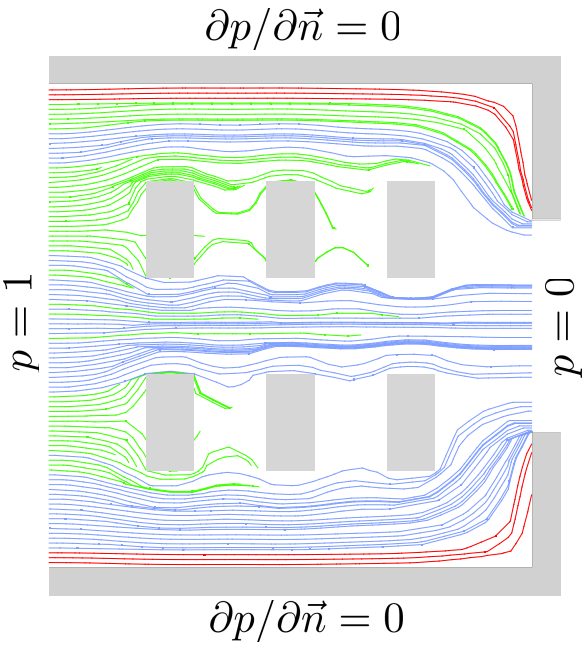


Figure 5.1: An example of steady state diffusion streamlines, as resulting from P_1 Galerkin.

is assumed throughout the computational domain. Of the two strategies proposed in [61], the local LN algorithm is computationally more appealing. In fact the solution of small algebraic systems (one for every star) is more efficient than solving a bigger system for the whole mesh and velocity corrections for each star can be evaluated independently leading to parallel, more efficient code.

Applications of the LN correction algorithm in the context of variably saturated groundwater flow have been proposed in [54]. Conforming and nonconforming finite element formulations for the solution of the Richards' equation are compared in this study. Note that, diffusivity ranges were limited to one order of magnitude in the proposed numerical examples.

Our study focuses instead on the LN strategy applied to cases where diffusivity ranges of several order of magnitudes are specified between adjacent elements. For relatively coarse meshes and especially for areas where streamlines exhibit high curvatures, we show that LN overestimates velocities in regions of low diffusivity. An inexpensive correction of the LN methodology is therefore suggested (MLN) which restores correct velocities across regions of high jumps in diffusivity.

In section 5.2 the LN post-processing algorithm is highlighted within the framework of steady state diffusion. Section 5.3 proposes a modification of the LN scheme which improve solutions with large diffusivity gradients. The lowest order Raviart-Thomas (RT_0) interpolation is discussed in Section 5.3.3 for cases where a non-zero source is applied. Examples of 2D trajectories are reported in section 5.3.4. The convergence rate of various algorithms is investigated in section 5.4 together with a comparison between MLN, LN and MH resultant steady state trajectories.

5.2 The heterogeneous flow problem

The general steady state diffusion equation can be written as:

$$\begin{cases} -\nabla \cdot (K \nabla p) = f(x) & \text{in } \Omega \\ p(x) = g(x) & \text{on } \Gamma^D \\ K \nabla p \cdot \vec{n} = q(x) & \text{on } \Gamma^N \end{cases} \tag{5.1}$$

defined on a domain $\Omega \subset \mathbb{R}^d$ (with $d = 2$ or 3), a bounded and nonempty open set with Lipschitz continuous boundary $\Gamma = \partial\Omega = \overline{\Gamma^D} \cup \overline{\Gamma^N}$, where Γ^D is the Dirichlet boundary, Γ^N the Neumann boundary satisfying $\Gamma^D \cap \Gamma^N = \varnothing$ and Γ^D having positive measure. We assume sufficient regularity properties for $f(x)$, $g(x)$ and $q(x)$. We consider a non degenerate scalar diffusion coefficient, such that $K(x) \geqslant \gamma > 0$ a.e. in Ω can vary abruptly within Ω. The velocity field $\vec{v}(x)$ is given by:

$$\vec{v} = -K(x)\nabla p. \tag{5.2}$$

Note that the name *velocity* is chosen to match the physical meaning this quantity has in some applications (i.e., seepage). We also use the term *flux* to indicate the normal velocity integrated over mesh edges or faces. The solution $p(x)$ of (5.1) is continuous and smooth on Ω; in order to match physical requirements on interfaces where jumps in the diffusion coefficient occur, the gradient ∇p is assumed to be discontinuous, while the velocity vector has continuous normal derivative. Moreover, if $f(x) = 0$, $p(x)$ satisfies the maximum principle, i.e., no local maxima/minima occur in the interior of Ω. This property reflects the standard regularity properties of the velocity field $v(x)$, the so called *divergence-free* property.

In the present work, we focus on two possible strategies for the numerical solution of (5.1), namely the P_1 Galerkin and lowest order Mixed-Hybrid finite element methods. For simplicity, all developments will be formulated in two dimensions ($d = 2$). The results illustrated in the following sections can be readily extended to the $d = 3$ case.

Let $\mathcal{T}_h(\Omega) = \{T_e, e = 1, \ldots, m_\tau\}$ be a triangulation of Ω, with diameter h, featuring m_τ triangles, m_ϵ edges, and m_ν nodes.

The P_1 Galerkin method evaluates the discrete pressure field $p_h(x)$ as:

$$p_h(x) = \sum_{i=1}^{m_\nu} p_i N_i(x), \tag{5.3}$$

where $N_i(x)$ is the piecewise linear basis function on $\mathcal{T}_h(\Omega)$ such that $N_i(x_j) = \delta_{ij}$ for $i, j = 1, \ldots, m_\nu$ and δ_{ij} is the Kronecker delta. The coefficients p_i are the solution of the linear system

$$\int_\Omega K\nabla p_h \nabla N_j \, d\Omega = \int_\Omega f N_j \, d\Omega \qquad\qquad j = 1, 2 \ldots, m_\nu. \tag{5.4}$$

The P_1 velocity is evaluated on every triangle T_e as:

$$\nabla p_{h,T_e} = \sum_{k \in n_{T_e}} p_k \nabla N_k, \tag{5.5}$$

where n_{T_e} is the index set of the three vertices in T_e. The ensuing discrete velocity field on T_e is

$$\vec{v}_{h,T_e} = -K\nabla p_{h,T_e}. \tag{5.6}$$

Let us recall here some relevant properties of the P_1 numerical solution $(p_h, \vec{v}_h)$ [78]

- p_h is continuous, piecewise linear;
- $\vec{v}_h$ is elementwise constant;
- the tangential component of the gradient $\nabla p_h \cdot \vec{t}_\epsilon$ is continuous across each edge ϵ ($\vec{t}_\epsilon$ is the unit tangent to edge ϵ);

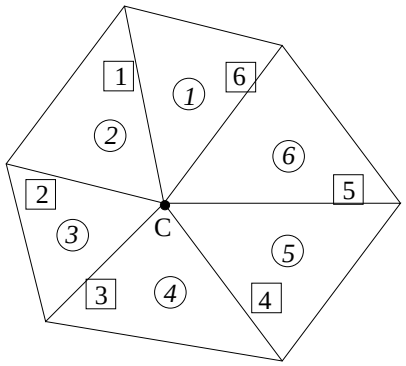


Figure 5.2: Typical element star, centered at node C. Local element (circles) and edge (squares) numbers are shown.

- the normal flux $\vec{v}_h \cdot \vec{n}_\epsilon$ is discontinuous across every internal edge ϵ ($\vec{n}_\epsilon$ is the outward unit normal to edge ϵ).

For comparison purposes, we also consider lowest order Raviart-Thomas space with hybridization (MH), see [18] for details. Let us recall some relevant properties of the MH solution, p'_h, and the velocity field $\vec{v}'_h$ [57, 77].

- p'_h is piecewise linear, and discontinuous across element edges;
- when evaluated on triangle centroids, $\vec{v}'_h$ yields an element-by-element piecewise constant field;
- continuity of the gradient tangential component across edges is not assured;
- the normal flux is continuous across every internal edge.

The last property ensures elementwise mass conservation (see, i.e., [77]). In an attempt to provide a self contained exposition, the next section reviews the post-processing method proposed in [61].

5.2.1 Larson-Niklasson post-processing

Let us denote by Ω_C a star of elements centered on node C (i.e., the set of all triangles sharing node C). Figure 5.2 shows a typical configuration. LN initially requires a trial velocity vector, $\vec{\sigma}_\epsilon$, to be assembled at every edge ϵ. A reasonable initial guess for $\vec{\sigma}_\epsilon$ is the arithmetic average of the P_1 Galerkin velocities for the two triangles sharing edge ϵ. However, the algorithm is robust to different choices in this regard; for example, similar corrections are produced using velocities with minimum or maximum magnitude. Let ϵ_l and ϵ_r denote the elements to the left (l) and right (r) of edge ϵ. The quantities U_{C,T_e} are velocity corrections associated to each element T_e of the star. Let U_{C,ϵ_l} and U_{C,ϵ_r} be the flux corrections associated to the element to the left, and to the right of ϵ, respectively and let $N_C(x)$ be the P_1 basis function assigned to node C.

If we consider a partial Galerkin P_1 projection using only the shape function $N_C(x)$ and we apply the Green's lemma, the following residual for element $T_e \in \Omega_C$ is obtained.

$$\sum_{\epsilon \in \partial T_e \cap \Gamma_I} \int_\epsilon \vec{\sigma}_\epsilon \cdot \vec{n}_\epsilon \, N_C \, ds - \int_{T_e} K \nabla p_h \cdot \nabla N_C \, d\Omega + \int_{T_e} f \, N_C \, d\Omega + \int_{\partial T_e \cap \Gamma_N} q^{(N)} \, N_C \, ds = R(p_h) \tag{5.7}$$

where Γ_I is the set of internal mesh edges, Γ_D, Γ_N are the set of Dirichlet and Neumann boundary edges, respectively and $q^{(N)}$ is the prescribed Neumann flux. The local LN strategy modifies (5.7) in order to generate a null residual on each element of the star. This is accomplished, for every $\epsilon \in \Gamma_I$ by first adding the correction $\delta u_\epsilon = (U_{C,\epsilon_l}\vec{n}_{\epsilon_l} - U_{C,\epsilon_r}\vec{n}_{\epsilon_r}) \cdot \vec{n}_l$ for end C, and then using linear interpolation between the two end nodes of ϵ. Note that $\vec{n}_l$ is arbitrarily chosen as the positive flux direction for edge ϵ. The following balance equation for element T_e is formulated:

$$\begin{aligned}
&\sum_{\epsilon \in \partial T_e \cap \Gamma_I} \int_\epsilon \left[(U_{C,\epsilon_l} - U_{C,\epsilon_r})\, \vec{n}_\epsilon^{(T_e)} \cdot \vec{n}_l \right] N_C \, ds = \\
&\sum_{\epsilon \in \partial T_e \cap \Gamma_I} \int_\epsilon \vec{\sigma}_\epsilon \cdot \vec{n}_\epsilon^{(T_e)} \, N_C \, ds - \int_{T_e} K \nabla p_h \cdot \nabla N_C \, d\Omega \\
&+ \int_{T_e} f \, N_C \, d\Omega + \int_{\partial T_e \cap \Gamma_N} q^{(N)} \, N_C \, ds
\end{aligned} \tag{5.8}$$

Equation (5.8) is written for each element of the star, yielding a symmetric positive semi-definite algebraic system of equations in the form $By = c$. With reference to the star sketched in Figure 5.2, we have:

$$B = \begin{bmatrix}
l_6 + l_1 & -l_1 & 0 & 0 & 0 & -l_6 \\
-l_1 & l_1 + l_2 & -l_2 & 0 & 0 & 0 \\
0 & -l_2 & l_2 + l_3 & -l_3 & 0 & 0 \\
0 & 0 & -l_3 & l_3 + l_4 & -l_4 & 0 \\
0 & 0 & 0 & -l_4 & l_4 + l_5 & -l_5 \\
-l_6 & 0 & 0 & 0 & -l_5 & l_6 + l_5
\end{bmatrix} \tag{5.9}$$

$$y = \begin{bmatrix} U_{C,1} \\ U_{C,2} \\ U_{C,3} \\ U_{C,4} \\ U_{C,5} \\ U_{C,6} \end{bmatrix} \quad c = \begin{bmatrix}
-\vec{\sigma}_6 \cdot \vec{n}_6^1 l_6 - \vec{\sigma}_1 \cdot \vec{n}_1^1 l_1 + 2G_1 \\
-\vec{\sigma}_1 \cdot \vec{n}_1^2 l_1 - \vec{\sigma}_2 \cdot \vec{n}_2^2 l_2 + 2G_2 \\
-\vec{\sigma}_2 \cdot \vec{n}_2^3 l_2 - \vec{\sigma}_3 \cdot \vec{n}_3^3 l_3 + 2G_3 \\
-\vec{\sigma}_3 \cdot \vec{n}_3^4 l_3 - \vec{\sigma}_4 \cdot \vec{n}_4^4 l_4 + 2G_4 \\
-\vec{\sigma}_4 \cdot \vec{n}_4^5 l_4 - \vec{\sigma}_5 \cdot \vec{n}_5^5 l_5 + 2G_5 \\
-\vec{\sigma}_5 \cdot \vec{n}_5^6 l_5 - \vec{\sigma}_6 \cdot \vec{n}_6^6 l_6 + 2G_6
\end{bmatrix}, \tag{5.10}$$

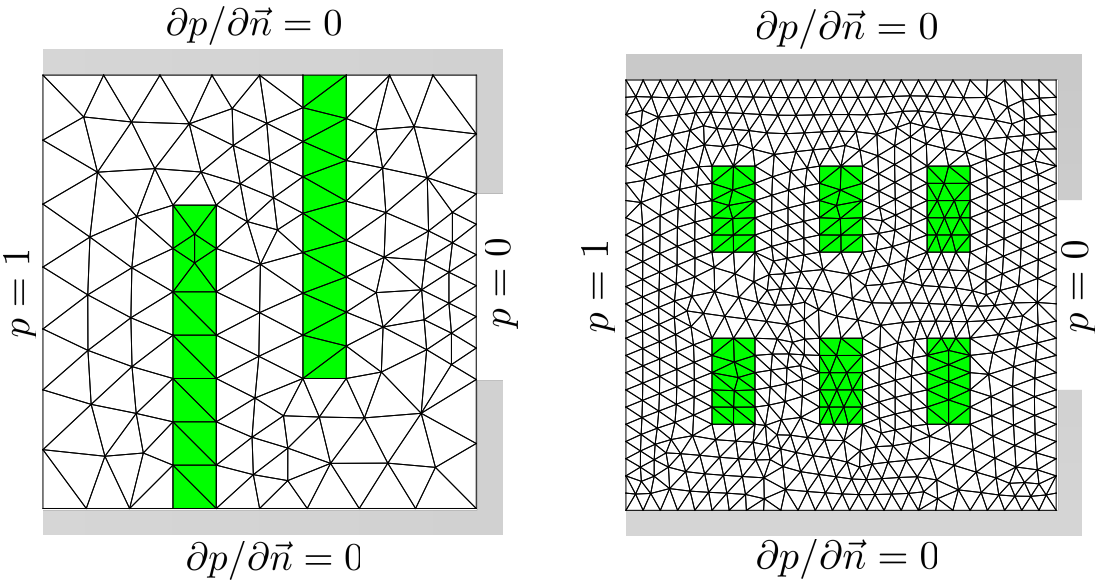
where l_ϵ is the length of edge ϵ. Recall that $\vec{n}_\epsilon^{(T_e)}$ is a unit normal to edge ϵ pointing outward of element T_e. Moreover, G_{T_e} is the Galerkin residual flux for element T_e, i.e.

$$G_{T_e} = \int_{T_e} f \, N_C \, d\Omega - \int_{T_e} K \nabla p_h \cdot \nabla N_C \, d\Omega. \tag{5.11}$$

For internal stars (where all edges sharing node C are in Γ_I), the above linear system exhibits the following properties:

1. Since the elements of the right hand side vector are P_1 Galerkin flux residuals on the elements of the star, they add up to zero;
2. Matrix B is singular as constant vectors span its null space. One can therefore arbitrarily assign a zero value to the fist unknown, i.e., $U_{C,1} = 0$.

For every edge ϵ and node C a correction is computed $\delta U_{C,\epsilon} = U_{C,\epsilon_l} - U_{C,\epsilon_r}$ for the initial velocity estimate $\vec{\sigma}_\epsilon$. If a parametrization of edge ϵ (defined from node N_1 to N_2) is introduced through a scalar parameter $\psi \in [0, 1]$, corrected normal velocities on ϵ are obtained as $U'_\epsilon = \vec{\sigma}_\epsilon + (1-\psi)\delta U_{N_1,\epsilon} + \psi \delta U_{N_2,\epsilon}$. The post-processed LN edge velocities U'_ϵ fulfill integral counterparts of divergence-free requirements. Associated fluxes can be interpolated using the lowest order Raviart Thomas RT_0 finite element space (see, i.e., [80]), thus providing a velocity field on $\mathcal{T}_h(\Omega)$.



(a) Test case 1 (b) Test case 2

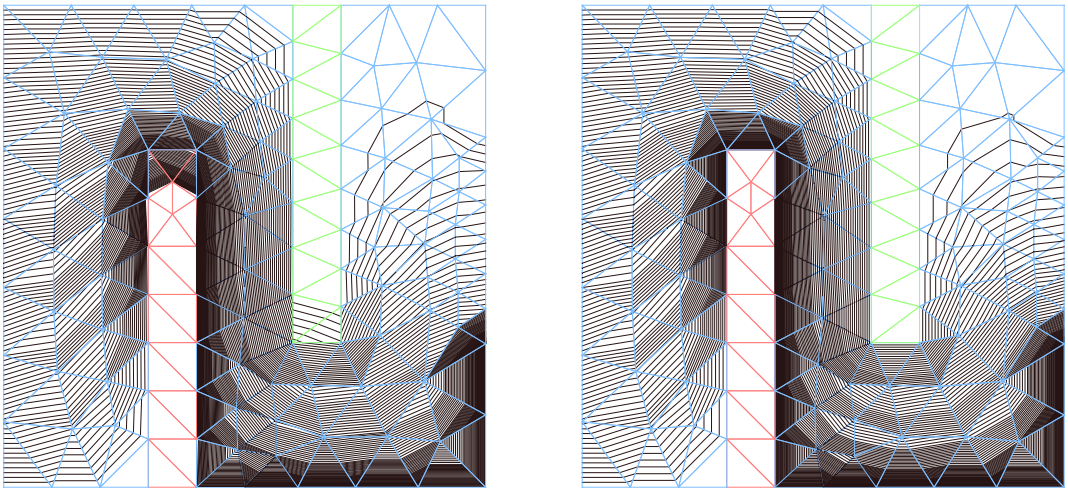
Figure 5.3: Test cases 1 and 2 with associated geometries, underlying Delaunay triangulations and boundary conditions.

We analyze streamline flow patterns solving (5.1) on two heterogeneous test cases using the LN and MH techniques, similarly to what proposed in [77]. We assume $\Omega = [0, 1]^2$ with inflow from the left side, i.e., where a $p_h = 1$ Dirichlet boundary condition is imposed. Outflow on a central portion of the right boundary is set, where $p_h = 0$ is imposed. Zero flux Neumann boundary conditions are set elsewhere. The scalar, homogeneous diffusivity K changes abruptly on two internal fins, whose scalar diffusivity is $K_1 = 10^{-6} \times K$. Figure 5.3 shows all relevant details. The second test case involves the same domain and boundary conditions; six low diffusivity fins, shown in Figure 5.3, are inserted. Each fin has diffusivity $K_1 = 10^{-6} \times K$.

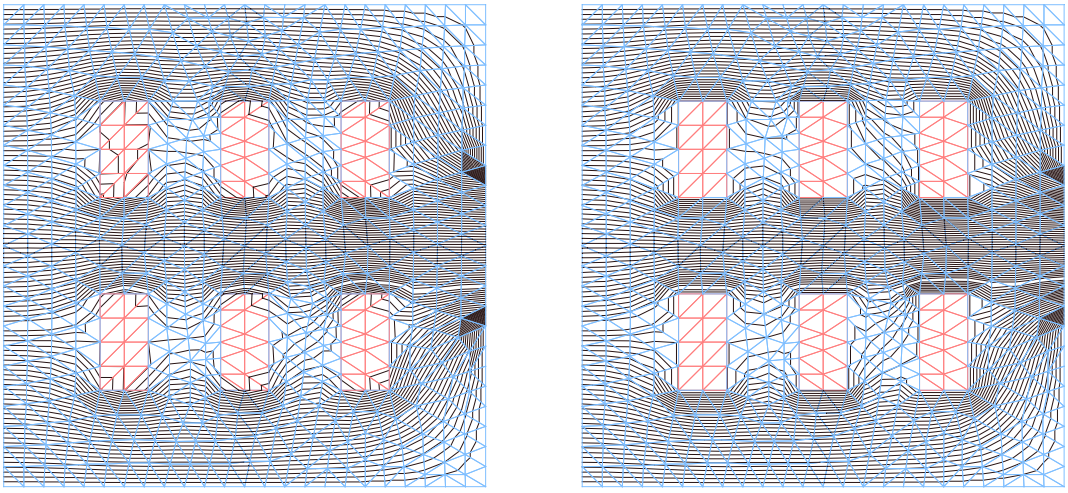
As in [77], both streamlines terminating exactly on nodes (where the velocity field is non unique) and trajectories exiting from zero flux boundaries, are identified. Figure 5.4 shows a comparison between streamlines obtained either by LN post-processing or MH approach. At a first glance, the two sets of streamlines look very similar. Both velocity fields are conservative, with differences concentrated at the corner elements of the fins. In particular, unlike in the MH approach, LN streamlines cross low diffusivity fins. This behaviour can be observed only on a limited number of stars on areas becoming smaller with mesh refinement. Nonetheless, with particular reference to coarse discretizations, advection of quantities are negatively affected at those locations. In the following sections, we propose a modification of the LN technique which aims at minimizing such situations.

5.3 Modified LN scheme

In the algorithm discussed in section 5.2.1, velocity corrections are not influenced by element diffusivities. The final algebraic system is in fact formulated merely on flux balance arguments. The present section proposes an extension to account for such an information. We need to understand, in first place, how to evaluate a reference flux magnitude that can be used for this purpose. To develop some intuition, we use the patch examples shown in Figure 5.5. Our purpose is to find a methodology to identify the node stars where overestimated velocities result from the LN procedure.



(a) (b)



(c) (d)

Figure 5.4: Streamlines computed for test cases 1 (a,b) and 2 (c,d) are shown. The LN (a,c), and MH (b,d) approaches are used.

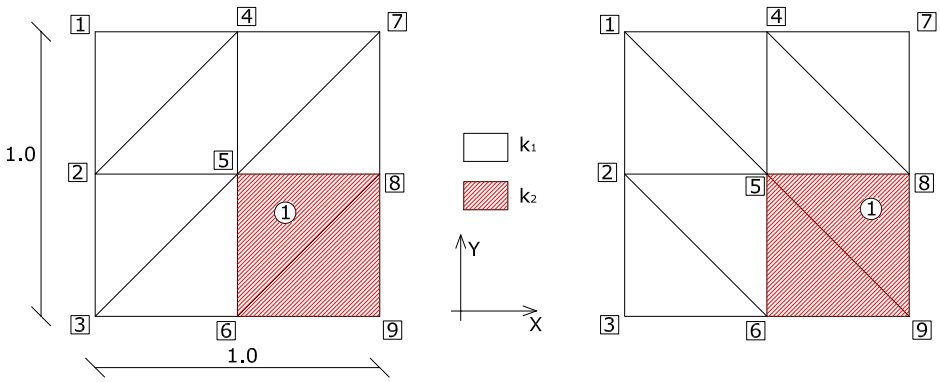


Figure 5.5: Element patches selected for numerical tests. Dirichlet boundary conditions are imposed on all boundary edges.

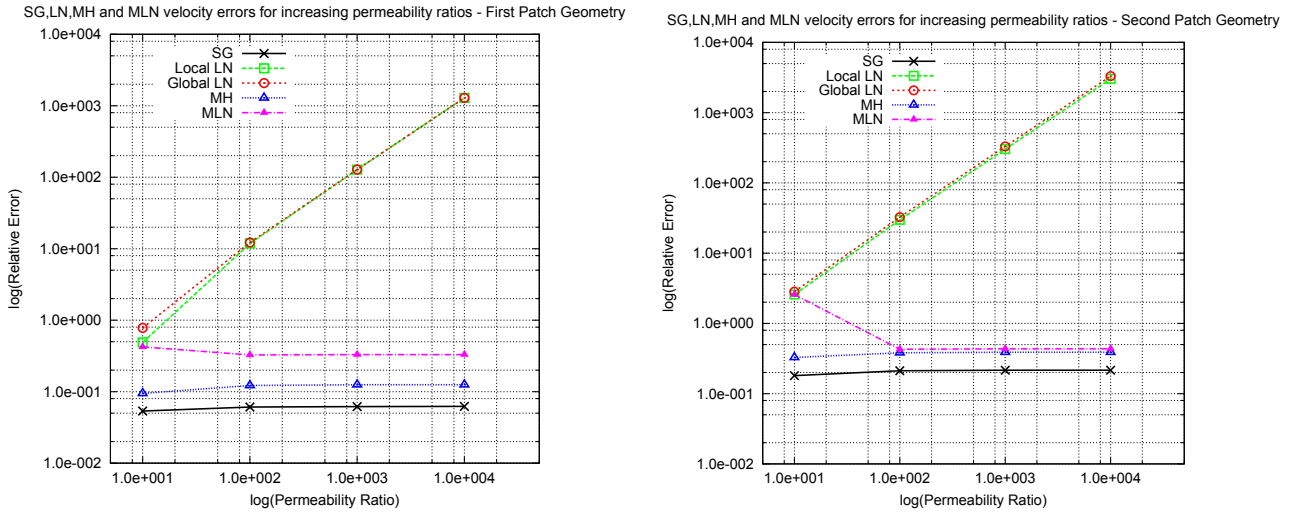


Figure 5.6: Velocity errors in patch tests computed for increasing diffusivity ratios.

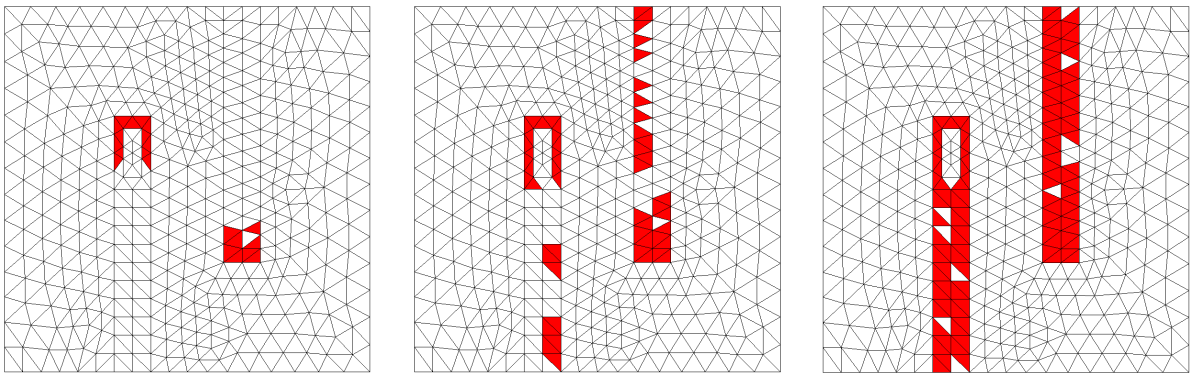


Figure 5.7: Red elements are located where the error estimate E_{h,T_e} exceeds $10^5\%$ (a), $10^4\%$ (b) and $10^3\%$ (c), respectively.

5.3.1 Error identification

As the P_1 Galerkin velocity magnitudes are readily available at the beginning of the post-processing stage, we investigate if they can be used as a reference. Assume we solve problem (5.1) on a square, by using either one of the meshes shown in Figure 5.5. We use two different patch configurations in order to assess the sensitivity of the computed corrections to the number of low diffusivity elements in the central star (one and two, respectively). Dirichlet conditions are imposed all around the patch; $p = 1$ is applied on the left side, $p = 0$ on the right side and a linear variation is assumed at the bottom and top boundary edges. A K_2 diffusivity is applied on hatched elements, while $K_1 = 1$ is considered for all others. Numerical simulations are performed for the following values of $K_2 = K_1 \cdot 10^{-k}$, $k = 0, 1, 2, 3, 4$. The global P_1 Galerkin algebraic system reduces to a single scalar equation with p_5 as the only unknown.

We focus, in particular, to the velocity magnitude as corrected by the LN method at the centroid of triangle 1. A reference solution, in this regard, is computed via P_1 Galerkin, using a mesh (obtained by uniform refinements) with diameter equal to $h = 4.42 \times 10^{-2}$. Relative error results are summarized in Figure 5.6, where the relative velocity error is plotted against the diffusivity ratio, in log-log scale. Unlike errors for both P_1 Galerkin and MH that are practically unaffected

by diffusivity ratios, errors in LN velocities increase significantly. Hence, ratios between original P_1 Galerkin and post-processed velocities can be used in the LN approach as an error marker. Results obtained with the proposed MLN correction method are also shown in Figure 5.6.

We therefore define an error estimate, as follows:

$$E_{h,T_e} = \frac{\|\vec{v}_{h,T_e}^{(LN)}\| - \|\vec{v}_{h,T_e}^{(P_1)}\|}{\|\vec{v}_{h,T_e}^{(P_1)}\|} \tag{5.12}$$

Once a suitable tolerance value τ_h is selected, elements where $E_{h,T_e} > \tau_h$ are identified. Smaller values of τ_h lead to larger number of elements needing corrections, as shown for test case 1 in Figure 5.7.

5.3.2 Error correction

Given a node C, Let $\Sigma_{h,C}$ be the subset of elements in the star exceeding threshold τ_h, i.e.

$$\Sigma_{h,C} = \{T_e \in \mathcal{T}_{h,C} : E_{h,T_e} > \tau_h\}$$

For each element $T_e \in \Sigma_{h,C}$ we add the following constraint to the LN linear system (written in terms of the Lagrange multiplier λ):

$$\lambda \left\{ \int_\epsilon \left[(U_{C,T_e} - U_{C,\tilde{T}_e})\, \vec{n}_\epsilon^{(T_e)} \cdot \vec{n}_l \right] N_C\, ds + \int_\epsilon (\vec{v}_\epsilon^{(LN)} \cdot \vec{n}_\epsilon^{(T_e)})\, N_C\, ds - \int_\epsilon (\vec{v}_\epsilon^{(P_1)} \cdot \vec{n}_\epsilon^{(T_e)})\, N_C\, ds \right\} = 0, \tag{5.13}$$

where element $\tilde{T}_e$ shares edge ϵ with element T_e, $\vec{v}_\epsilon^{(LN)}$ is the linear velocity profile on edge ϵ reconstructed with LN and $\vec{v}_\epsilon^{(P_1)}$ is the P_1 Galerkin velocity distribution at edge ϵ. In other words, equation (5.13) restores the flux on ϵ to the value computed with P_1 Galerkin.

After the LN post-processing is performed, a new correction step is applied in the MLN strategy only for the stars with elements in $\Sigma_{h,C}$. The following modified equation set is assembled:

$$\begin{bmatrix} B & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} y \\ \lambda \end{bmatrix} = \begin{bmatrix} c \\ d \end{bmatrix}$$

where B is the matrix given by eq. (5.9), A and d are the matrix and right hand side, respectively, obtained by discretizing eq. (5.13), y our correction array, λ is the Lagrange multiplier vector and c is the right-hand side in eq. (5.10).

The following issues deserve attention:

- Our modified linear system is symmetric and singular, as in the LN formulation;
- A source term $C^{(T_e)}\lambda$ is added to $T_e \in \Sigma_{h,C}$ and to their neighbors;
- The procedure above can easily be extended to higher dimensional problems, and arbitrary polygonal meshes;
- The following limit in the number of Lagrangian equations n_λ is also adopted:

  $$n_\lambda \leqslant n_{\Sigma_c} - 1,$$

  where $n_{\Sigma_{h,C}}$ is $\Sigma_{h,C}$ cardinality;

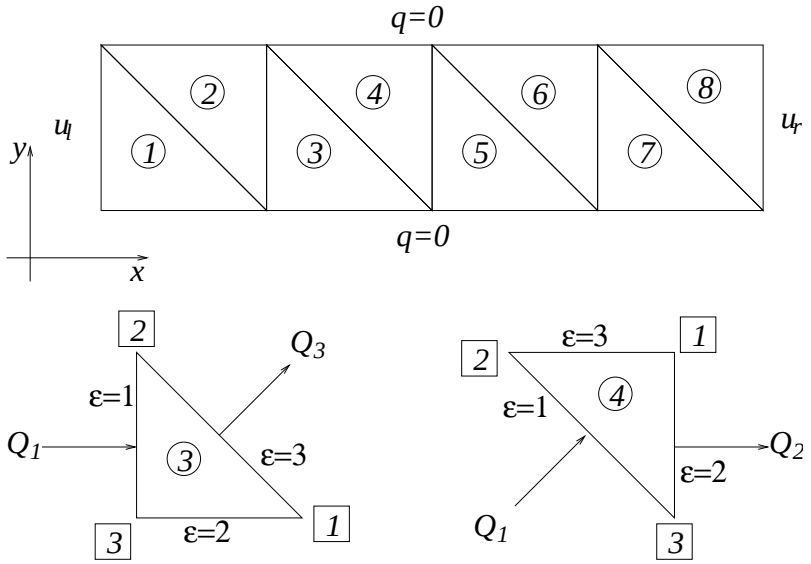


Figure 5.8: Selected 2D mesh (top) with boundary conditions for the proposed Poisson problem, with detail of elements 3 and 4 (bottom).

Our technique follows the same steps as in LN, therefore edge fluxes need to be interpolated using RT_0, in order to extract velocities on elements. Unlike LN, a source term now appears in MLN and the RT_0 interpolation strategy needs to be applied to elements with unbalanced edge fluxes. This needs special treatment, as discussed in the next section.

5.3.3 RT_0 interpolation and source terms

Let $\vec{w}_i^{(T_e)}$, $i = 1, 2, 3$, the RT_0 vector basis functions associated with a local edge numbering system. Edge flux interpolation can be written as

$$\vec{v}_h^{(T_e)}(x_1, x_2) = \sum_{\epsilon=1}^{3} q_\epsilon \cdot \vec{w}_\epsilon^{(T_e)}, \quad \text{where} \quad q_\epsilon = \int_\epsilon \vec{v}_h^{(T_e)} \cdot \vec{n}_\epsilon \, ds. \tag{5.14}$$

The adopted vector basis functions RT_0 are (see, i.e., [70])

$$\vec{w}_\epsilon^{(T_e)} = \frac{1}{2|T_e|} \left(\vec{x} - \vec{x}^{(\epsilon)} \right), \tag{5.15}$$

$\vec{x}^{(\epsilon)}$ being the vertex in T_e which does not belong to edge ϵ.

Note that, when source terms are included, spurious velocity components may be introduced by RT_0 interpolation. This can be easily seen in Figure 5.8 where a one-dimensional solution is simulated using a two-dimensional triangular grid. Dirichelet boundary conditions, u_l and u_r are applied to the left and to the right edge of the domain, respectively. A constant, non zero source term f is also applied to all triangles. The two triangles T_3 and T_4 are extracted as shown in the lower part of Figure 5.8. Flux conservation for element T_3 amounts to

$$Q_1 + Q_3 = f\,|T_3|. \tag{5.16}$$

Let $\vec{x}^{(O)} = (\vec{x}_1^{(O)}, \vec{x}_2^{(O)})$ be the centroid of T_e. The velocity vector $\vec{v}^{(T_3)} = (v_1^{(T_3)}, v_2^{(T_3)})$ on T_3 is

$$\begin{bmatrix} v_1 \\ v_2 \end{bmatrix}^{(T_3)} = \begin{bmatrix} \dfrac{1}{2|T_3|} \left[\left(x_1^{(O)} - x_1^{(1)} \right) \cdot Q_1 + \left(x_1^{(O)} - x_1^{(3)} \right) \cdot Q_3 \right] \\ \dfrac{1}{2|T_3|} \left[\left(x_2^{(O)} - x_2^{(1)} \right) \cdot Q_1 + \left(x_2^{(O)} - x_2^{(3)} \right) \cdot Q_3 \right] \end{bmatrix}. \tag{5.17}$$

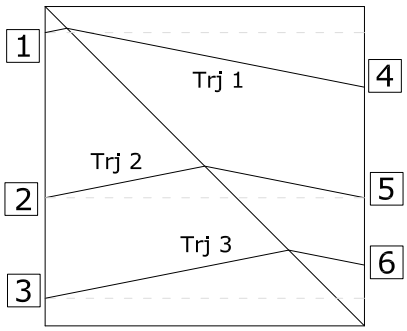


Figure 5.9: Computed trajectories. Points 4 and 6 at outflow are closer than points 1 and 3 at inflow.

This results in:

$$v_2^{(3)} = \frac{f}{2}\left(x_2^{(O)} - x_2^{(1)}\right) \quad \text{and} \quad v_2^{(4)} = \frac{f}{2}\left(x_2^{(O)} - x_2^{(2)}\right). \tag{5.18}$$

Figure 5.9 shows what happens to streamlines in triangles 3 and 4, where distance between streamlines descreases along x. Note that the this fact is invariant to edge swapping.

Source terms are added to locations close to edges where original Galerkin fluxes are restored. Moreover, at those locations, flux direction has already a significant component alog the diffusivity jump. Using these observations as a starting point, we propose a modification of the standard RT_0 interpolation which accounts for source terms. Using the local edge numbering in Figure 5.8, mass balance reads

$$\begin{aligned} Q_1 + Q_2 + Q_3 &= f|T_e| \\ v(\vec{x}) &= \frac{1}{2|T_e|}\left[Q_1(\vec{x}^{(O)} - \vec{x}_1) + Q_2(\vec{x}^{(O)} - \vec{x}_2) + Q_3(\vec{x}^{(O)} - \vec{x}_3)\right] \\ &= \frac{1}{2|T_e|}\left[Q_1(\vec{x}_3 - \vec{x}_1) + Q_2(\vec{x}_3 - \vec{x}_2) + f\,|T_e|(\vec{x}^{(O)} - \vec{x}_3)\right] \end{aligned} \tag{5.19}$$

Let i be the index of the edge where absolute maximum flux is evaluated, i.e. $Q_i = \max_{\epsilon\in 1,2,3}|Q_\epsilon|$. The proposed correction is:

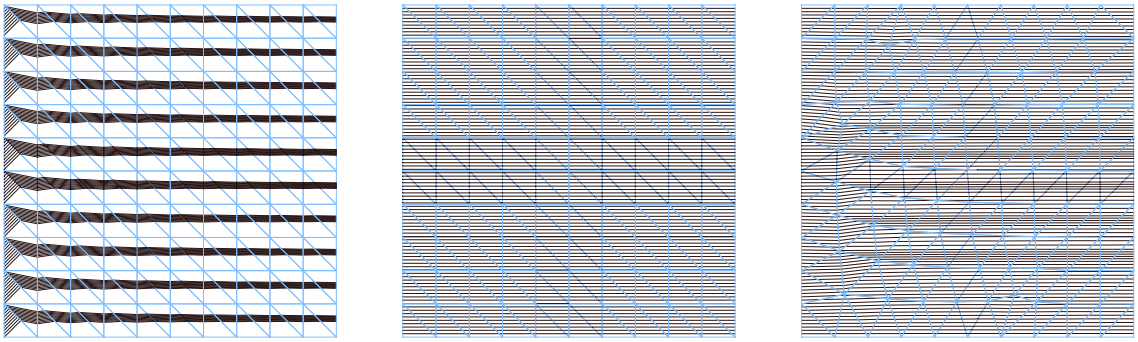
$$\Delta v_{corr} = \frac{f(\vec{x}^{(O)} - \vec{x}_i)}{2}\,\mathrm{sign}(Q_i). \tag{5.20}$$

For boundary triangles, all Dirichlet edges must be considered in (5.20).

We tested the proposed RT_0 interpolation on a problem defined over a square, with exact solution equal to a constant horizontal velocity field. Figure 5.10a shows the streamlines computed using the standard RT_0 velocity field while the effects of the proposed modified interpolation on a structured and unstructured grid are shown in Figure 5.10b and 5.10c, respectively. It can be seen how the proposed modified RT_0 interpolation produces streamlines much closer to the expected solution.

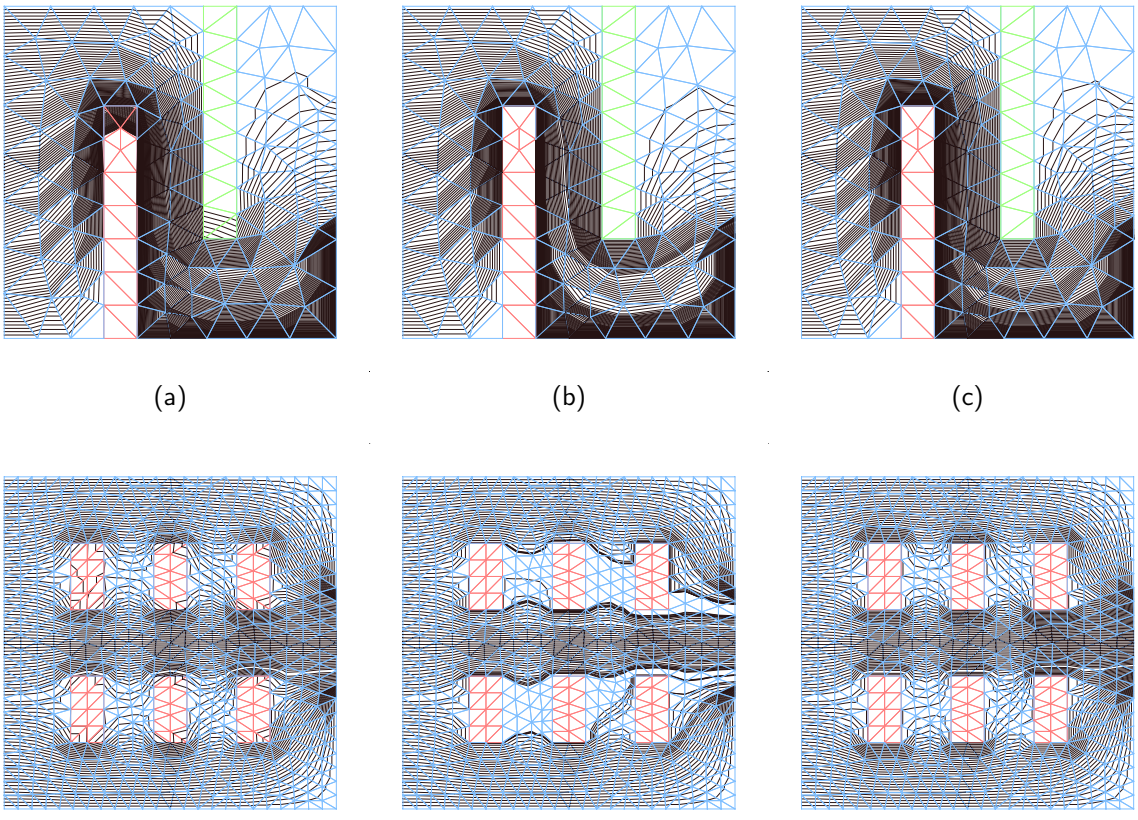
5.3.4 Corrected 2D trajectories

Streamlines calculated using MH, LN and MLN are compared in the present Section for test cases 1 and 2, respectively. Centroid velocities result from the lower order Raviart-Thomas shape functions (RT_0) for both LN and MH schemes, while the proposed modification of the RT_0 interpolation is employed in MLN. A permeability ratio $K_2 = 10^{-6} \times K_1$ is set in both cases. It can be seen how the proposed MLN correction restores physical velocities in areas of large diffusivity gradients. This improves advection in these areas especially in cases where a coarse discretization is needed. In practice, no streamline enters in areas of low diffusivity.



(a) (b) (c)

Figure 5.10: Streamlines computed from a 2D simulation of a 1D Poisson problem. Results are illustrated for RT_0 (a), together with the proposed approach both for a uniform (b) and non-uniform (c) mesh configuration.



(a) (b) (c)

(d) (e) (f)

Figure 5.11: LN (a,d), MLN (b,e) and MH (c,f) streamlines computed for the proposed test cases.

Refinement level	Elements	Nodes	Edges	Max edge size	$\mathcal{E}_p^{(SG)}$
I	98	64	161	0.217365	6.312×10^{-03}
II	392	225	616	0.108682	1.548×10^{-03}
III	1568	841	2408	0.054341	3.834×10^{-04}
IV	6272	3249	9520	0.027171	9.630×10^{-05}

Table 5.1: Mesh statistics

		Local LN		Global LN		MLN	
Mesh	Edge Size	Velocity error	Rate	Velocity error	Rate	Velocity error	Rate
I	2.174×10^{-01}	$1.583 \times 10^{+00}$	–	$1.617 \times 10^{+00}$	–	$2.934 \times 10^{+00}$	–
II	1.087×10^{-01}	8.411×10^{-01}	1.0955	8.645×10^{-01}	1.1074	$1.730 \times 10^{+00}$	1.3116
III	5.434×10^{-02}	4.337×10^{-01}	1.0464	4.383×10^{-01}	1.0204	9.405×10^{-01}	1.1379
IV	2.717×10^{-02}	2.170×10^{-01}	1.0012	2.179×10^{-01}	0.9917	4.991×10^{-01}	1.0937

Table 5.2: Errors for centroid velocity magnitudes. Local LN, Global LN and Modified LN (MLN) approaches are considered.

5.4 Convergence analysis

This section discusses how the schemes mentioned in the previous sections (P_1 Galerkin, Local LN, Global LN, MH and MLN) converge to an analytical solution by successive mesh refinements. A reference solution is chosen for system (5.1) where $\Omega = [0,1]^2$ with Dirichlet boundary conditions consistent with the potential:

$$p(x,y) = \sin^2(2\pi x) + \cos^2(2\pi y) + x + y + 5. \tag{5.21}$$

The corresponding source term is therefore equal to

$$f(x,y) = -16\pi^2(\cos^2(2\pi x) - (\cos^2(2\pi y)), \tag{5.22}$$

and the exact velocity field is

$$\begin{aligned} V_x(x,y) &= -4\pi \sin(2\pi x)\cos(2\pi x) - 1 \\ V_y(x,y) &= 4\pi \sin(2\pi y)\cos(2\pi y) - 1. \end{aligned}$$

Convergence rates are computed for velocity magnitudes and angle at element centroid as well as for edge fluxes. Let $\tilde{u}$ be the computed numerical approximation of the latter quantities and be u the associated exact solution of (5.1). The numerical error is evaluated as:

$$\mathcal{E} = \left(\sum_{i=1}^{N} |E_i| (u_i - \tilde{u}_i)^2 \right)^{\frac{1}{2}} \simeq \|u - \tilde{u}\|_{L_2}. \tag{5.23}$$

where $|E_i|$ is the area of the i-th element or the length of edge i, respectively.

		MH		SG	
Mesh	Edge Size	Velocity error	Rate	Velocity error	Rate
I	2.174×10^{-01}	$2.246 \times 10^{+00}$	–	$1.635 \times 10^{+00}$	–
II	1.087×10^{-01}	8.645×10^{-01}	0.7260	6.495×10^{-01}	0.7509
III	5.434×10^{-02}	4.569×10^{-01}	1.0870	2.943×10^{-01}	0.8755
IV	2.717×10^{-02}	2.199×10^{-01}	0.9477	1.446×10^{-01}	0.9758

Table 5.3: Velocity errors for MH and P_1 Galerkin.

Mesh	Subdivisions	Local LN FLUX Error	Rate	Global LN FLUX Error	Rate	MLN FLUX Error	Rate
I	364	$4.310 \times 10^{+00}$	–	$4.337 \times 10^{+00}$	–	$4.251 \times 10^{+00}$	–
II	230	$3.005 \times 10^{+00}$	1.9211	$3.022 \times 10^{+00}$	1.9190	$3.005 \times 10^{+00}$	1.9971
III	140	$1.841 \times 10^{+00}$	1.4156	$1.844 \times 10^{+00}$	1.4028	$1.841 \times 10^{+00}$	1.4156
IV	80	$1.255 \times 10^{+00}$	1.8067	$1.255 \times 10^{+00}$	1.8017	$1.255 \times 10^{+00}$	1.8067

Table 5.4: Edge flux errors. Local LN, Global LN and Modified LN (MLN) approaches are considered.

Mesh	Subdivisions	MH FLUX Error	Rate	SG FLUX Error	Rate
I	364	$4.328 \times 10^{+00}$	–	$3.710 \times 10^{+00}$	–
II	230	$2.896 \times 10^{+00}$	1.7252	$2.841 \times 10^{+00}$	2.5981
III	140	$1.816 \times 10^{+00}$	1.4842	$1.804 \times 10^{+00}$	1.5271
IV	80	$1.250 \times 10^{+00}$	1.8553	$1.247 \times 10^{+00}$	1.8778

Table 5.5: Flux errors for MH and P_1 Galerkin.

Mesh	Edge Size	Local LN Angle error	Rate	Global LN Angle error	Rate	MLN Angle error	Rate
I	2.174×10^{-01}	$2.129 \times 10^{+01}$	–	$2.085 \times 10^{+01}$	–	$3.976 \times 10^{+01}$	–
II	1.087×10^{-01}	$1.393 \times 10^{+01}$	1.6355	$1.402 \times 10^{+01}$	1.7469	$2.672 \times 10^{+01}$	1.7448
III	5.434×10^{-02}	$6.862 \times 10^{+00}$	0.9786	$7.123 \times 10^{+00}$	1.0234	$1.408 \times 10^{+01}$	1.0820
IV	2.717×10^{-02}	$4.284 \times 10^{+00}$	1.4717	$4.221 \times 10^{+00}$	1.3246	$7.395 \times 10^{+00}$	1.0759

Table 5.6: Velocity angle errors. Local LN, Global LN and Modified LN (MLN) approaches are considered.

Mesh	Edge Size	MH Angle error	Rate	SG Angle error	Rate
I	2.174×10^{-01}	$2.996 \times 10^{+01}$	–	$2.451 \times 10^{+01}$	–
II	1.087×10^{-01}	$1.402 \times 10^{+01}$	0.9131	$9.422 \times 10^{+00}$	0.7250
III	5.434×10^{-02}	$7.526 \times 10^{+00}$	1.1139	$7.800 \times 10^{+00}$	3.6695
IV	2.717×10^{-02}	$4.251 \times 10^{+00}$	1.2133	$3.308 \times 10^{+00}$	0.8079

Table 5.7: Velocity angle errors for MH and P_1 Galerkin.

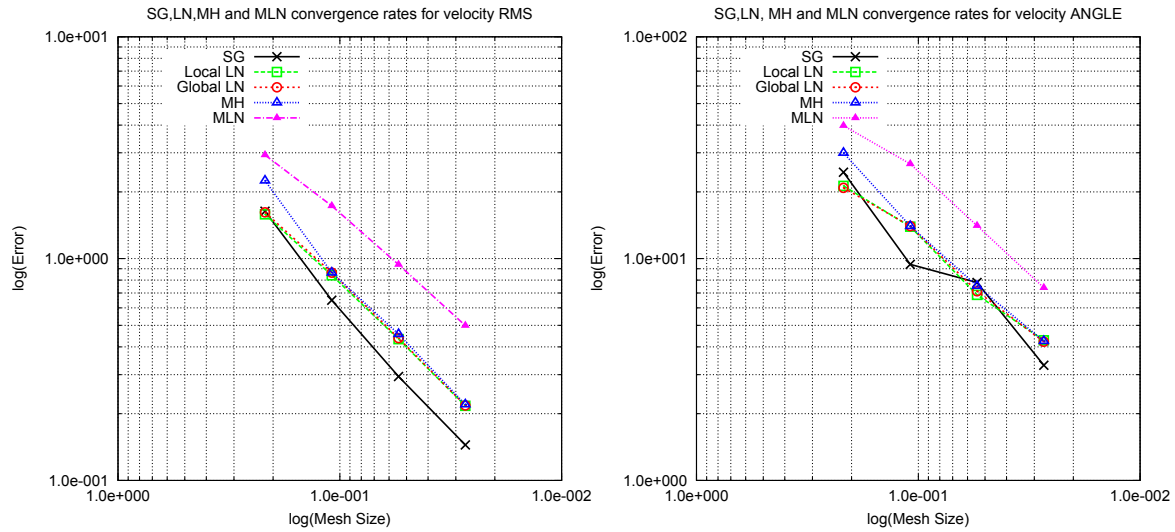


Figure 5.12: Convergence profiles for velocity magnitudes (a) and angles (b).

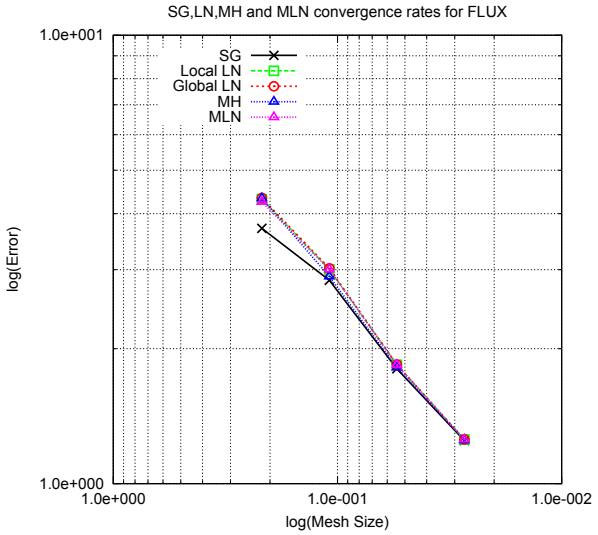


Figure 5.13: Convergence profiles for edge fluxes.

Table 5.1 reports all relevant mesh statistics, while Tables 5.2 and 5.3 show values of $\mathcal{E}$ for element velocities. Error estimates for velocity angles are summarized in Tables 5.6 and 5.7. The proposed MLN method produces higher errors, while convergence rates are very similar compared to the other schemes. We stress that, altough larger errors are produced respect to the LN correction, velocities are consistent with large diffusivity gradients and do not enter in areas of negligible diffusivity. Higher errors are therefore compensated by improved advection properties of the MLN velocities. All velocities used in the error estimation above are inevitably affected by interpolation. Therefore, convergence on edge fluxes is also investigated, as it is independent on interpolation and shown in Tables 5.4 and 5.5. Analytical fluxes are evaluated by numerical integration of exact normal velocities using a trapezoidal rule and a relative tolerance equal to 1.0×10^{-8}. All methods show very similar errors and convergence rates according to edge fluxes. It can be deduced how the modified RT_0 interpolation strategy is responsible for a large part of the errors in the velocities computed with MLN.

5.5 Conclusions

This study focuses on applications of the Larson-Niklasson post-processing algorithm, to the solution of the diffusion equation. This technique is used to post-process P_1 Galerkin velocities restoring elementwise conservativeness; it can be used both in 2D and 3D, with conforming and not conforming meshes. The performance of the LN scheme is investigated for situations where large diffusivity gradients are specified throughout the computational domain. Two-dimensional streamlines are used to provide graphical intuition on the quality of the velocity field produced by various schemes. Both Local and Global formulations are investigated for the LN approach, while a Mixed-Hybrid finite element implementation is used as a reference. Using numerical benchmarks, we show that velocities in areas where large diffusivity gradients and streamlines curvature occur are overestimated by LN, if compared to P_1 Galerkin and MH approximations. We therefore propose a modified strategy (MLN), where LN velocities are compared with P_1 Galerkin estimates; this identifies areas where a further correction is needed. Modified algebraic systems are formulated for element stars exhibiting excessive errors and local P_1 Galerkin fluxes are restored using Lagrange multipliers. Finally, a modified RT0 interpolation scheme is used to compute element velocities for cases where additional source terms are present. The convergence properties of all schemes (P_1 Galerkin, Local LN, Global LN, MH, and MLN) are assessed both for velocities and fluxes, resulting in similar convergence rates. The proposed approach restores physically meaningful velocities in areas of large diffusivity gradients and prevents streamlines for entering subdomains of negligible diffusivity.

Chapter 6

Conclusion

A novel framework for non-intrusive Uncertainty Propagation is proposed in this work.

It combines the ability of a multiresolution approximation in capturing piecewise smooth stochastic responses of physical systems, with the recent advances in signal processing and compression. Using greedy recovery algorithms, responses are reconstructed keeping to a minimum the number of deterministic solutions needed. Importance driven sampling strategies are also applied to the proposed framework in order to further improve converge rates for responses exhibiting sharp gradients and even discontinuities.

It has been also applied to a number of benchmark problems as well as to application of passive control of dynamical systems under uncertainty. The resulting convergence rates to first-order and second-order statistics have been compared to more traditional approaches, resulting in very promising results.

However, further research is required with regards to the size of the Alpert basis for high dimensional stochastic problems together with more efficient algorithms to compute the reconstruction coefficients able to benefit from implementations on massively parallel architectures as well as capable of exploiting the incremental nature of the adaptive sampling process. The highly coherent nature of the adopted multiresolution basis set is also of concern as it affects the convergence to a unique sparse representation of a given stochastic response.

Bibliography

[1] R. Abgrall, P.M. Congedo, C. Corre, S. Galera, et al. A simple semi-intrusive method for uncertainty quantification of shocked flows, comparison with a non-intrusive polynomial chaos method. 2010.

[2] N. Agarwal and NR Aluru. A domain adaptive stochastic collocation approach for analysis of mems under uncertainties. *Journal of Computational Physics*, 228(20):7662–7688, 2009.

[3] M. Ainsworth and T.J. Oden. A procedure for a posteriori error estimation for h-p finite element methods. *Computer Methods Appl. Mech. Engr.*, 101:73–96, 1992.

[4] M. Ainsworth and T.J. Oden. A posteriori error estimation for second order elliptic systems. part 2. an optimal order process for calculating self equilibrating fluxes. *Computer Math. Appl.*, 26:75–87, 1993.

[5] M. Ainsworth and T.J. Oden. A unified approach to a posteriori error estimation using element residual methods. *Numer. Math.*, 65:23–50, 1993.

[6] B. Alpert, G. Beylkin, R. Coifman, and V. Rokhlin. Wavelet-like bases for the fast solution of second-kind integral equations. *SIAM Journal on Scientific Computing*, 14:159, 1993.

[7] B. Alpert, G. Beylkin, D. Gines, and L. Vozovoi. Adaptive solution of partial differential equations in multiwavelet bases. *Journal of Computational Physics*, 182(1):149–190, 2002.

[8] B.K. Alpert. A class of bases in l^2 for the sparse representation of integral operators. *Siam J. Math. Anal*, 24:246, 1993.

[9] B.K. Alpert, J. Mohamed, et al. Wavelet and other bases for fast numerical linear algebra. 1992.

[10] R. Askey and J.A. Wilson. *Some basic hypergeometric orthogonal polynomials that generalize Jacobi polynomials*. Number 54-319. Amer Mathematical Society, 1985.

[11] I. Babuška, F. Nobile, and R. Tempone. A stochastic collocation method for elliptic partial differential equations with random input data. *Siam J. Numer. Anal*, 45(3):1005–1034, 2007.

[12] I. Babuška, F. Nobile, and R. Tempone. A stochastic collocation method for elliptic partial differential equations with random input data. *Siam J. Numer. Anal*, 45(3):1005–1034, 2007.

[13] I. Babuška, R. Tempone, and G.E. Zouraris. Galerkin finite element approximations of stochastic elliptic partial differential equations. *SIAM Journal on Numerical Analysis*, pages 800–825, 2005.

[14] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[15] L. Bergamaschi and M. Putti. Mixed finite elements and newton-type linearizations for the solution of richards' equation. *Int. J. Meth. Engng.*, 45:1025–1046, 1999.

[16] M. Bieri, R. Andreev, and C. Schwab. Sparse tensor discretization of elliptic spdes. *SIAM J. Sci. Comput*, 31(6):4281–4304, 2009.

[17] P. Boufounos, M.F. Duarte, and R.G. Baraniuk. Sparse signal reconstruction from noisy compressive measurements using cross validation. In *Statistical Signal Processing, 2007. SSP'07. IEEE/SP 14th Workshop on*, pages 299–303. IEEE, 2007.

[18] F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods*. Springer, Berlin, 1991.

[19] 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), 2009.

[20] R. H. Cameron and W. T. Martin. The orthogonal development of non-linear functionals in series of fourier-hermite functionals. *Annals of Mathematics*, 48(2):pp. 385–392, 1947.

[21] E.J. Candes and T. Tao. Decoding by linear programming. *Information Theory, IEEE Transactions on*, 51(12):4203–4215, 2005.

[22] E.J. Candes, M.B. Wakin, and S.P. Boyd. Enhancing sparsity by reweighted l 1 minimization. *Journal of Fourier Analysis and Applications*, 14(5):877–905, 2008.

[23] T. Chantrasmi, A. Doostan, and G. Iaccarino. Padé–legendre approximants for uncertainty analysis with discontinuous response surfaces. *Journal of Computational Physics*, 228(19):7159–7180, 2009.

[24] R. Chartrand and W. Yin. Iteratively reweighted algorithms for compressive sensing. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pages 3869–3872. IEEE, 2008.

[25] S.S. Chen, D.L. Donoho, and M.A. Saunders. Atomic decomposition by basis pursuit. *SIAM review*, pages 129–159, 2001.

[26] M. Chugunova and D. Pelinovsky. On the uniform convergence of the chebyshev interpolants for solitons. *Mathematics and Computers in Simulation*, 80(4):794–803, 2009.

[27] C.W. Clenshaw and A.R. Curtis. A method for numerical integration on an automatic computer. *Numerische Mathematik*, 2(1):197–205, 1960.

[28] P.G. Constantine, D.F. Gleich, and G. Iaccarino. Spectral methods for parameterized matrix equations. *Arxiv preprint arXiv:0904.2040*, 2009.

[29] M. Cotronei, L.B. Montefusco, and L. Puccio. Multiwavelet analysis and signal processing. *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, 45(8):970–987, 1998.

[30] M.S. Crouse, R.D. Nowak, and R.G. Baraniuk. Wavelet-based statistical signal processing using hidden markov models. *Signal Processing, IEEE Transactions on*, 46(4):886–902, 1998.

[31] G.M. Davis, V. Strela, and R. Turcajová. Multiwavelet construction via the lifting scheme. *LECTURE NOTES IN PURE AND APPLIED MATHEMATICS*, pages 57–80, 2000.

[32] R. DeVore, I. Daubechies, M. Fornasier, and C.S. Gunturk. Iteratively re-weighted least squares minimization for sparse recovery, 2008.

[33] D. Donoho and J. Tanner. Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1906):4273–4293, 2009.

[34] D.L. Donoho. Compressed sensing. *Information Theory, IEEE Transactions on*, 52(4):1289–1306, 2006.

[35] D.L. Donoho, I. Drori, Y. Tsaig, and J.L. Starck. Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit. 2006.

[36] D.L. Donoho, N. Dyn, D. Levin, and T.P.Y. Yu. Smooth multiwavelet duals of alpert bases by moment-interpolating refinement. *Applied and Computational Harmonic Analysis*, 9(2):166–203, 2000.

[37] D.L. Donoho, A. Maleki, and A. Montanari. Message-passing algorithms for compressed sensing. *Proceedings of the National Academy of Sciences*, 106(45):18914, 2009.

[38] A. Doostan and G. Iaccarino. A least-squares approximation of partial differential equations with high-dimensional random inputs. *Journal of Computational Physics*, 228(12):4332–4345, 2009.

[39] A. Doostan and H. Owhadi. A non-adapted sparse approximation of pdes with stochastic inputs. *Journal of Computational Physics*, 230(8):3015 – 3034, 2011.

[40] M.F. Duarte, M.B. Wakin, and R.G. Baraniuk. Fast reconstruction of piecewise smooth signals from incoherent projections. *SPARS'05*, 2005.

[41] M.F. Duarte, M.B. Wakin, and R.G. Baraniuk. Wavelet-domain compressive signal reconstruction using a hidden markov tree model. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pages 5137–5140. Ieee, 2008.

[42] M. Elad. *Sparse and redundant representations: from theory to applications in signal and image processing*. Springer Verlag, 2010.

[43] M.S. Eldred, C.G. Webster, and P. Constantine. Evaluation of non-intrusive approaches for wiener-askey generalized polynomial chaos. In *Proceedings of the 10th AIAA Non-Deterministic Approaches Conference, number AIAA-2008-1892, Schaumburg, IL*, 2008.

[44] D. Fong and M. Saunders. Lsmr: An iterative algorithm for sparse least-squares problems. *Arxiv preprint arXiv:1006.0758*, 2010.

[45] M. Fornasier and H. Rauhut. Compressive sensing. *Handbook of Mathematical Methods in Imaging. Springer, to appear*, 2010.

[46] W. Fraser and MW Wilson. Remarks on the clenshaw-curtis quadrature scheme. *SIAM Review*, 8(3):322–327, 1966.

[47] G. Gambolati. *Lezioni di Metodi Numerici*. Edizioni libreria Cortina, Padova, 1994.

[48] R.G. Ghanem and P.D. Spanos. *Stochastic finite elements: a spectral approach*. Dover Pubns, 2003.

[49] D. Gottlieb and D. Xiu. Galerkin method for wave equations with uncertain coefficients. *Commun. Comput. Phys*, 3(2):505–518, 2008.

[50] A. Grossmann and J. Morlet. Decomposition of hardy functions into square integrable wavelets of constant shape. 1984.

[51] A.C. Hansen. Generalized sampling and infinite dimensional compressed sensing. Technical report, Technical report NA2011/02, DAMTP, University of Cambridge, 2011.

[52] John D. Jakeman, Richard Archibald, and Dongbin Xiu. Characterization of discontinuities in high-dimensional stochastic problems on adaptive sparse grids. *Journal of Computational Physics*, 230(10):3977 – 3997, 2011.

[53] C. Johnson. *Numerical Solutions of Partial Differential Equations by the Finite Element Method*. Cambridge University Press, New York, 1987.

[54] C. E. Kees, M. W. Farthing, and C. N. Dawson. Locally conservative, stabilized finite element methods for variably saturated flow. *Comput. Methods Appl. Mech. Engrg.*, 197:4610–4625, 2008.

[55] F. Keinert. *Wavelets and multiwavelets*. Chapman & Hall/CRC, 2004.

[56] D.W. Kelly. The self-equilibration of residuals and complementary a posteriori error estimates in the finite element method. *Int. J. Numer. Meth. Engng.*, 20:1491–1506, 1984.

[57] R. A Klausen and T. F Russell. Relationships among some locally conservative discretization methods which handle discontinuous coefficients. *Comput. Geosci.*, 8(4):341–377 (2005), 2004.

[58] R.H. Kraichnan. Direct-interaction approximation for a system of several interacting simple shear waves. *Physics of Fluids*, 6:1603, 1963.

[59] C. La and M.N. Do. Signal reconstruction using sparse tree representations. *Proc. Wavelets XI at SPIE Optics and Photonics, San Diego*, 2005.

[60] C. La and M.N. Do. Tree-based orthogonal matching pursuit algorithm for signal reconstruction. In *Image Processing, 2006 IEEE International Conference on*, pages 1277–1280. IEEE, 2006.

[61] M. G. Larson and A. J. Niklasson. A conservative flux for the continuous Galerkin method based on discontinuous enrichment. *Calcolo*, 41(2):65–76, 2004.

[62] O.P. Le Maître and O.M. Knio. *Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics*. Springer Verlag, 2010.

[63] O.P. Le Maıtre, 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.

[64] O.P. Le Maıtre, 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.

[65] X. Ma and N. Zabaras. An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations. *Journal of Computational Physics*, 228(8):3084–3113, 2009.

[66] S.G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 11(7):674–693, 1989.

[67] S.G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *Signal Processing, IEEE Transactions on*, 41(12):3397–3415, 1993.

[68] API manual. *Documentation for the Straus7 application programming interface, Edition 6.a, API release 2.3*. Strand7 Pty Ltd, 2005.

[69] A. Mazzia and M. Putti. High order godunov mixed methods on tetrahedral mesh for density driven flow simulations in porous media. *J. Comput. Phys.*, 208:154–174, 2005.

[70] Annamaria Mazzia. An analysis of monotonicity conditions in the mixed hybrid finite element method on unstructured triangulations. *Int. J. Numer. Meth. Engng*, 76(3):351–375, 2008.

[71] Y. Meyer. Wavelets-algorithms and applications. *Wavelets-Algorithms and applications Society for Industrial and Applied Mathematics Translation., 142 p.*, 1, 1993.

[72] Y. Meyer, D.H. Salinger, and Cambridge University Press. *Wavelets and operators*, volume 2. Cambridge Univ Press, 1992.

[73] D. Needell and J.A. Tropp. Cosamp: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26(3):301–321, 2009.

[74] S.A. Orszag and L.R. Bissonnette. Dynamical properties of truncated wiener-hermite expansions. *Physics of Fluids*, 10:2603, 1967.

[75] K. Petras. Fast calculation of coefficients in the smolyak algorithm. *Numerical Algorithms*, 26(2):93–109, 2001.

[76] M. Putti and C. Cordes. Finite element approximation of the diffusion operator on tetrahedra. *SIAM J. on Scient. and Stat. Comp.*, 19(4):1154–1168, 1998.

[77] M. Putti and F. Sartoretto. Linear galerkin vs mixed finite element 2d flow fields. *Int. J. Numer. Meth. Fluids*, 2008.

[78] Alfio Quarteroni and Alberto Valli. *Numerical Approximation of Partial Differential Equations*, volume 23 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1994.

[79] H. Rauhut and R. Ward. Sparse legendre expansions via l1-minimization. *J. Approx. Theory*, 164(5):517–533, May 2012.

[80] P.A. Raviart and J.M. Thomas. A mixed finite element method for second order elliptic problems. *Mathematical Aspects of the Finite Element Method*, 606, 1977. Galliani I., Magenes E. (eds), Springer: Berlin, New York.

[81] F. Simon, P. Guillen, P. Sagaut, and D. Lucor. A gpc-based approach to uncertain transonic aerodynamics. *Computer Methods in Applied Mechanics and Engineering*, 199(17-20):1091–1099, 2010.

[82] SA Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. In *Dokl. Akad. Nauk SSSR*, volume 4, page 111, 1963.

[83] H. Sobieczky. Parametric airfoils and wings. *Notes on Numerical Fluid Mechanics*, 68:71–88, 1998.

[84] J.L. Starck, F. Murtagh, and J.M. Fadili. *Sparse image and signal processing: wavelets, curvelets, morphological diversity*. Cambridge Univ Pr, 2010.

[85] V. Strela. *Multiwavelets: Theory and applications*. PhD thesis, Citeseer, 1996.

[86] R.J. Tibshirani and J. Taylor. The solution path of the generalized lasso. *The Annals of Statistics*, 39(3):1335–1371, 2011.

[87] L.N. Trefethen. Is gauss quadrature better than clenshaw-curtis? *SIAM review*, 50(1):67–87, 2008.

[88] J.A. Tropp and A.C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *Information Theory, IEEE Transactions on*, 53(12):4655–4666, 2007.

[89] J. Tryoen, O. Le Maitre, A. Ern, et al. Adaptive anisotropic spectral stochastic methods for uncertain scalar conservation laws. 2012.

[90] 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.

[91] E. van den Berg and M.P. Friedlander. Sparse optimization with least-squares constraints. Technical report, Technical Report, University of British Columbia, Vancouver, 2010.

[92] J. Waldvogel. Fast construction of the fejér and clenshaw–curtis quadrature rules. *BIT Numerical Mathematics*, 46(1):195–202, 2006.

[93] X. Wan and G.E. Karniadakis. An adaptive multi-element generalized polynomial chaos method for stochastic differential equations. *Journal of Computational Physics*, 209(2):617–642, 2005.

[94] X. Wan and G.E. Karniadakis. Multi-element generalized polynomial chaos for arbitrary probability measures. *SIAM Journal on Scientific Computing*, 28(3):901–928, 2007.

[95] N. Wiener. The homogeneous chaos. *American Journal of Mathematics*, 60(4):897–936, 1938.

[96] J.A.S. Witteveen and G. Iaccarino. Simplex stochastic collocation with random sampling and extrapolation for nonhypercube probability spaces. *SIAM Journal on Scientific Computing*, 34:A814, 2012.

[97] D. Xiu and J.S. Hesthaven. High-order collocation methods for differential equations with random inputs. *SIAM Journal on Scientific Computing*, 27(3):1118, 2006.

[98] D. Xiu and G.E. Karniadakis. The wiener–askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.*, 24(2):619–644, February 2002.

[99] Dongbin Xiu and George Em Karniadakis. The wiener–askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.*, 24(2):619–644, 2002.

[100] T. Zhou and T. Tang. Galerkin methods for stochastic hyperbolic problems using bi-orthogonal polynomials. *Journal of Scientific Computing*, pages 1–19, 2011.