Model order reduction techniques for stochastic problems

Mohamed Raed Blel

École des Ponts
ParisTech

THÈSE DE DOCTORAT
de l'École des Ponts ParisTech

Model Order Reduction for Stochastic Problems: Reduced Basis and Dynamical Orthogonal methods

École doctorale MSTIC N°532

Mathématiques

Thèse préparée au Centre d'Enseignement et de Recherche en MathématIques et Calcul Scientifique, CERMICS

Thèse soutenue le 01 Juin 2022, par

Mohamed Raed Blel

Tony LELIÈVRE	École nationale des ponts et chaussées	Directeur de thèse
Virginie EHRLACHER	École nationale des ponts et chaussées	Co-directrice de thèse
Olivier ZAHM	Inria	Examinateur
Marie BILLAUD-FRIESS	Ecole centrale de Nantes	Examinatrice
Clémentine PRIEUR	Université Grenoble Alpes	Rapporteure
Guillaume PERRIN	Université Gustave Eiffel	Examinateur
Fabien CASENAVE	Safran	Examinateur
Julien SALOMON	Sorbonne Université, Lab. J.-L. Lions	Rapporteur
Abdellah CHKIFA	Université Polytechnique Mohamed 6	Invité

REMERCIEMENTS

La plage de la Marsa en méditerranée, dans la banlieue de Tunis,

J'avais neuf ans et j'étais avec mon père. Je voyais de loin des jeunes revenir de leur baignade, les mains chargées d'oursins. Je me suis précipité tout de suite vers les rochers d'où ils sortaient. C'était un très beau spectacle de la nature pour le gamin que j'étais. Il y avait plusieurs petits poissons et pleins d'autres bêtes. Je faisais en sorte de revenir plusieurs fois à cet endroit, jusqu'au jour ou mon père m'avait dit: "Alors fiston, tu n'arrives toujours pas à en attraper?", j'ai répondu au quart de tour : "J'y arriverai papa, ne t'en fais pas, j'y arriverai !". N'ayant pas les outils nécessaires de pêche, j'ai du improviser et faire avec les moyens du bord. J'ai réussi à fabriquer une lance rudimentaire avec un bout de bois et un couteau à l'extrémité. J'ai passé des heures et des heures de mon enfance à contempler la mer et l'admirer, à essayer de déchiffrer ses secrets profonds mais aussi à réfléchir sur la technique à suivre pour attraper un poisson. Le temps semblait s'arrêter, il fallait que j'y arrive. A la fin je n'ai pas réussi ! Je sais que je parviendrai à le faire un jour. De ces temps insoucieux de mon enfance passé à me prendre pour le capitaine Nemo, j'ai conservé mon sens de l'observation, de l'analyse et

de recherche de solutions. J'ai eu la chance d'affiner ces compétences et les mettre à profit lors de mes 3 ans de thèse de doctorat au CERMICS à l'école des ponts et chaussées. Et quelle chance et honneur de pouvoir le faire en compagnie de deux personnes magnifiques, mes encadrants, Tony et Virginie. Le doctorat aura été un beau chapitre de ma vie. Très dur à gérer psychologiquement, il m'en aura fait voir de toutes les couleurs. Mais je pense que c'est la que réside tout le plaisir qu'il procure, un peu comme un fruit après l'avoir épluché. Ce fut une étape pénible et laborieuse certes mais dans tout ca, ce qui est important à mon avis, résultats et objectifs à part, c'est la volonté, la persévérance et la confiance en soi, bien qu'ils soient mis à l'épreuve à tout moment. A mon humble avis, tout ce qu'on rêve d'atteindre est atteignable avec ces trois points. En gardant la barre haute, le succès en est une conséquence.

En ce 1er jour du mois de Juin de l'année 2022, avec la même ténacité du Raed de 9 ans, je concrétise enfin un objectif qui m'a tenu à cœur pendant ces trois années et demi et ceci n'aurait jamais eu lieu sans le soutien indéfectible de mes chers...

A Mes encadrants, Tony et Virginie, et quels encadrants ! Vous êtes juste magnifiques, je ne peux pas demander plus au bon dieu que d'être encadré par des personnes aussi compétentes et aussi humaines que vous. Certes, c'était dur de concilier vos avis (des fois contraires) mais comme me disait souvent Tony: "c'est l'avantage d'avoir deux encadrants". Merci à vous deux. A Tony, j'ai toujours admiré ta façon d'expliquer et ta finesse d'interprétation, tes conseils je les garderai pour toujours: "Raed! si tu bégaies une fois c'est que tu ne l'as pas assimilé!". Mention spéciale pour ta punch line si simple :" Raed! je n'ai rien compris" alors que je venais de finir dix

minutes de démonstration... Grâce à toi, j'ai appris ce que c'était d'être rigoureux et de bien structurer mes réflexions. A Virginie, merci pour ta douceur exceptionnelle qui m'a rendu à l'aise même à chaque fois où je n'y parvenais pas, tu étais toujours la malgré tes responsabilités, et je t'en suis infiniment reconnaissant.

Je tiens à remercier chaleureusement Clémentine Prieur et Julien Salomon pour avoir accepté d'être rapporteurs de ma thèse, je remercie aussi les membres du jury Olivier Zahm, Guillaume Perrin, Fabien Casenave, Marie Billaud-Friess et Abdellah Chkiffa pour m'avoir honoré de leurs présences, leurs salutations et encouragements vis-à-vis de mes travaux. Merci beaucoup.

A mes professeurs, de l'école primaire jusqu'au master, "sti" Habiba avec son son batton d'olivier "Massouda", madame Bacha et "si" Chekir avec qui le plaisir de faires des maths a commencé et avec qui je ne ratait aucun match de foot au stade El Menzah, Madame Saadoun pour les heures de rattrapage le dimanche matin pendant le Bac, à Monsieur Attia avec qui la prépa est devenue une leçon de vie et pour finir, à Sonia Fliss avec qui j'ai passé une année de master inoubliable et qui a eu le bon flair en m'envoyant la proposition de thèse. Merci infiniment à vous tous.

A mes amis, Tarek, Piko, Toutou, Yassine, Faten, Skander, Omar, Iheb et ceux qui m'ont apporté de la joie de vivre dans ce chemin sinueux qu'est la vie, à Mahran pour les heures de discussions passionnantes qu'on a passé ensemble, à mon cher Nour "L'encyclopédie en personne" et "mon éternel bras droit" pour son soutien infaillible pendant le combat psychologique que nous avons mené pendant le doctorat et que nous menons encore dans les batailles de la vie. Merci

à vous tous, chacun à sa façon a su me fournir un peu d'essence pour aller de l'avant.

A ma famille, mon frére Rami, ma soeur Manel, mon beau neuveu Rafa que j'aime tellement, mes cousins Baligh, Nizar et Abir et à toute la famille Blel et Rais. Merci beaucoup.

A mon père qui a incarné en moi l'envie de faire des maths, de lire et de découvrir... Merci cher père.

A celle qui m'a toujours porté en elle, bien plus que les 9 mois de gestation, et pour laquelle je ne trouverai jamais un terme précis pour exprimer mon amour et ma gratitude, je ne peux tout simplement pas de demander de mieux au bon dieu, à toi "moma".

Et il n'y pas mieux pour conclure ces remerciements qu'en dialecte tunisien: « Rabbi ifadhlek w ifadhel naboula w ifadhel kol mère et père w nchalah rabbi yarhem mawtena ».

Contents

CHAPTER I

INTRODUCTION

L'objet de ce chapitre est de donner une présentation succinte des enjeux liés aux méthodes de réduction de modèles dans des contextes stochastiques qui ont été abordés au cours de cette thèse. La Section I.1 présente une brève introduction aux méthodes de réduction de modèles les plus classiques utilisées dans des contextes déterministes. Les méthodes les plus classiquement utilisées pour accélérer des études paramétriques dans des contextes déterministes sont présentées dans la Section I.2. La Section I.3 présente les principales techniques de réduction de variance utilisées pour accélérer le calcul d'espérance par des méthodes de Monte-Carlo. Enfin, la Section I.4 présente les principales contributions de cette thèse.

I.1 Introduction à la réduction de modèles

Cette section contient une brève introduction aux enjeux de la réduction de modèle, suivie d'une section illustrant la malédiction de la dimension sur quelques exemples de problèmes déterministes ou stochastiques. Ensuite, quelques méthodes classiques pour la réduction de modèle sont présentées.

I.1.1 Motivation

Dans de nombreux domaines industriels ou financiers, des modèles mathématiques sont utilisés pour prédire les valeurs de certaines quantités caractérisant l'état d'un système d'intérêt. Ces modèles sont ensuite simulés numériquement pour obtenir des approximations de ces quantités d'intérêt. Ces modèles s'écrivent sous forme de systèmes d'équations aux dérivées partielles, ou d'équations différentielles stochastiques, et les quantités d'intérêt du système considéré peuvent souvent être calculées explicitement en fonction de la solution du modèle. Plusieurs paramètres interviennent souvent dans la définition du modèle mathématique. En conséquence, la solution du modèle dépend elle aussi de la valeur de ces paramètres.

Dans de nombreux contextes applicatifs, tels que la calibration de modèles, l'optimisation, l'analyse de sensibilité ou la quantification d'incertitudes, il est nécessaire de calculer la solution de ces modèles pour un très grand nombre de valeurs de ces paramètres, ce qui peut devenir extrêmement coûteux en termes de temps de calcul si de telles méthodes de calcul sont effectuées

avec des méthodes naïves. L'objectif d'une méthode de réduction de modèles est de proposer un algorithme dans le but d'accélérer de manière significative de telles études paramétriques.

Avant de présenter le principe général des méthodes de réduction de modèles, nous présentons tout d'abord le problème de la dimension ci-dessous, suivi par quelques exemples de modèles paramétriques d'intérêt dans le cadre de cette thèse. Dans toute la suite, nous supposons que le modèle mathématique dépend d'un vecteur de paramètres noté $\mu \in \mathcal{P}$ où l'ensemble $\mathcal{P} \subset \mathbb{R}^{d_p}$ (avec $d_p \in \mathbb{N}^*$) représente l'ensemble des valeurs possibles de ce vecteur. Nous noterons également $\mathcal{M}$ l'ensemble des solutions des problèmes (déterministes ou stochastiques) paramétrés considéré et supposerons que $\mathcal{M}$ est un sous-ensemble d'un espace de Hilbert V. L'ensemble des solutions $\mathcal{M}$ est explicité dans chacun des exemples donnés ci-dessous.
Le paragraphe suivant illustre le problème de la dimension dans le cas d'une résolution standard.

I.1.2 Sur la malédiction de la dimension

En pratique, les techniques de résolution standard ne peuvent pas résoudre les modèles en grande dimension ou faisant intervenir plusieurs paramètres [23]. Pour bien expliquer ce problème, prenons un domaine $[0,1]^d$, avec $d \in \mathbb{N}^*$, et une fonction $f : [0,1]^d \to \mathbb{R}$ de classe $\mathcal{C}^m$ pour $m \in \mathbb{N}^*$. Supposons qu'on veuille reconstruire f à partir de N_h valeurs $\{f(x_i)\}_{1\leq i\leq N_h}$ où $x_1, \cdots, x_{N_h} \in [0,1]^d$ et $N_h \in \mathbb{N}^*$. Alors depuis [24] on a,

Proposition I.1.1. *Soit $d \in \mathbb{N}^*$, pour $f : [0,1]^d \to \mathbb{R}$ une fonction de classe $\mathcal{C}^m$ avec $m \in \mathbb{N}^*$ et la famille $(x_i)_{1\leq i\leq N_h}$ une discrétisation uniforme de $[0,1]^d$ avec un pas h.*
Alors pour toute approximation polynomiale $P(f)$ de f on a,

$$\|f - P(f)\|_{L^\infty(\Omega)} \leq Ch^m,$$

avec $C > 0$ une constante indépendante de h. Sachant que le nombre de points N_h est de l'ordre de h^{-d}, alors l'erreur d'approximation en N_h est:

$$\|f - P(f)\|_{L^\infty(\Omega)} \leq CN_h^{-m/d}.$$

Ainsi, la vitesse de décroissance de l'erreur diminue en augmentant la dimension.
Il est même prouvé dans [24] qu'il est impossible de trouver une reconstruction qui atteigne un meilleur résultat.
Ceci peut être expliqué en introduisant la largeur non linéaire (N_h-*width*). Soit L un espace normé, $\|\cdot\|_L$ sa norme, et $K \subset L$. Considérons l'application continue $E : K \to \mathbb{R}^{N_h}$ (*encodage*) et $R : \mathbb{R}^{N_h} \to L$ (*reconstruction*). La distorsion de la paire (E, R) dans K est définie comme:

$$\sup_{f\in K} \|f - R(E(f))\|_L .$$

Cette erreur représente la pire erreur sur toutes les fonctions $f \in K$ par le schéma d'encodage-reconstruction. La largeur N_h-*width* de K est définie comme l'infinimum de la distorsion sur toutes les pairs de l'application continue (E, R):

$$d_{N_h}(K)_L := \inf_{\left\{\begin{array}{c} E : K \to \mathbb{R}^{N_h} \\ R : \mathbb{R}^{N_h} \to L \end{array}\right. \text{continues}} \sup_{f\in K} \|f - R(E(f))\|_L .$$

Le théorème suivant montre que ce $d_{N_h}(K)_L$ est équivalent à ${N_h}^{-\frac{m}{d}}$ pour un L et K spécifiques.

Theorem I.1.2. *[24] Soit $d_{N_h}(K)_{L^\infty([0,1]^d)}$ la largeur N_h-width de K dans $L^\infty([0,1]^d)$ tel que K est défini par,*

$$K = \left\{ f \in \mathcal{C}^m([0,1]^d) \mid \forall \alpha \in \mathbb{N}^d,\ |\alpha| \leq m, \|\partial^\alpha f\|_{L^\infty([0,1]^d)} \leq 1 \right\}$$

est la boule unité de $\mathcal{C}^m([0,1]^d)$ dans une norme appropriée.
Alors il existe $c, C > 0$ indépendants de d tel que pour tout $N_h \in \mathbb{N}^*$,

$$cN_h^{-m/d} \leq d_{N_h}(K)_{L^\infty([0,1]^d)} \leq CN_h^{-m/d}.$$

Ainsi pour approximer une fonction $f \in \mathcal{C}^m([0,1]^d)$ tel que l'erreur soit plus petite qu'une erreur fixée, nécessairement le nombre de noeuds N_h du maillage varie exponentiellement en fonction de d. Ce qui explique le coût prohibitif des problèmes en grande dimension en utilisant les méthodes standards.

Dans ce qui suit nous présentons quelques problèmes dont la résolution souffre de la malédiction de la dimension.

I.1.3 Exemples de problèmes paramétriques

Exemple 1: Problème elliptique (Équation de diffusion) Un premier exemple de problème défini par un système d'équations aux dérivées partielles paramétrique déterministe est le suivant.
Soit Ω un ouvert borné et régulier de $\mathbb{R}^d$ pour $d \in \mathbb{N}^*$ et f une fonction de $L^2(\Omega)$. Pour tout $\mu \in \mathcal{P}$, $A^\mu : \Omega \to \mathbb{R}$ est une fonction mesurable et telle qu'il existe $\alpha, \beta > 0$ tels que

$$\forall \mu \in \mathcal{P}, \quad \alpha \leq A^\mu \leq \beta \quad \text{ presque partout sur } \Omega.$$

Grâce au théorème de Lax-Milgram, il est aisé de vérifier que, pour tout $\mu \in \mathcal{P}$, il existe une unique solution $u^\mu \in H_0^1(\Omega)$ au problème suivant:

$$\begin{cases} -\operatorname{div}\left[A^\mu \nabla_x u^\mu\right] = f & \text{dans } \Omega, \\ u^\mu = 0 & \text{sur } \partial\Omega \end{cases} \tag{I.1}$$

La solution u^μ de ce système représente le champ de température à l'équilibre d'un matériau occupant le domaine Ω en présence de sources de chaleur volumiques f. Le vecteur de paramètres μ peut intervenir par exemple dans la définition des propriétés de diffusion thermique du matériau au sein du domaine Ω.

Notons que (I.1) est équivalent à: trouver $u^\mu \in V$ solution de

$$a^\mu(u^\mu, v) = l(v), \quad \forall v \in V, \tag{I.2}$$

où $V := H_0^1(\Omega)$, $l : V \to \mathbb{R}$ est une forme linéaire continue définie par

$$l(v) = \int_\Omega fv, \quad \forall v \in V,$$

et tel que pour tout $\mu \in \mathcal{P}$, $a^\mu : V \times V \to \mathbb{R}$ est forme bilinéaire coércive continue définie par

$$\forall v, w \in V, \quad a^\mu(v, w) := \int_\Omega \nabla v \cdot A^\mu \nabla w.$$

L'ensemble des solutions de ce système d'équations aux dérivées partielles paramétriques est alors défini comme

$$\mathcal{M} := \{u^\mu,\ \mu \in \mathcal{P}\}$$

et est un sous-ensemble de l'espace de Hilbert $V := H_0^1(\Omega)$.

Notons par ailleurs que le calcul de la solution u^μ pour chaque valeur de $\mu \in \mathcal{P}$ peut être extrêmement coûteux si le problème (I.1) est résolu par une méthode d'éléments finis avec un très grand nombre de degrés de liberté.

Exemple 2: Dynamique de Langevin suramortie Citons ici un deuxième exemple motivé par des applications en dynamique moléculaire. Considérons un système composé de N atomes en dimension $d \in \mathbb{N}^*$. Pour tout $\mu \in \mathcal{P}$, soit $V^\mu : \mathbb{R}^{dN} \to \mathbb{R}$ une fonction régulière représentant un potentiel d'interaction qui à une configuration du système moléculaire associe son énergie. Plus précisément, si pour tout $1 \leq i \leq N$, $X_i \in \mathbb{R}^d$, $V^\mu(X_1, \cdots, X_N)$ représente l'énergie de la molécule dans le cas où l'atome i est situé à la position X_i pour tout $1 \leq i \leq N$. En pratique, pour une molécule donnée, le potentiel d'interaction exact n'est pas connu et dans de nombreux modèles celui-ci est donné comme une fonction dépendant de paramètres empiriques $\mu \in \mathcal{P}$ dont les valeurs exactes pour un système moléculaire donné sont à déterminer.

Soit $(\Omega, \mathcal{F}, \mathbb{P})$ un espace de probabilités. Étant donné ce potentiel d'interaction V^μ, de nombreuses quantités macroscopiques d'intérêt, appelées observables, pour ce système moléculaire sont obtenues comme des moyennes statistiques des quantités d'intérêt calculées à partir de la solution $X_t^\mu \in \mathbb{R}^{dN}$ de l'équation différentielle stochastique suivante, appelée équation de Langevin suramortie:

$$dX_t^\mu = -\nabla V^\mu(X_t^\mu)dt + \beta dW_t, \tag{I.3}$$

où $\beta > 0$ est propotionnel à la racine carrée de la température W_t est un mouvement brownien Nd-dimensionnel.

Un exemple d'observable d'intérêt est le coefficient de diffusion, obtenu à partir du calcul de la moyenne de la déviation quadratique des particules sur un temps $T > 0$ définie par

$$MSD_\mu^T = \mathbb{E}\left[\int_0^T \frac{(X_{t+T}^\mu - X_t^\mu)^2}{T} dt\right] \tag{I.4}$$

Remarquons que pour chaque valeur du paramètre μ, calculer ce coefficient MSD_μ^T nécesssite d'échantillonner un grand nombre de trajectoires stochastiques pour estimer cette espérance. Le calcul est clairement très coûteux, notamment quand le nombre de particules N dans le système est importante.

L'ensemble des solutions de ce système d'équations différentielles stochastiques paramétriques est alors défini comme

$$\mathcal{M} := \{u^\mu := (X_t^\mu)_{0 \leq t \leq T},\ \mu \in \mathcal{P}\}$$

et est un sous-ensemble de l'espace de Hilbert $V := L^2_{\mathbb{P}}\left(\Omega; L^2([0,T]; \mathbb{R}^{dN})\right)$.

Exemple 3: Équation de Black-Scholes Mentionnons enfin un dernier exemple de système d'équations différentielles stochastiques paramétrique issu d'applications en finance: le modèle de Black-Scholes qui est utilisé pour modéliser le *pricing* d'options. Soit $(\Omega, \mathcal{F}, \mathbb{P})$ un espace de probabilité.

Dans ce contexte, il est nécessaire de calculer l'espérance d'une variable aléatoire Z^μ, définie pour tout $\mu \in \mathcal{P}$ comme:

$$Z^\mu = g^\mu(X_T^\mu) - \int_0^T f^\mu(s, X_s^\mu)ds, \tag{I.5}$$

où $T > 0$, et où $g^\mu : \mathbb{R} \to \mathbb{R}$, $f^\mu : [0,T] \times \mathbb{R}$ sont des fonctions dépendantes du vecteur de paramètres $\mu \in \mathcal{P}$, et où $(X_t^\mu)_{0\leq t\leq T}$ est solution de l'équation différentielle stochastique paramétrique suivante:

$$X_t^\mu = x + \int_0^t b^\mu(s, X_s^\mu)ds + \int_0^t \sigma^\mu(s, X_s^\mu)dW_s, \tag{I.6}$$

avec $b^\mu : [0,T] \times \mathbb{R} \to \mathbb{R}$ et $\sigma^\mu : [0,T] \times \mathbb{R} \to \mathbb{R}$.

Il est alors nécessaire de *calibrer un tel modèle* c'est-à-dire de trouver la valeur du vecteur de paramètres μ qui permette de minimiser la différence entre les prix observés dans le marché et les prix donnés par modèle. Pour ce faire, il est nécessaire de disposer d'un code de calcul numérique qui permette de donner rapidement une approximation de la valeur de $(X_t^\mu)_{0\leq t\leq T}$ pour de nombreuses réalisations aléatoires et de nombreuses valeurs du vecteur de paramètres $\mu \in \mathcal{P}$.

Dans cet exemple, l'ensemble des solutions est alors défini comme

$$\mathcal{M} := \{u^\mu := (X_t^\mu)_{0\leq t\leq T},\ \mu \in \mathcal{P}\}$$

et est un sous-ensemble de l'espace de Hilbert $V := L^2_{\mathbb{P}}(\Omega; L^2([0,T]))$.

I.1.4 Principe général d'une méthode de réduction de modèles

L'objectif d'une méthode de réduction de modèles est de permettre de calculer très rapidement des approximations des solutions des modèles paramétriques considérés $u^\mu \in V$ pour un grand nombre de valeurs de vecteurs de paramètres $\mu \in \mathcal{P}$.

La majorité des techniques de réduction de modèles sont constituées de deux étapes:

- Une première étape dite 'hors ligne' dans laquelle le calcul de la solution exacte u_μ du modèle considéré est effectué pour un certain nombre (si possible petit) de valeurs du vecteur de paramètres $\mu \in \mathcal{P}$ bien choisie.

- Dans une deuxième étape, dite 'en ligne', un modèle approché, dit réduit, est construit à partir du petit nombre de calculs exacts qui ont été effectués lors de la phase hors-ligne. Ce modèle réduit, qui est beaucoup moins coûteux pour n'importe quelle valeur du vecteur de paramètres que le problème initial, est ensuite utilisé pour calculer très rapidement des approximations de la solution u_μ pour un très grand nombre de valeurs du paramètre $\mu \in \mathcal{P}$.

Il est à noter que la plupart des techniques de réduction de modèles ont été développées pour accélérer le temps de calcul de problème paramétriques *déterministes*.

Les contributions exposées dans cette thèse ont pour objectif de proposer de nouvelles méthodes pour la réduction de systèmes d'équations différentielles *stochastiques* paramétriques.

Nous présentons tout d'abord l'état de l'art des principales méthodes de réduction de modèles dans le cadre déterministe avant de détailler les contributions de cette thèse.

I.2 Principales méthodes de réduction de modèles pour des problèmes paramétriques déterministes

Quelques méthodes classiques de réduction de modèles utilisées dans des contextes déterministes sont présentées dans cette section.

I.2.1 Construction d'un espace réduit

Dans la plupart des technique de réduction de modèles, une première étape consiste à déterminer un sous-espace linéaire X_n de V, de dimension finie n (si possible petite), appelé espace réduit, de telle sorte que chaque élément $u_\mu \in \mathcal{M}$ soit bien approché en un certain sens par un élément de X_n. Nous présentons ci-dessous les deux méthodes les plus classiquement utilisées dans des contextes déterministes pour la construction de tels espaces réduits, à savoir la Décomposition Orthogonale Propre et l'algorithme glouton classiquement utilisé dans la méthode des bases réduites. Nous supposons dans le reste de ce chapitre d'introduction que (i) $\mathcal{P}$ est un sous-ensemble borné de $\mathbb{R}^p$ et (ii) $\mathcal{M}$ est un sous-ensemble compact de V.

De plus, dans la suite, pour tout sous-espace vectoriel W de dimension finie de V, nous noterons Π_W l'opérateur de projection orthogonale sur cet espace.

Décomposition Orthogonale Propre

Le but du théorème ci-dessous est de définir une Décomposition Orthogonale Propre (POD) d'une fonction $U \in L^2(\mathcal{P}; V)$. Dans la littérature, une telle décomposition est également appelée décomposition de Karhunen-Loève, ou décomposition en composantes principales.

Theorem I.2.1. *Soit $U \in L^2(\mathcal{P}; V)$. Alors, il existe une base orthonormale de V, notée $(e_i)_{i\in\mathbb{N}^*}$, une base orthonormale de $L^2(\mathcal{P})$, notée $(f_i)_{i\in\mathbb{N}^*}$ et une suite décroissante de réels non-négatifs, notée $(\sigma_i)_{i\in\mathbb{N}^*}$, tels que*

$$\text{pour presque tout } \mu \in \mathcal{P}, \quad U(\mu,\cdot) = \sum_{i\in\mathbb{N}^*} \sigma_i f_i(\mu) e_i.$$

De plus, pour tout $n \in \mathbb{N}^*$, le sous-espace vectoriel $X_n^{\mathrm{POD}} := \mathrm{Span}\{e_1, \cdots, e_n\}$ est solution du problème de minimisation suivant:

$$X_n^{\mathrm{POD}} \in \underset{\substack{V_n \subset V \text{ sous-espace linéaire} \\ \dim V_n = n}}{\operatorname{argmin}} \int_{\mathcal{P}} \|U(\mu,\cdot) - \Pi_{V_n} U(\mu,\cdot)\|_V^2 \, d\mu. \tag{I.7}$$

Enfin, l'identité suivante est vérifiée:

$$\int_{\mathcal{P}} \|U(\mu,\cdot) - \Pi_{X_n^{\mathrm{POD}}} U(\mu,\cdot)\|_V^2 \, d\mu = \sum_{i\geq n+1} \sigma_i^2.$$

Le théorème I.2.1 donne une première manière de définir un espace réduit X_n^{POD} de dimension n pour approcher l'ensemble de solutions $\mathcal{M}$. En effet, en définissant pour tout $\mu \in \mathcal{P}$, $U(\mu,\cdot) := u_\mu$, si la fonction U ainsi définie est bien un élément de $L^2(\mathcal{P}; V)$, alors il est possible

d'appliquer le Théorème I.2.1. L'espace réduit obtenu X_n^{POD} est alors optimal au sens de (I.7), c'est-à-dire en un sens L^2 par rapport au paramètre $\mu \in \mathcal{P}$.

Le calcul de l'espace réduit X_n^{POD} peut être néanmoins très coûteux en terme de temps de calcul. En effet, (i) elle nécessite de connaître les valeurs des solutions u_μ pour presque toutes les valeurs de $\mu \in \mathcal{P}$ (ii) en pratique, elle requiert la résolution d'un problème aux valeurs propres dont le coût computationnel peut être prohibitif.

Mise en oeuvre de la méthode, cas d'un espace de paramètres fini:

Soit V un espace d'Hilbert muni de son produit scalaire $< .,. >_V$: $(V, V) \to \mathbb{R}$ et de la norme associée $||.||_V$.

Considérons $\mathcal{P}_p$ la version discrète de cardinal fini $p = |\mathcal{P}_p|$ de l'espace des paramètres $\mathcal{P}$. Soit la variété $\mathcal{M}(\mathcal{P})$:

$$\mathcal{M}(\mathcal{P}) = \{u_\mu, \mu \in \mathcal{P}\} \subset V$$

Et soit $\mathcal{M}(\mathcal{P}_p)$ une approximation de $\mathcal{M}(\mathcal{P})$:

$$\mathcal{M}(\mathcal{P}_p) = \{u_\mu, \mu \in \mathcal{P}_p\} \subset V$$

Notons que si $\mathcal{P}_p$ est assez fin alors $\mathcal{M}(\mathcal{P}_p)$ est une bonne représentation de $\mathcal{M}(\mathcal{P})$.
Dans une première phase dite d'exploration on génère tous les u_μ pour $\mu \in \mathcal{P}_p$ et ensuite dans une phase de compression, on ne garde que l'information essentielle.
La base POD, X_n^{POD} de dimension n, est telle que,

$$X_n^{POD} \in \underset{V_n \text{de dimension } n}{\arg\inf} \sqrt{\frac{1}{p} \sum_{\mu \in \mathcal{P}_p} ||u_\mu - \Pi_{V_n} u||_V^2} \tag{I.8}$$

où la première minimisation est effectuée sur tous les espaces vectoriels de dimension n de la forme $\text{Span}\{u_{\mu_1}, \cdots, u_{\mu_p}\}$ sous espace de l'espace $V_p = \text{Span}\{u_\mu, \mu \in \mathcal{P}_p\}$. Notons qu'ici, pour simplifier, on affaiblit la forme de (I.8) par rapport à (I.7) . Pour construire la base X_n^{POD} on va utiliser l'opérateur linéaire et symétrique $C : V_p \to V_p$ défini par:

$$C(v) = \frac{1}{p} \sum_{i=1}^{p} \langle v, u_{\mu_i} \rangle_V u_{\mu_i}, \ \forall v \in V_p \tag{I.9}$$

Considérons maintenant les valeurs propres $(\lambda_i)_{1 \le i \le p}$, ordonnées d'une manière décroissante, et les vecteurs propres associés $(\xi_i)_{1 \le i \le p}$ normalisés de C. On a ainsi:

$$\langle C(\xi_i), u_{\mu_j} \rangle_V = \lambda_i \langle \xi_i, u_{\mu_j} \rangle_V, \ 1 \le j \le p, \quad \text{avec} \quad \lambda_i = \frac{\sigma_i^2}{p} \tag{I.10}$$

La base orthogonale $(\xi_i)_{1 \le i \le p}$ est alors une base de p vecteurs qui génère V_p.
On peut alors tronquer cette base et prendre les n premiers vecteurs correspondants aux plus hautes valeurs propres pour avoir $X_n^{POD} = \text{Span}\{\xi_1, \cdots, \xi_n\}$. Cet espace X_n^{POD} réalise alors le minimum du problème (I.8).

Proposition I.2.2. *Soit le projecteur P_n de V dans X_n^{POD} par:*

$$P_n(v) = \sum_{i=1}^{n} < v, \xi_i >_V \xi_i \tag{I.11}$$

L'erreur de la projection P_n si elle est appliquée sur tous les éléments de V_p s'écrit,

$$\sqrt{\frac{1}{p}\sum_{i=1}^{p}||u_{\mu_i} - P_n(u_{\mu_i})||_V^2} = \sqrt{\sum_{i=n+1}^{p}\lambda_i}. \tag{I.12}$$

Cette estimation d'erreur est la seule qu'on a pour le moment pour la méthode POD, ainsi l'absence d'estimateurs d'erreur rend cette technique moins avantagée.
L'algorithme suivant (1) présente en pratique la génération de l'espace $X_n^{POD} = \text{Span}\{\xi_1, \cdots, \xi_n\}$.

Algorithm 1 Algorithme POD

Initialisation: Discrétiser $\mathcal{P}$ en échantillonnant p points μ_i et générer les fonctions u_{μ_i}, pour $1 \leq i \leq p$

Soit la matrice de corrélation $C \in \mathbb{R}^{p\times p}$ tel que:

$$C_{ij} = \frac{1}{p} < u_{\mu_i}, u_{\mu_j} >_V,\ 1 \leq i, j \leq p$$

Résoudre le problème des valeurs propres $(\lambda_i)_{1\leq i\leq p}$ et vecteurs propres normalisés $(\Psi_i)_{1\leq i\leq p}$ de C:

$$C\Psi_i = \lambda_i\Psi_i,\ 1 \leq i \leq p$$

Ce qui est équivalent à (I.10).
Garder les n premiers vecteurs propres $(\Psi_i)_{1\leq i\leq n}$ correspondants aux n plus grandes valeurs propres pour construire l'espace $X_n^{POD} = \text{Span}\{\xi_1, \cdots, \xi_n\}$. avec,

$$\xi_i(x) = \frac{1}{\sqrt{p}}\sum_{j=1}^{p}(\Psi_i)_j u_{\mu_j}(x),\ 1 \leq i \leq n$$

où $(\Psi_i)_j$ est la j-ème composante du vecteur propre Ψ_i.
Output: $X_n^{POD} := \text{Span}\{\xi_1, \cdots, \xi_n\}$

Dans le paragraphe suivant nous présentons la Décomposition en Valeurs Singulières (SVD) d'une matrice et nous faisons le lien avec la POD.

Décomposition en valeurs singulières

Pour construire la base X_n quelques méthodes utilisent la décomposition en valeurs singulières qu'on présente dans cette section.

Theorem I.2.3. *Pour toute matrice réelle $M \in \mathbb{R}^{d\times p}$, il existe une matrice unitaire $\bar{U} \in \mathbb{R}^{d\times d}$ et une matrice unitaire $\bar{V} \in \mathbb{R}^{p\times p}$ ainsi qu'une matrice diagonale $\bar{\Sigma} \in \mathbb{R}^{d\times p}$ tel que ses coefficients sont tous des réels positifs ou nuls pour lesquelles on a:*

$$M = \bar{U}\bar{\Sigma}\bar{V}^T \tag{I.13}$$

Notons que (I.13) se réécrit:

$$M = \sum_{k=1}^{\min(d,p)} \sigma_k U_k V_k^T \tag{I.14}$$

Avec $(U_1, \cdots, U_d)$ les d vecteurs colonnes de la matrice $\bar{U}$ et $(V_1, \cdots, V_p)$ les p vecteurs colonnes de la matrice $\bar{V}$. Cette décomposition n'est pas unique car si on pose $\bar{U} = \bar{\bar{U}}Q$ et $\bar{V} = \bar{\bar{V}}Q$ avec Q une matrice de permutation alors on a

$$M = \bar{\bar{U}}Q\bar{\Sigma}Q^T\bar{\bar{V}}^T \tag{I.15}$$

avec $Q\Sigma Q^T$ une matrice diagonale, et $\bar{\bar{U}}$ et $\bar{\bar{V}}$ deux matrices unitaires.
Ainsi, imposer sur les coefficients de $\bar{\Sigma}$ un ordre décroissant (ce qu'on supposera dans la suite) détermine $\bar{\Sigma}$ d'une façon unique. Ces coefficients sont appelés valeurs singulières de M.
Nous présentons maintenant quelques interprétations de cette décomposition.

Une représentation géométrique:

La décomposition en valeurs singulières a une très belle interprétation géométrique, les matrices unitaires $\bar{U}$ et $\bar{V}^T$ exercent une rotation sur les vecteurs dans l'espace $\mathbb{R}^d$ et $\mathbb{R}^p$ respectivement, alors que la matrice diagonale $\bar{\Sigma}$ effectue une homothétie des vecteurs dépendemment de la valeur singulière ce qui est résumé par la figure I.1.

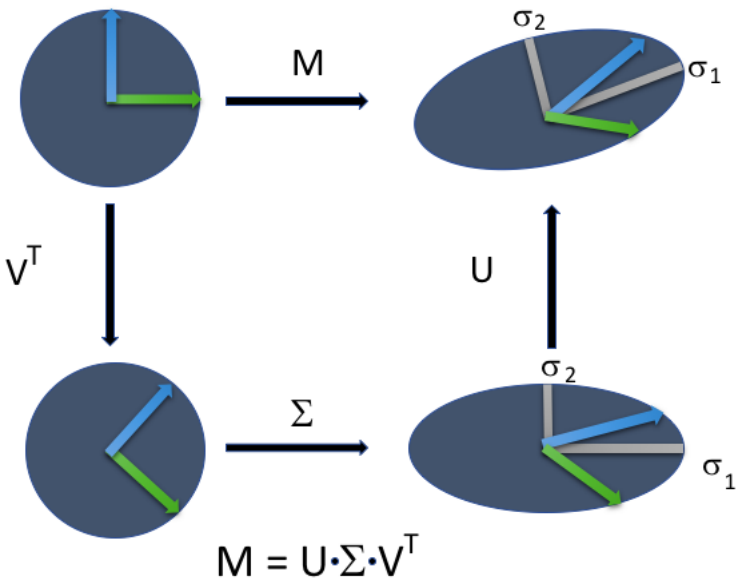


Figure I.1: Représentation géométrique de la SVD dans le cas n=m=2, Wikipédia.

Une interprétation énergétique:

La décomposition en valeurs singulières donne aussi une interprétation énergétique, si M est une matrice qui représente une image en deux dimensions alors la décomposition en SVD fera que les premières colonnes de $\bar{U}$ associées aux plus grandes valeurs singulières contiendront le

maximum d'énergie de l'information existante dans l'image initiale, d'où d'un vecteur singulier à un autre on peut évaluer le taux d'énergie ou de l'information contenue dans l'un relativement à l'autre. Ainsi on peut construire une approximation de l'image initiale en utilisant non plus tous les vecteurs de $\bar{U}$ et de $\bar{V}$ mais ceux associés aux valeurs singulières les plus élevées. D'où la notion de la SVD tronquée.

Décomposition SVD tronquée:

Notons $\mathcal{R}_r$ l'ensemble des matrices de $\mathbb{R}^{d\times p}$ de rang r. La décomposition SVD tronquée de rang r est le tenseur $M_r \in \mathcal{R}_r$ défini par la somme des r premiers termes dans la décomposition SVD:

$$M_r = \sum_{k=1}^{r} \sigma_k U_k V_k^T \tag{I.16}$$

Proposition I.2.4. *Soit M une matrice de $\mathbb{R}^{d\times p}$, alors la décomposition tronquée M_r (I.16) de rang r est une meilleure approximation de M parmi tous les tenseurs de $\mathcal{R}_r$ en utilisant la norme de Frobenius (ou la norme 2 des opérateurs) au sens où:*

$$\inf_{N_r\in\mathcal{R}_r} ||M-N_r||_F^2 = \min_{N_r\in\mathcal{R}_r} ||M-N_r||_F^2 = ||M-M_r||_F^2 = \sum_{i=r+1}^{\min(m,n)} \sigma_i^2 \tag{I.17}$$

avec $(\sigma_i)_{1\leq i\leq \min(d,p)}$ les valeurs singulières de M obtenues par une SVD (I.2.3).

Nous pouvons maintenant discuter l'intérêt de l'approximation SVD tronquée.
Depuis l'écriture $M_r = \sum_{k=1}^{r} \sigma_k U_k V_k^T$ on remarque qu'il suffit de $r + r \times d + r \times p$ réels pour représenter M_r tandis qu'il nous faut $d \times p$ valeurs pour connaître M. Ainsi, si on assure $r(1+d+p) \ll d\times p$ et une erreur $\sqrt{\sum_{i=r+1}^{\min(d,p)} \sigma_i^2}$ inférieure à un certain seuil, on gagne en temps de calcul et en stockage machine en utilisant la matrice M_r au lieu de M. En plus de ce gain la SVD est utilisée dans plusieurs contextes: on la trouve en traitement de signal, imagerie, problèmes inverses et elle est aussi utilisée pour définir la pseudo-inverse d'une matrice.

Lien avec la POD:

Prenons l'exemple utilisé dans l'algorithme POD (1) où on suppose que pour tout $\mu \in \mathcal{P}_p$, u_μ s'écrit $u_\mu(x) = \sum_{k=1}^{N_h} (U_\mu)_k \phi_k(x)$ avec $\{\phi_k\}_{1\leq k\leq N_h}$ représente une famille orthonormée (pour tout $1 \leq i,j \leq N_h$ on a $\langle \phi_i, \phi_j\rangle_V = \delta_{i,j}$) et $U_\mu \in \mathbb{R}^{N_h}$, alors le produit scalaire $\langle u_{\mu_i}, u_{\mu_j}\rangle_V$ est remplacé par le produit scalaire euclidien dans $l^2(\mathbb{R}^{N_h})$ suivant,

$$\sum_{k=1}^{N_h} (U_{\mu_i})_k (U_{\mu_j})_k$$

alors la matrice de corrélation C s'écrit,

$$C = \frac{1}{p} U_p^T U_p$$

avec $U_p \in \mathbb{R}^{N_h\times p}$ la matrice qui a pour i-ème colonne le vecteur U_{μ_i}. Ainsi les valeurs propres de C correspondent aux carrés des valeurs singulières de $\frac{1}{\sqrt{p}} U_p$.

Algorithme glouton

Nous détaillons dans cette section une deuxième méthode pour obtenir un espace réduit de dimension n, tel que chaque élément de l'ensemble solution $\mathcal{M}$ soit bien approché par un élément de cet espace réduit.

Cette deuxième méthode repose sur l'utilisation d'un algorithme itératif, appelé algorithme glouton, que nous présentons ci-dessous.

Algorithm 2 Algorithme Glouton

Initialisation : Soit $\mu_1 \in \mathcal{P}$ tel que

$$\mu_1 \in \arg\sup_{\mu\in\mathcal{P}} \|u_\mu\|_V .$$

Soit $X_1^G := \mathrm{Span}\{u_{\mu_1}\}$ et $n = 2$.

Iteration $n \geq 2$**:** Soit $\mu_n \in \mathcal{P}$ tel que

$$\mu_n \in \arg\sup_{\mu\in\mathcal{P}} \left\| u_\mu - \Pi_{X_{n-1}^G} u_\mu \right\|_V$$

Soit $X_n^G := X_{n-1}^G + \mathrm{Span}\{u_{\mu_n}\} = \mathrm{Span}\{u_{\mu_1}, \cdots, u_{\mu_n}\}$ et $n := n + 1$.

Nous présentons également ici une deuxième classe plus large d'algorithmes gloutons, dépendant d'un paramètre $0 < \gamma \leq 1$, appelée algorithme faiblement glouton. Notez qu'un algorithme faiblement glouton tel que $\gamma = 1$ est un algorithme glouton.

Algorithm 3 Algorithme Faiblement Glouton

Initialisation: Soit $\gamma \in]0, 1]$, et soit $\mu_1 \in \mathcal{P}$ tel que

$$\|u_{\mu_1}\|_V^2 \geq \gamma^2 \sup_{\mu\in\mathcal{P}} \|u_\mu\|_V^2 .$$

Soit $X_1^{wG,\gamma} := \mathrm{Span}\{u_{\mu_1}\}$ et $n = 2$.

Iteration $n \geq 2$**:** Soit $\mu_n \in \mathcal{P}$ tel que

$$\left\| u_{\mu_n} - \Pi_{X_{n-1}^{wG,\gamma}} u_{\mu_n} \right\|_V^2 \geq \gamma^2 \sup_{\mu\in\mathcal{P}} \left\| u_\mu - \Pi_{X_{n-1}^{wG,\gamma}} u_\mu \right\|_V^2 .$$

Soit $X_n^{wG,\gamma} := X_{n-1}^{wG,\gamma} + \mathrm{Span}\{u_{\mu_n}\} = \mathrm{Span}\{u_{\mu_1}, \cdots, u_{\mu_n}\}$ et $n := n + 1$.

L'épaisseur de Kolmogorov

Une question naturelle est de pouvoir quantifier la qualité de l'approximation des éléments de $\mathcal{M}$ par des éléments de X_n^G ou $X_n^{wG,\gamma}$ pour une valeur de paramètre $0 < \gamma \leq 1$. Pour ce faire, nous introduisons ici, pour tout $n \in \mathbb{N}^*$, la n-ème épaisseur de Kolmogorov de l'ensemble $\mathcal{M}$ définie comme suit:

$$d_n(\mathcal{M}) := \inf_{\substack{V_n \subset V \text{ sous-espace vectoriel} \\ \dim V_n = n}} \sup_{\mu\in\mathcal{P}} \|u_\mu - \Pi_{V_n} u_\mu\|_V$$

La quantité $d_n(\mathcal{M})$ représente la meilleure erreur d'approximation qui puisse être possiblement obtenue lorsque les éléments de $\mathcal{M}$ sont approchés par un sous-espace vectoriel de dimension n de V.

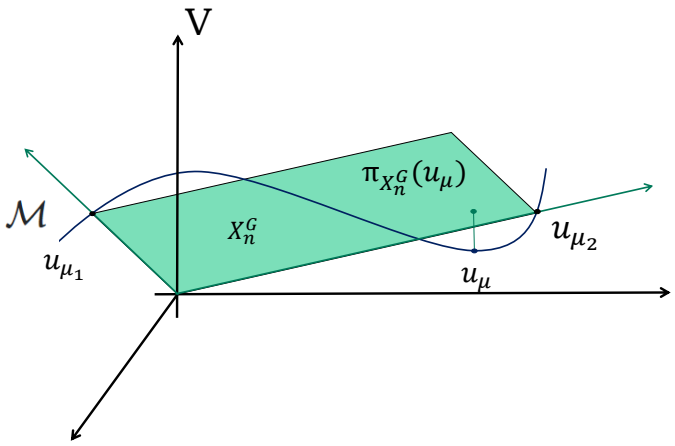


Figure I.2: Représentation géométrique de la variété $\mathcal{M}$ et de l'espace de dimension fini X_n^G.

Notons de plus,

$$\sigma_n^G(\mathcal{M}) := \sup_{\mu\in\mathcal{P}} \|u_\mu - \Pi_{X_n^G} u_\mu\|_V$$

et

$$\sigma_n^{wG,\gamma}(\mathcal{M}) := \sup_{\mu\in\mathcal{P}} \|u_\mu - \Pi_{X_n^{wG,\gamma}} u_\mu\|_V.$$

Il est alors naturel de chercher à comparer les quantités $\sigma_n^G(\mathcal{M})$ et $\sigma_n^{wG,\gamma}(\mathcal{M})$ avec $d_n(\mathcal{M})$. Notons tout d'abord qu'il est évident de remarquer que

$$\forall n \geq 1, \quad \sigma_n^G(\mathcal{M}) \geq d_n(\mathcal{M}) \quad \text{et} \quad \sigma_n^{wG,\gamma}(\mathcal{M}) \geq d_n(\mathcal{M}).$$

Plusieurs travaux successifs [7, 4, 11] ont permis d'établir le théorème suivant:

Theorem I.2.5. *[11] Pour tout $n \in \mathbb{N}^*$, $\sigma_n^{wG,\gamma}(\mathcal{M}) \leq \sqrt{2}\gamma^{-1} \min_{0\leq m<n} (d_m(\mathcal{M}))^{\frac{n-m}{n}}$. En particulier, pour tout $n \in \mathbb{N}^*$,*

$$\sigma_{2n}^{wG,\gamma}(\mathcal{M}) \leq \sqrt{2}\gamma^{-1}\sqrt{d_n(\mathcal{M})}. \tag{I.18}$$

Ce résultat montre que les espaces réduits obtenus à l'aide d'algorithmes gloutons ou faiblement gloutons ont alors des propriétés d'approximation quasi-optimales au sens de (I.18).

Notons qu'en pratique, il est impossible d'implémenter ces algorithmes en parcourant entièrement l'espace des paramètres $\mathcal{P}$ lorsque celui-ci est de cardinal infini. En pratique, un sous-ensemble $\mathcal{P}_p \subset \mathcal{P}$ de cardinal fini p, appelé *training set* doit être introduit pour mettre en oeuvre un tel algorithme glouton. L'objet du travail [10] consiste à analyser les propriétés d'un algorithme glouton où un *training set* est choisi de manière aléatoire à chaque itération de l'algorithme glouton.

I.2.2 Méthodes de réduction de problèmes déterministes classiques: bases réduites et POD-Galerkin

Nous présentons dans cette section comment les différentes méthodes de construction d'espaces réduits présentées à la Section I.2.1 peuvent être utilisées pour construire des modèles réduits dans le cas d'équations elliptiques stationnaires ou de systèmes d'équations différentielles ordinaires. Nous présentons enfin une troisième méthode de réduction de systèmes d'équations aux dérivées partielles ordinaires, appelée méthode POD dynamique.

Méthodes des bases réduites pour la réduction d'équations aux dérivées partielles stationnaires

Nous supposons ici que le problème paramétrique d'intérêt est un problème elliptique de la forme suivante. Pour tout $\mu \in \mathcal{P}$, $u^\mu \in V$ est défini comme l'unique solution dans V du problème posé sous forme variationnelle

$$a^\mu(u^\mu, v) = l(v), \quad \forall v \in V,$$

où $l : V \to \mathbb{R}$ est une forme linéaire continue et où $a^\mu : V \times V \to \mathbb{R}$ est une forme bilinéaire continue et coercive.

La méthode des bases réduites consiste à construire un modèle réduit pour calculer efficacement une approximation de la solution u^μ pour de nombreuses valeurs du paramètre $\mu \in \mathcal{P}$ dans la phase online. L'idée est d'utiliser une méthode d'approximation de Galerkin sur un espace réduit X_n de dimension n obtenu à l'issue de la phase offline (par une méthode POD ou par un algorithme glouton).

Pour tout $\mu \in \mathcal{P}$, une approximation $u_n^\mu \in X_n$ de u^μ est alors définie comme l'unique solution du problème variationnel:

$$a^\mu(u_n^\mu, v_n) = l(v_n), \quad \forall v_n \in X_n.$$

Un exemple utilisant l'algorithme Glouton:

Considérons à nouveau l'équation (I.2). Nous discrétisons l'espace V par un espace V_h de dimension finie N_h tel que $V_h \subset V$. On cherche ainsi à approximer $u_\mu \in V$ par $u_\mu^h \in V_h$. Cet espace V_h peut être construit en utilisant la méthode des éléments finis. On note $\{\phi_i\}_{i=1}^{N_h}$ les fonctions de la base d'éléments finis.
On cherche alors $u_\mu^h \in V_h$ tel que:

$$a^\mu(u_\mu^h, v^h) = l(v^h), \ \forall v^h \in V_h \tag{I.19}$$

L'écriture matricielle de ce problème (de dimension finie) nous donne:
Pour $\mu \in \mathcal{P}$ trouver $U_\mu^h \in \mathbb{R}^{N_h}$ tel que:

$$A_\mu^h U_\mu^h = L^h \tag{I.20}$$

avec $(A_\mu^h)_{ij} = a_\mu(\phi_j, \phi_i)$ et $(L_\mu^h)_i = l_\mu(\phi_i)$. Ainsi l'approximation u_μ^h est donnée par

$$u_\mu^h(x) = \sum_{i=1}^{N_h} (U_\mu^h)_i \phi_i(x). \tag{I.21}$$

Remarquons ici qu'on doit calculer A_μ^h pour tout $\mu \in \mathcal{P}$ et résoudre le problème (I.20) dans $\mathbb{R}^{N_h}$.

Le modèle réduit

Le but de la base réduite est alors de trouver une base X_n de dimension n tel que $n \ll N_h$, et de sorte que dans la partie *online* on n'a pas à recalculer des matrices de taille $N_h \times N_h$ pour chaque μ. On peut utiliser l'algorithme Glouton (2) pour générer l'espace $X_n^G = \text{Span}\{\xi_1, ..., \xi_n\}$ telle que:

$$\xi_j = \sum_{i=1}^{N_h} B_{ij}\phi_i \tag{I.22}$$

avec $B \in \mathbb{R}^{N_h \times n}$ la matrice de passage de la base $\{\phi_i\}_{i=1}^{N_h}$ vers la base $\{\xi_i\}_{i=1}^{n}$. Le problème réduit est alors,
Trouver $U_\mu^{br} \in \mathbb{R}^n$ solution de

$$A_\mu^{br} U_\mu^{br} = L_\mu^{br} \tag{I.23}$$

Avec $A_\mu^{br} = B^T A_\mu^h B$ et $L_\mu^{br} = B^T L_\mu^h$. Ainsi la solution $u_\mu^{br} \in X_n$ du problème réduit s'écrit:

$$u_\mu^{br}(x) = \sum_{i=1}^{n} (U_\mu^{br})_i \xi_i(x). \tag{I.24}$$

Le problème à résoudre est non plus sur $\mathbb{R}^{N_h}$ mais sur $\mathbb{R}^n$.

Remarques:

- Cette remarque concerne le choix du training set $\mathcal{P}_p$. En utilisant la notion de ϵ-*covering number*, on peut montrer que, si l'épaisseur de Kolmogorov de la variété des solutions fines décroit en $O(n^{-s})$, pour s positif, alors le cardinal de P_p se comporte typiquement comme $\exp(C\epsilon^{-1/s})$ pour obtenir des erreurs online de l'ordre de ϵ, pour un C positif. Dans le travail [10], A. Cohen et al. montrent qu'on peut choisir un training set de taille polynomiale en ϵ^{-1} (à comparer au comportement en $\exp(C\epsilon^{-1/s})$ ci-dessus) quitte à accepter des résultats avec une erreur supérieure à ϵ avec une faible probabilité.

- Dans l'algorithme glouton (2), nous avons besoin d'estimer l'erreur entre la solution exacte et la solution approchée construite sur la base réduite, pour chaque valeur du paramètre dans le training set. En pratique, ceci est bien sûr prohibitif en terme de coût calcul. Dans le cas d'EDP paramétriques, il existe souvent des estimateurs d'erreur a posteriori, qui permettent d'obtenir une approximation de la vraie erreur, à un coût calcul en $O(n)$ et non pas $O(N_h)$. Il n'est plus alors nécessaire de calculer la solution du problème fin pour chaque élément du training set, mais uniquement pour les n fonctions de bases choisies. Noter que grâce à cette méthode, le coût calcul *offline* de l'algorithme glouton est nettement moindre que le coût calcul offline d'une méthode POD (qui nécessite de calculer l'ensemble des solutions de référence pour tous les éléments du training set). Plus précisément, la méthode des bases réduites a un coût *offline* de l'ordre de $O(nN_h^2)$ (en supposant que la résolution du problème fin est en $O(N_h^2)$), alors que le coût *offline* de

la POD est en $O(N_p N_h^2)$, sans tenir compte du coût calcul de la SVD nécessaire pour extraire la base réduite.

- Pour que l'algorithme de base réduite soit efficace en pratique, il faut que la dépendance de la forme bilinéaire a et de la forme linéaire l satisfassent une décomposition dite affine de la forme suivante :

$$a(u,v,\mu)=\sum_{i=1}^{N_a}\theta_i^a(\mu)a_i(u,v),\quad l(v,\mu)=\sum_{i=1}^{N_l}\theta_i^l(\mu)l_i(v).$$

En effet, sous cette hypothèse, on peut montrer que le calcul de l'estimateur a posteriori dans la phase *offline* d'une part, et de la matrice à inverser dans la phase *online* d'autre part peut se faire en une complexité indépendante de N_h. Par exemple, le coût *online* pour l'évaluation de la solution en une valeur du paramètre est de l'ordre de $O(n^3+n^2N_a+nN_l)$.

Méthode POD-Galerkin pour la réduction de systèmes d'équations différentielles ordinaires

Soit $T>0$ un temps final et $d\in\mathbb{N}^*$. Pour tout $\mu\in\mathcal{P}$, soit $\mathcal{F}^\mu:[0,T]\times\mathbb{R}^d\to\mathbb{R}^d$ une fonction Lipschitz. Pour tout $\mu\in\mathcal{P}$ et pour tout $X_0^\mu\in\mathbb{R}^d$, on note $X^\mu:[0,T]\to\mathbb{R}^d$ l'unique solution du problème de Cauchy-Lipschitz suivant:

$$\begin{cases}\dot{X}^\mu(t)=\mathcal{F}^\mu(t;X^\mu(t)),\quad \forall t\in[0,T],\\ X^\mu(0)=X_0^\mu.\end{cases}\tag{I.25}$$

L'ensemble solution de cette équation différentielle ordinaire paramétrée est alors définie comme

$$\mathcal{M}:=\{X^\mu(t);\mu\in\mathcal{P},\ t\in[0,T]\}\subset V:=\mathbb{R}^d.$$

Supposons qu'un sous-espace vectoriel de petite dimension $V_n\subset\mathbb{R}^d$ ait été construit de telle sorte que chaque élément de $\mathcal{M}$ soit proche de sa projection orthogonale sur V_n (en utilisant par exemple l'algorithme POD ou l'algorithme glouton présentés dans les sections précédentes), la méthode POD-Galerkin consiste alors à construire un modèle réduit pour approcher la solution de (I.25) de la manière suivante. Pour tout $\mu\in\mathcal{P}$, soit $\mathcal{F}_n^\mu:[0,T]\times V_n\to V_n$ telle que

$$\forall t\in[0,T], Y\in V_n,\quad \mathcal{F}_n^\mu(t,Y):=\Pi_{V_n}\mathcal{F}^\mu(t,Y).$$

Alors, l'approximation $X_n^\mu:[0,T]\to V_n$ de la solution X^μ du système d'équations différentielles ordinaires (I.25) est alors définie comme l'unique solution du problème de Cauchy-Lipschitz (de petite dimension)

$$\begin{cases}\dot{X}_n^\mu(t)=\mathcal{F}_n^\mu(t,X_n^\mu(t)),\ \forall t\in[0,T],\\ X_n^\mu(0)=\Pi_{V_n}X_0^\mu.\end{cases}$$

Un exemple utilisant la POD:

Considérons un exemple d'équations différentielle ordinaire paramétrée. Soit pour tout $\mu\in\mathcal{P}$, la fonction $f_\mu:[0,T]\times\mathbb{R}\to\mathbb{R}$. On est intéressé par la solution $u:\mathcal{P}\times[0,T]\to\mathbb{R}$ de l'équation suivante :

$$\begin{cases} \partial_t u(\mu,t) = f_\mu(t,u(\mu,t)), \quad \forall t>0, \\ u(\mu,0)=0. \end{cases} \tag{I.26}$$

On discrétise $\mathcal{P}$ par un espace $\mathcal{P}_p$ de cardinal fini et on utilise la forme matricielle suivante. Soit $U(t)\in\mathbb{R}^p$ tel que $U(t)=(U_i(t))_{(1\leq i\leq p)}$ avec $U_i(t)=u(\mu_i,t)$. La dynamique sur $U(t)$ s'écrit:

$$\frac{d}{dt}U(t)=\mathcal{F}(t,U(t)), \tag{I.27}$$

avec $\mathcal{F}\in\mathbb{R}^p$ tel que $(\mathcal{F})_i=f_{\mu_i}(t,u(\mu_i,t))$. Cette dynamique est ensuite discrétisée en temps par un schéma d'Euler explicite par exemple. Considérons une discrétisation en temps de pas $\Delta t=\frac{T}{N_T}$ de sorte que $t_k=k\Delta t$ avec $0\leq k\leq N_T$. La résolution du système discret en temps nous donne ainsi un vecteur U^k comme approximation de $U(t_k)$:

$$\begin{cases} \dfrac{U^{k+1}-U^k}{\Delta t}=\bar{\mathcal{F}}(t_k,U^k), \\ U^0=U(0), \end{cases} \tag{I.28}$$

avec $\bar{\mathcal{F}}$ un opérateur obtenu par un schéma de discrétisation temporel appliqué à $\mathcal{F}$.

Pour construire la base POD on prend alors la matrice des snapshots $M=(U^1,\cdots,U^{N_T})$ de laquelle on extrait l'information essentielle. On applique la SVD sur la matrice M pour l'écrire $M=\bar{U}\bar{\Sigma}\bar{V}^T$, avec $\bar{U}\in\mathbb{R}^{p\times p}$, $\bar{\Sigma}\in\mathbb{R}^{p\times N_T}$, $\bar{V}\in\mathbb{R}^{N_T\times N_T}$, on prend les n premiers vecteurs colonnes de $\bar{U}=(V^1,\cdots,V^p)$, où $n\in\mathbb{N}^*$, pour construire la base X_n^{POD} de la POD.
On cherche alors $U_n(t)$ comme approximation de $U(t)$ par une projection de Galerkin sur la base X_n^{POD}, en écrivant $U_n(t)=\sum_{j=1}^n c_j(t)V^j$. On obtient,

$$\frac{d}{dt}\left(\sum_{j=1}^n c_j(t)V^j\right)=\mathcal{F}(t,U_n(t)). \tag{I.29}$$

En utilisant le fait que la famille $\{V^i\}_{1\leq i\leq n}$ est orthonormée, on projette sur V^i pour obtenir:

$$\frac{d}{dt}c_i(t)=(V^i)^T\mathcal{F}(t,\sum_{j=1}^n c_j(t)V^j),\ 1\leq i\leq n \tag{I.30}$$

On obtient alors une dynamique sur $\mathbb{R}^n$ et non plus sur $\mathbb{R}^p$ portée par n équations indépendantes à n inconnus $(c_i)_{1\leq i\leq n}$.

Remarques:

- Noter que la POD ne nécessite pas de disposer d'un estimateur a posteriori. Mais il est difficile de choisir le nombre de paramètres p à utiliser a priori, pour avoir une erreur d'approximation fixée. La méthode des bases réduites permet au contraire de s'affranchir de cette difficulté, on arrêtant l'enrichissement de la base réduite durant la phase *offline* quand l'erreur souhaitée est atteinte.
- La méthode POD utilise une base fixe en temps. Comme nous allons maintenant l'expliquer dans la section suivante, il peut être utile en pratique d'utiliser une base réduite qui évolue en temps pour obtenir de meilleurs résultats.

I.2.3 Méthode POD dynamique pour la réduction de systèmes d'équations différentielles ordinaires

Nous présentons ici une troisième méthode de réduction de modèles qui ne nécessite pas, contrairement aux deux méthodes précédentes, de construire au préalable un sous-espace vectoriel de dimension faible permettant d'approcher de manière satisfaisante l'ensemble des solutions du problème. Nous présentons ici le principe de la méthode dans le cas de la réduction d'un système d'équations différentielles ordinaires paramétré de la forme (I.25) dans le cas où l'ensemble des valeurs de paramètres $\mathcal{P}$ est un ensemble de cardinal fini noté p. Notons alors $\mu_1, \cdots, \mu_p$ les éléments de $\mathcal{P}$ de telle sorte que $\mathcal{P} := \{\mu_1, \cdots, \mu_p\}$. Pour tout $t \in [0,T]$, notons également $X(t) \in \mathbb{R}^{d\times p}$ la matrice définie telle que sa $q^{\text{ème}}$ colonne soit égale à $X^{\mu_q}(t)$ pour tout $1 \leq q \leq p$. Notons également $\mathcal{F} : [0,T] \times \mathbb{R}^{d\times p} \to \mathbb{R}^{d\times p}$ la fonction définie telle que, pour tout $X = (X_1, \cdots, X_p) \in \mathbb{R}^{d\times p}$ et pour tout $1 \leq q \leq p$,

$$\left(\mathcal{F}(t,X)\right)_q := \mathcal{F}^{\mu_q}(t; X_q),$$

où $\left(\mathcal{F}(t,X)\right)_q$ désigne la $q^{\text{ème}}$ colonne de $\mathcal{F}(t,X)$.

Le problème (I.25) se réécrit alors sous la forme du problème matriciel suivant:

$$\begin{cases} \dot{X}(t) = \mathcal{F}(t; X(t)), \ \forall t \in [0,T], \\ X(0) = X_0, \end{cases} \tag{I.31}$$

où $X_0 = (X_0^{\mu_1}, \cdots, X_0^{\mu_p}) \in \mathbb{R}^{d\times p}$.

La méthode présentée ici est une méthode d'approximation par rang faible appelée méthode POD dynamique, introduite tout d'abord par Lubich et Koch [21]. Cette méthode a par la suite été étendue à de nombreux autres types de problèmes [20].

Le principe de la méthode est le suivant: pour une valeur de $r \in \mathbb{N}^*$ et pour tout $t \in [0,T]$, on cherche à approcher la matrice $X(t)$ par un élément de la variété

$$\mathcal{R}_r = \{X_r \in \mathbb{R}^{d\times p}, \, rg(X_r) = r\},$$

des matrices de $\mathbb{R}^{d\times p}$ de rang r.

Il est connu (voir Section I.2.1) que pour tout $t \in [0,T]$, une meilleure approximation de rang r de la matrice $X(t)$, solution du problème de minimisation suivant

$$X_r(t) \in \underset{X_r \in \mathcal{R}_r}{\arg\min} \left\| X(t) - X_r \right\|, \tag{I.32}$$

est obtenue en considérant une décomposition SVD tronquée de rang r de la matrice $X(t)$.

Pour tout $t \in [0,T]$, il existe une matrice $U(t) = (U_1(t), \cdots, U_d(t)) \in \mathbb{R}^{d\times d}$ orthogonale, une matrice $V(t) = (V_1(t), \cdots, V_p(t)) \in \mathbb{R}^{p\times p}$ orthogonale et une matrice $S(t) := (S_{ij}(t))_{1\leq i\leq d,\, 1\leq j\leq p} \in \mathbb{R}^{d\times p}$ diagonale à coefficients positifs ou nuls telles que

$$X(t) = U(t)S(t)V(t)^T.$$

En supposant que $r \leq \min(p,d)$, une meilleure approximation de rang r de la matrice $X(t)$ est alors donnée par

$$X_r(t) = \overline{U}_r(t)\overline{S}_r(t)\overline{V}_r(t)^T,$$

où $\overline{U}_r(t) := (U_1(t), \cdots, U_r(t)) \in \mathbb{R}^{d\times r}$, $\overline{V}_r(t) := (V_1(t), \cdots, V_r(t)) \in \mathbb{R}^{p\times r}$ et $\overline{S}_r(t) := (S_{ij}(t))_{1\leq i,j\leq r} \in \mathbb{R}^{r\times r}$.

Cependant, en pratique, calculer la décomposition SVD de la matrice $X(t)$ nécessite (i) de calculer la matrice $X(t)$ complète pour tout $t \in [0,T]$ et (ii) coûte très cher d'un point de vue computationnel lorsque p et d sont grands.

La méthode POD dynamique consiste à construire pour tout $t \in [0,T]$ une approximation $Y_r(t) \in \mathcal{R}_r$ de $X(t)$ de la forme

$$Y(t) = \widetilde{U}_r(t)\widetilde{S}_r(t)\widetilde{V}_r(t)^T, \tag{I.33}$$

avec $\widetilde{U}_r(t) \in \mathbb{R}^{d\times r}$, $\widetilde{V}_r(t) \in \mathbb{R}^{p\times r}$ et $\widetilde{S}_r(t) \in \mathbb{R}^{r\times r}$ obtenus comme les solutions d'un système d'équations différentielles ordinaires couplées. Plus précisément, la méthode de la POD dynamique consiste à déterminer la dérivée par rapport au temps $\dot{Y}(t)$ comme approchant au mieux la dérivée par rapport au temps $\dot{X}(t)$. De manière idéale, on chercherait à déterminer $\dot{Y}(t)$ comme solution du problème de minimisation

$$\dot{Y}(t) \in \underset{Z\in\mathcal{T}_{Y(t)}\mathcal{R}_r}{\arg\min} \left\| Z - \dot{X}(t) \right\|, \tag{I.34}$$

où pour tout $Y \in \mathcal{R}_r$, $\mathcal{T}_Y\mathcal{R}_r$ désigne l'espace tangent à la variété $\mathcal{R}_r$ au point Y. La condition initiale du modèle réduit est par ailleurs fixée à $Y(0) = X_r(0)$.

Bien sûr, le problème (I.34) ne peut pas être résolu en pratique, à moins de connaître a priori la solution du modèle de départ X. Cependant, dans le cas où X est solution de (I.31), il se trouve que $\dot{X}(t) = \mathcal{F}(t, X(t))$. Par ailleurs, en supposant que le modèle réduit ainsi construit $Y(t)$ soit une bonne approximation de $X(t)$ pour tout $t \in [0,T]$, on peut considérer la quantité $A(t) := \mathcal{F}(t, Y(t))$ comme une approximation de $\mathcal{F}(t, X(t))$.

Au final, la méthode de la POD dynamique consiste à calculer numériquement pour tout $t \in [0,T]$ une approximation $Y(t) \in \mathcal{R}_r$ de $X(t)$ de telle sorte que $Y(0) = X_r(0)$ et

$$\dot{Y}(t) \in \underset{Z\in\mathcal{T}_{Y(t)}\mathcal{R}_r}{\arg\min} \|Z - A(t)\|. \tag{I.35}$$

Résoudre (I.35) revient à écrire un système d'équations différentielles ordinaires couplées pour décrire l'évolution des matrices $\widetilde{U}_r$, $\widetilde{V}_r$ et $\widetilde{S}_r$ intervenant dans la décomposition (I.33) en fonction du temps.

Caractérisation des espaces tangents

Toute matrice $Y \in \mathcal{R}_r$ peut s'écrire sous la forme (non unique),

$$Y = USV^T, \tag{I.36}$$

avec $U \in \mathbb{R}^{d\times r}$, $V \in \mathbb{R}^{p\times r}$ deux matrices orthogonales et $S \in \mathbb{R}^{r\times r}$ non singulière. Plus précisément, les matrices U et V vérifient

$$U^TU = I_r \quad \text{and} \quad V^TV = I_r, \tag{I.37}$$

avec I_r la matrice identité dans $\mathbb{R}^{r\times r}$.

Le fait que cette décomposition ne soit pas unique fait que caractériser de manière simple l'espace tangent $\mathcal{T}_Y\mathcal{R}_r$ n'est pas évident. Dans [21], Lubich et Koch montrent la Proposition I.2.6 ci-dessous. Pour tout $m \in \mathbb{N}^*$, soit $\mathcal{V}_{m,r}$ la variété de Stiefel des matrices orthogonales de $\mathbb{R}^{m\times r}$. Pour tout $U \in \mathcal{V}_{m,r}$, l'espace tangent à la variété $\mathcal{V}_{m,r}$ au point U est donné par

$$\mathcal{T}_U\mathcal{V}_{m,r} = \left\{\partial U \in \mathbb{R}^{m\times r}, \partial U^T U + U^T \partial U = 0\right\}.$$

Le résultat de [21] est alors le suivant:

Proposition I.2.6. *Soit $\mathcal{A}(r)$ l'ensemble des matrices antisymétriques de $\mathbb{R}^{r\times r}$. L'application linéaire suivante:*

$$\begin{aligned} \mathbb{R}^{r\times r} \times \mathcal{T}_U\mathcal{V}_{m,r} \times \mathcal{T}_V\mathcal{V}_{n,r} &\to \mathcal{T}_Y\mathcal{R}_r \times \mathcal{A}(r) \times \mathcal{A}(r) \\ (\partial S, \partial U, \partial V) &\to (\partial U S V^T + U \partial S V^T + U S \partial V^T, U^T\partial U, V^T \partial V) \end{aligned} \tag{I.38}$$

est un isomorphisme.

Le résultat suivant [21] est un corollaire de la Proposition I.2.6 et permet de caractériser les éléments de $\mathcal{T}_Y\mathcal{R}_r$ pour tout $Y \in \mathcal{R}_r$.

Proposition I.2.7. *Soit $Y \in \mathcal{R}_r$ telle que $Y = USV^T$ avec $U \in \mathcal{V}_{d,r}$, $V \in \mathcal{V}_{p,r}$ et $S \in \mathbb{R}^{r\times r}$ inversible. Pour tout $\partial Y \in \mathcal{T}_Y\mathcal{R}_r$, il existe $\partial U \in \mathcal{T}_U\mathcal{V}_{d,r}$, $\partial V \in \mathcal{T}_V\mathcal{V}_{p,r}$ et $\partial S \in \mathbb{R}^{r\times r}$ telles que*

$$\partial Y = \partial U S V^T + U \partial S V^T + U S \partial V^T. \tag{I.39}$$

Si de plus, on impose les contraintes d'orthogonalité suivantes:

$$U^T \partial U = 0, \quad \text{et} \quad V^T \partial V = 0, \tag{I.40}$$

alors les matrices ∂S, ∂U et ∂V sont déterminées d'une façon unique.

Système réduit pour la méthode POD dynamique

Soit $Y \in \mathcal{R}_r$ telle que $Y = USV^T$ avec $U \in \mathcal{V}_{d,r}$, $V \in \mathcal{V}_{p,r}$ et $S \in \mathbb{R}^{r\times r}$. Notons $P_U = UU^T$ et $P_V = VV^T$ les projecteurs orthogonaux sur les colonnes de U et de V respectivement ainsi que $P_U^\perp = I_d - P_U$ et $P_V^\perp = I_p - P_V$. Soit $\partial Y \in \mathcal{T}_Y\mathcal{R}_r$ s'écrivant sous la forme (I.39), où les conditions d'orthogonalité (I.40) sont imposées. Alors, il vient immédiatement

$$\begin{aligned} \partial S &= U^T \partial Y V, \\ \partial U &= P_U^\perp \partial Y V S^{-1}, \\ \partial V &= P_V^\perp \partial Y^T U S^{-T}. \end{aligned} \tag{I.41}$$

Il existe donc un isomorphisme entre l'espace

$$\{(\partial S, \partial U, \partial V) \in \mathbb{R}^{r\times r} \times \mathbb{R}^{d\times r} \times \mathbb{R}^{p\times r}, U^T\partial U = 0, V^T \partial V = 0\}$$

et l'espace tangent $\mathcal{T}_Y\mathcal{R}_r$.

Revenons à la méthode POD dynamique et au problème de minimisation (I.35). En notant pour tout $t \in [0,T]$ $A(t) := \mathcal{F}(t, Y(t))$, le problème de minimisation (I.35) est équivalent au problème variationnel suivant: trouver $\dot{Y}(t) \in \mathcal{T}_{Y(t)}\mathcal{R}_r$ solution de:

$$\left\langle \dot{Y}(t) - A(t), \partial Y \right\rangle = 0, \quad \forall \partial Y \in \mathcal{T}_{Y(t)}\mathcal{R}_r. \tag{I.42}$$

où pour toute matrice $A, B \in \mathbb{R}^{d\times p}$, $\langle A, B\rangle := \mathrm{Tr}(A^T B)$. En utilisant le résultat de la Proposition I.2.7, on obtient le résultat suivant [21]:

Proposition I.2.8. *Pour tout $Y(t) = U(t)S(t)V(t)^T \in \mathcal{R}_r$, $S(t) \in \mathbb{R}^{r\times r}$ non singulière, $U(t) \in \mathcal{V}_{d,r}$ et $V(t) \in \mathcal{V}_{p,r}$, alors $\dot{Y}(t)$ est solution de (I.35) (et de (I.42)) si et seulement si:*

$$\dot{Y}(t) = \dot{U}(t)S(t)V(t)^T + U(t)\dot{S}(t)V(t)^T + U(t)S(t)\dot{V}(t)^T \tag{I.43}$$

avec

$$\begin{aligned} \dot{S}(t) &= U(t)^T A(t) V(t), \\ \dot{U}(t) &= P^{\perp}_{U(t)} A(t) V(t) S(t)^{-1}, \\ \dot{V}(t) &= P^{\perp}_{V(t)} A(t)^T U(t) S(t)^{-T}. \end{aligned} \tag{I.44}$$

La méthode POD dynamique revient alors à résoudre le système d'équations différentielles ordinaires (I.43)-(I.44) pour construire un modèle réduit $Y(t))_{t\in[0,T]}$ de telle sorte que pour tout $t \in [0,T]$, $Y(t)$ soit une approximation de rang r de $X(t)$.

L'analyse de l'erreur entre $X(t)$ et $Y(t)$ dans [21] repose sur des estimées de la courbure de la variété $\mathcal{R}_r$.

Remarquons que l'inverse de la matrice $S(t)$ intervient dans les équations (I.44). Un mauvais conditionnement de cette matrice engendre des problèmes de stabilité.

Citons ici d'autres travaux en lien avec l'analyse de la méthode POD dynamique. Feppon et Lermusiaux [15], ont plus étudié la dépendance de la solution de rang faible $Y(t)$ obtenue par la dynamique orthogonale avec la courbure de $\mathcal{R}_r$ au point $Y(t)$ en dérivant une approche géométrique pour étudier le lien entre cette courbure et la plus petite valeur singulière. Cette approche utilise l'application de Weingarten et leur a permis d'améliorer les résultats de [21].

Musharbach et Nobile ont développé des analyses d'erreur de l'approximation dynamique de rang faible pour des EDP paraboliques dépendantes d'un paramètre aléatoire, où ils proposent une formulation de la dynamique de rang faible avec une contrainte d'orthogonalité uniquement sur la base spatiale et une contrainte d'espérance nulle sur les vecteurs aléatoires. Ils obtiennent un résultat semblable au résultat de Koch et Lubich [20].

Le schéma *Projector-Splitting*

Pour résoudre les équations (I.44) il est nécessaire d'utiliser des schémas de discrétisation en temps adaptés. Lubich et Oseledets proposent un schéma d'intégration numérique, basé sur une méthode de splitting, développé dans [19] qui présente plusieurs propriétés que nous présentons ici.

Le problème (I.42) est équivalent à effectuer une projection orthogonale de la matrice $A(t)$ sur l'espace tangent $\mathcal{T}_{Y(t)}\mathcal{R}_r$, ce qui s'écrit

$$\begin{cases} \dot{Y}(t) = \Pi_{\mathcal{T}_{Y(t)}\mathcal{R}_r}(A(t)), \\ Y(0) = X_r(0), \end{cases} \tag{I.45}$$

avec $A(t) = \mathcal{F}(t, Y(t))$ et où $X_r(0)$ est une meilleure approximation de rang r de $X(0)$. Le lemme suivant est prouvé dans [21].

Lemma I.2.9. *[21] Soit $Y \in \mathcal{R}_r$ telle que $Y = USV^T$ où $U \in\in \mathcal{V}_{d,r}$, $V \in \mathcal{V}_{p,r}$ et $S \in \mathbb{R}^{r\times r}$. Alors le projecteur orthogonal sur le plan tangent $\mathcal{T}_Y\mathcal{R}_r$ au point Y est donné par, pour tout $Z \in \mathbb{R}^{d\times p}$,*

$$\Pi_{\mathcal{T}_Y\mathcal{R}_r} Z = ZP_V - P_U ZP_V + P_U Z. \tag{I.46}$$

Le schéma *Projector-Splitting* utilisé se base sur les étapes de Lie-Trotter qui consistent à résoudre séparément chaque projection du projecteur (I.46) dans un ordre précis. Soit $\Delta t > 0$ un pas de temps et pour tout $n \in \mathbb{N}$, on note Y^n l'approximation numérique donnée par le schéma de discrétisation en temps de la valeur $Y(t_n)$ où $t_n := n\Delta t$.

Le schéma numérique *Projector-Splitting* est constitué de trois étapes pour calculer Y^{n+1} en fonction de Y^n que nous détaillons ci-dessous. Supponsons tout d'abord que $Y_n = U_n S_n V_n^T$ avec $U_n \in \mathcal{V}_{d,r}$, $V_n \in \mathcal{V}_{p,r}$ et $S_n \in \mathbb{R}^{r\times r}$.

- Etape 1: Soit
$$\partial Y^1_{n+1} := A_n P_{V_n}$$
où $A_n := \mathcal{F}(t_n, Y_n)$ et soit $Y^1_{n+1} := Y_n + \Delta t \partial Y^1_{n+1}$. On calcule alors $U_{n+1} \in \mathcal{V}_{d,r}$ et $S^1_{n+1} \in \mathbb{R}^{r\times r}$ de telle sorte que
$$Y^1_{n+1} := U_{n+1} S^1_{n+1} V_n^T.$$

- Etape 2: Soit
$$\partial Y^2_n := -P_{U_{n+1}} A^1_{n+1} P_{V_n}$$
où $A^1_{n+1} := \mathcal{F}(t_{n+1}, Y^1_{n+1})$ et soit $Y^2_n := Y^1_{n+1} + \Delta t \partial Y^2_n$. On calcule alors $S^2_n \in \mathbb{R}^{r\times r}$ de telle sorte que
$$Y^2_n := U_{n+1} S^2_n V_n^T.$$

- Etape 3: Soit
$$\partial Y^3_{n+1} := P_{U_{n+1}} A^2_n$$
où $A^2_n := \mathcal{F}(t_n, Y^2_n)$ et soit $Y_{n+1} := Y^2_n + \Delta t \partial Y^3_{n+1}$. On calcule alors $V_{n+1} \in \mathcal{V}_{p,r}$ et $S_{n+1} \in \mathbb{R}^{r\times r}$ de telle sorte que
$$Y_{n+1} := U_{n+1} S_{n+1} V^T_{n+1}.$$

Il est prouvé que ce schéma est d'ordre 1. Notons qu'il est possible d'utiliser des schémas d'ordre plus élevé, comme ceux présentés dans [19].

I.3 Méthodes de réduction de variance pour le calcul d'espérances

Le but du deuxième chapitre est d'utiliser les bases réduites pour construire une variable de contrôle pour réduire la variance de l'estimateur. Nous présentons ici quelques méthodes de réduction de variance pour le calcul d'espérances.

I.3.1 Méthode de Monte-Carlo

Soit Z un vecteur aléatoire de dimension $d \in \mathbb{N}^*$ et $f : \mathbb{R}^d \to \mathbb{R}$.

Il existe deux types de méthodes pour approcher numériquement la valeur de l'espérance $\mathbb{E}[f(Z)]$: les méthodes déterministes et les méthodes probabilistes.

Les méthodes déterministes nécessitent la connaissance explicite de la loi de Z. Dans le cas où Z est un processus aléatoire solution d'une Equation Différentielle Stochastique, elles nécessitent de résoudre une Equation aux Dérivées Partielles donnée par la formule de Feynman-Kac. Ces méthodes déterministes sont limitées à des problèmes en petite dimension.

Les méthodes probabilistes se basent sur la construction d'estimateurs aléatoires pour approcher $\mathbb{E}[f(Z)]$. Le plus utilisé d'entre eux est l'estimateur de Monte-Carlo:

$$\mathbb{E}_M[f] = \frac{1}{M}\sum_{i=1}^{M} f(Z_i) \tag{I.47}$$

où $M \in \mathbb{N}^*$ et $(Z_i)_{1\leq i\leq M}$ est une famille de vecteurs aléatoires indépendants et identiquement distribués selon la loi de Z. Cet estimateur est sans biais et converge par la loi forte des grands nombres vers $\mathbb{E}[f(Z)]$ en norme L^1 et presque sûrement:

$$\mathbb{E}\left[|\mathbb{E}_M[f] - \mathbb{E}[f(Z)]|\right] \underset{M\to\infty}{\to} 0 \text{ et } \mathbb{E}_M[f] \underset{M\to\infty}{\to} \mathbb{E}[f(Z)] \ p.s \tag{I.48}$$

La loi faible des grands nombres assure également la convergence en norme L^2 de cet estimateur:

$$\mathbb{E}\left[|\mathbb{E}_M[f] - \mathbb{E}[f(Z)]|^2\right] = \sqrt{\frac{\mathrm{Var}[f(Z)]}{M}} \underset{M\to\infty}{\to} 0 \tag{I.49}$$

De plus, le théorème central limite permet d'obtenir une estimation de la loi de l'erreur d'approximation:

$$\mathbb{P}\left[|\mathbb{E}_M[f] - \mathbb{E}[f(Z)]| \leq a\sqrt{\frac{\mathrm{Var}[f(Z)]}{M}}\right] \underset{M\to\infty}{\to} \int_{-a}^{a} \frac{\exp -\frac{x^2}{2}}{\sqrt{2\pi}}. \tag{I.50}$$

Ce qui donne pour $a = 1.96$ un intervalle de confiance de 95%.
La formule (I.50) nous assure que l'erreur statistique commise entre $\mathbb{E}[f(Z)]$ et son approximation par (I.47) est typiquement de l'ordre de

$$\sqrt{\frac{\mathrm{Var}[f(Z)]}{M}}.$$

On peut diminuer cette erreur statistique soit en augmentant M, soit en diminuant la variance $\mathrm{Var}[f(Z)]$. Pour estimer la variance $\mathrm{Var}[f(Z)]$ en pratique on introduit aussi un estimateur de cette quantité

$$\mathrm{Var}_M[f] = \frac{1}{M}\sum_{i=1}^{M} f^2(Z_i) - (\frac{1}{M}\sum_{i=1}^{M} f(Z_i))^2,$$

qui converge par la loi forte des grands nombre vers $\mathrm{Var}\,[f(Z)]$. En utilisant le théorème de Slutsky on montre que:

$$\mathbb{P}\left[|\mathbb{E}_M[f] - \mathbb{E}[f(Z)]| \leq a\sqrt{\frac{\mathrm{Var}_M[f]}{M}}\right] \underset{M\to\infty}{\to} \int_{-a}^{a} \frac{\exp -\frac{x^2}{2}}{\sqrt{2\pi}}. \tag{I.51}$$

Le principe d'une méthode de réduction de variance est de construire un estimateur sans biais $\mathrm{K_M}$ de $\mathbb{E}[f(Z)]$ et telle que

$$\mathrm{Var}[\mathrm{K_M}] \ll \mathrm{Var}[\mathbb{E}_M(f)].$$

Noter qu'il faut aussi prendre en compte le coût de calcul de l'estimateur $\mathrm{K_M}$ pour juger de son efficacité par rapport à l'estimateur classique de Monte-Carlo.
Supposons par exemple que le coût de calcul de la méthode Monte-Carlo est de $C \times M$ avec C le coût de calcul de $f(Z_i)$ pour chaque i tel que $1 \leq i \leq M$. Soit ϵ l'erreur statistique tel que $\epsilon = \sqrt{\frac{\mathrm{Var}(f(Z))}{M}}$, ainsi le coût de calcul par la méthode Monte Carlo standard , $\mathrm{C_{sim}}(\mathbb{E}_\mathrm{M}(f))$, pour obtenir une erreur statistique ϵ est,

$$\mathrm{C_{sim}}(\mathbb{E}_\mathrm{M}(f)) = \frac{C \times \mathrm{Var}(f(Z))}{\epsilon^2}.$$

Le même raisonnement pour l'estimateur $\mathrm{K_M} = \frac{1}{M}\sum_{i=1}^{M} Y_i$ avec $(Y_i)_{1\leq \mathrm{i} \leq \mathrm{M}}$ des variables aléatoires iid selon une loi donnée, en supposant C_1 le coût de calcul d'un Y_i, nous donne $C_1 \times M$ comme coût de calcul de $\mathrm{K_M}$ et alors à erreur statistique fixée ϵ on a le coût de la méthode

$$\mathrm{C_{sim}}(\mathrm{K_M}) = \frac{C_1 \times \mathrm{Var}(\mathrm{Y_1})}{\epsilon^2}.$$

Le rapport $\mathrm{R} = \frac{C\mathrm{Var}((f(Z))}{C_1 \mathrm{Var}(Y_1)}$ doit ainsi être supérieur à 1 pour justifier l'efficacité de la méthode.

Les méthodes de réduction de variance ont pour objectif de proposer des estimateurs pour le calcul de $\mathbb{E}[f(Z)]$ dont la variance est plus faible que celle de l'estimateur de Monte-Carlo, de sorte à diminuer l'erreur statistique de l'approximation de l'espérance obtenue pour un nombre d'échantillons de Monte-Carlo de Z fixé. Nous présentons ci-dessous les deux méthodes les plus classiquement utilisées dans ce contexte, à savoir la méthode d'échantillonage préférentiel et la méthode de la variable de contrôle.

I.3.2 Échantillonnage préférentiel

Ce type de méthode est principalement utilisé dans le calcul d'évènements rares, en particulier lorsque la fonction f est presque nulle sur un domaine $D \subset \mathbb{R}^d$ tel la probabilité que $Z \in D$ soit très faible.

Supposons que la loi de Z soit caractérisée par une densité de probabilité ν définie sur $\mathbb{R}^d$. Une méthode d'échantillonage préférentiel consiste alors à introduire une nouvelle densité $q : \mathbb{R}^d$, que l'on sait facilement échantillonner, et de considérer la formule suivante:

$$\mathbb{E}[f(Z)] = \int_{\mathbb{R}^d} f(z)\nu(z)dz = \int_{\mathbb{R}^d} f(x)\frac{\nu(x)}{q(x)}q(x)dx. \tag{I.52}$$

La quantité $\mathbb{E}[f(Z)]$ est alors approchée par l'estimateur

$$\frac{1}{M}\sum_{i=1}^{M} f(X_i)\frac{\nu(X_i)}{q(X_i)},$$

où $(X_i)_{1\leq i\leq M}$ sont M vecteurs aléatoires indépendants et identiquement distribués selon la densité q.

L'inégalité de Cauchy-Schwartz montre que la densité optimale q pour obtenir la meilleure réduction de variance est donnée par $q_{\text{opt}}(x) = \frac{1}{\mathbb{E}[|f(Z)|]}|f(x)\nu(x)|$. En général, on ne sait pas échantillonner selon cette distribution optimale. Une bonne pratique est alors de prendre q de telle sorte à ce qu'elle favorise les régions où $|f(x)\nu(x)|$ prend des valeurs élevées.

I.3.3 Méthode de la Variable de contrôle

Le principe d'une méthode de variable de contrôle est d'introduire une nouvelle variable aléatoire Y de telle sorte que

- calculer l'espérance de Y d'une façon très rapide.
- réduire considérablement la variance de l'estimateur de Monte Carlo K_M qu'on va utiliser pour le calcul de la quantité d'intérêt $\mathbb{E}[f(Z)]$ devant la variance de l'estimateur standard de Monte Carlo $\mathbb{E}_M(f)$:

$$\text{Var}[\text{K}_\text{M}] \ll \text{Var}[\mathbb{E}_M(f)].$$

Remarquons qu'on peut écrire $\mathbb{E}[f(Z)]$ de la façon suivante:
$\forall\alpha \in \mathbb{R}$ on a

$$\mathbb{E}\left[f(Z)\right] = \mathbb{E}[f(Z) - \alpha Y] + \alpha\mathbb{E}[Y]. \tag{I.53}$$

Soit $\{Y_i\}_{1\leq i\leq M}$ une famille de variables aléatoires iid selon la loi de Y et $\alpha \in \mathbb{R}$. L'estimateur Monte-Carlo, en supposant $\mathbb{E}[Y]$ connue, est alors:

$$K_M(\alpha) = \frac{1}{M}\sum_{i=1}^{M}[f(Z_i) - \alpha Y_i] + \alpha\mathbb{E}[Y],$$

qui a une variance:

$$\text{Var}(K_M) = \frac{1}{M}[\text{Var}[f(Z)] - 2\alpha\text{Cov}(f(Z), Y) + \alpha^2\text{Var}(Y)],$$

en optimisant cette quantité $\text{Var}(K_M)$ en α, on obtient,

$$\alpha^* = \frac{\text{Cov}(f(Z), Y)}{\text{Var}(Y)},$$

et pour cette valeurs de α^*,

$$\text{Var}(K_M) = \frac{1}{M}\text{Var}(f(Z))[1 - \frac{\text{Cov}(f(Z), Y)^2}{\text{Var}(Y)\text{Var}(f(Z))}].$$

Rappelons que dans l'espace $\mathrm{L}^2(\Omega)$ des variables aléatoires à moyenne nulle muni du produit scalaire Cov(.,.) et de la norme Var(.) on a l'inégalité de Cauchy Schwartz:

$$(\mathrm{Cov}(f(Z),Y))^2 \leq \mathrm{Var}(f(Z))\mathrm{Var}(Y).$$

Donc égalité si et seulement si, $f(Z)$ et Y sont colinéaires. La réduction de variance est donc d'autant plus importante que Y et $f(Z)$ sont corrélées. Dans la pratique on prend $Y = g(Z)$ avec g une fonction qui est égale à f dans les régions où Z à une forte probabilité de se trouver et $\mathbb{E}[g(Z)]$ facilement calculable.

Cas de plusieurs variables de contrôle

Soit $g_1, .., g_s$, s fonctions de $\mathbb{R}^d$, $Y_i = (g_1(Z_i), .., g_s(Z_i))$ et $\alpha \in \mathbb{R}^s$. L'estimateur qu'on considère est maintenant:

$$K_M(\alpha) = \frac{1}{M}\sum_{i=1}^{M} f(Z_i) - \langle \alpha, Y_i - \mathbb{E}[Y_i]\rangle.$$

La variance de cet estimateur sans biais est:

$$\mathrm{Var}(K_M(\alpha)) = \frac{1}{M}\left(\mathrm{Var}[f(Z)] - 2\alpha^T\Sigma_{f,g} + \alpha^T\Sigma_g\alpha\right),$$

avec $(\Sigma_g)_{i,j} = \mathrm{Cov}(g_i(Z), g_j(Z))_{1\leq i,j\leq s}$ et $(\Sigma_{f,g})_i = \mathrm{Cov}(f(Z), g_i(Z))_{1\leq i\leq s}$.
Cette variance est ainsi minimisée pour $\alpha^* = \Sigma_g^{-1}\Sigma_{f,g}$ et on obtient donc:

$$\mathrm{Var}(\mathrm{K_M}(\alpha^*)) = \frac{1}{M}\mathrm{Var}[f(Z)]\left(1 - \frac{\Sigma_{f,g}^T\Sigma_g^{-1}\Sigma_{f,g}}{\mathrm{Var}(f(Z))}\right).$$

Ainsi on remarque que plus on augmente le nombre des variables de contrôle $g_i(Z)$, en supposant qu'elles soient linéairement indépendantes, et plus on gagne en réduction de variance mais alors plus ça nous coûte cher. Quand $s \to \infty$ on obtient une base de l'espace $L^2_{\nu,0}(\mathbb{R}^d) := \left\{g \in L^2_\nu(\mathbb{R}^d),\ \int_{\mathbb{R}^d} g\, d\nu = 0\right\}$, et alors $\Sigma_{f,g}^T\Sigma_f^{-1}\Sigma_{f,g} = \mathrm{Var}(f(Z))$ puisque $\Sigma_g^{-1} = I$ et $\Sigma_{f,g}^T\Sigma_{f,g} = \sum_{i=1}^{\infty}\mathrm{Cov}^2(f(Z), g_i(Z)) = ||f(Z)||^2_{L^2_{\nu,0}} = \mathrm{Var}(f(Z))$.

I.4 Contributions de la thèse

Le but de cette section est de présenter les deux principales contributions de la thèse, qui font l'objet des Chapitres 2 et 3 du manuscrit.

I.4.1 Réduction de variance par les bases réduites

La première contribution de la thèse, présentée dans le Chapitre 2, porte sur l'analyse mathématique d'une méthode de réduction de variance proposée par Boyaval et Lelièvre dans [6].

Nous rappelons succintement cette méthode ici, et ses objectifs, ainsi que les principaux résultats obtenus dans le cadre de la thèse.

Soit Z un vecteur aléatoire de dimension $d \in \mathbb{N}^*$ de loi donnée par une mesure de probabilité ν définie sur $\mathbb{R}^d$. On introduit l'espace $V := L^2_\nu(\mathbb{R}^d)$ défini par

$$L^2_\nu(\mathbb{R}^d) := \left\{g : \mathbb{R}^d \to \mathbb{R},\ \int_{\mathbb{R}^d} |g|^2\, d\nu < +\infty\right\}.$$

Cet espace est un espace de Hilbert muni du produit scalaire,

$$\langle f, g\rangle = \int_{\mathbb{R}^d} g(x)f(x)d\nu(x), \quad \forall g, f \in L^2_\nu(\mathbb{R}^d).$$

On note également $\|\cdot\|$ la norme associée.

Soit $\mathcal{P}$ un ensemble de valeurs de paramètres et pour tout $\mu \in \mathcal{P}$, soit $f_\mu \in L^2_\nu(\mathbb{R}^d)$. La méthode introduite dans [6] est une méthode de réduction de variance, utilisant un paradigme similaire à la méthode des bases réduites, de manière à calculer de manière efficace une approximation numérique de

$$\mathbb{E}\left[f_\mu\right]$$

pour tout $\mu \in \mathcal{P}$. Plus précisément, la méthode va consister à construire une variable de contrôle multiple $\bar{f}_\mu$ pour tout $\mu \in \mathcal{P}$ comme une combinaison linéaire d'un petit nombre n de fonctions $f_{\mu_1}, \cdots, f_{\mu_n}$ où les valeurs des paramètres $\mu_1, \cdots, \mu_n \in \mathcal{P}$ auront été sélectionné par un algorithme glouton similaire à celui présenté dans la Section I.2.1.

Cependant, un algorithme glouton ou faiblement glouton exact ne peut pas être implémenté en pratique dans un tel contexte stochastique. En effet, pour tout $f \in L^2_\nu(\mathbb{R}^d)$, la norme de f ne peut pas être calculée exactement, et il est nécessaire de faire appel à une méthode d'échantillonage de Monte-Carlo pour pouvoir approcher cette quantité.

L'objet du Chapitre 2 est de présenter une analyse mathématique qui permette de prédire a priori le nombre d'échantillons de Monte-Carlo devant être utilisés à chaque itération de l'algorithme glouton de manière à pouvoir garantir que ce dernier aura les mêmes propriétés d'approximation qu'un algorithme faiblement glouton avec une grande probabilité. Pour obtenir de telles estimées théoriques, nous utilisons en particulier des inégalités de concentration en distance de Wasserstein de la mesure empirique. Cependant, sur les exemples numériques que nous avons considéré, ces estimées semblent très pessimistes en comparaison de ce qui serait nécessaire pour obtenir des accélérations des temps de calcul significatifs par la méthode de réduction de variance considérée. Aussi, nous proposons également dans ce chapitre quelques méthodes heuristiques, inspirées du résultat théorique obtenu, afin de choisir le nombre d'échantillons de Monte-Carlo à utiliser à chaque itération de l'algorithme glouton.

I.4.2 Dynamique Orthogonale pour les équations différentielles stochastiques

L'objectif du Chapitre 3 est d'étudier numériquement le comportement d'une méthode de *POD* dynamique pour approcher les solutions d'Equations Différentielles Stochastiques paramétrées par des approximations de rang faible. Nous étudions en particulier le comportement d'un schéma numérique de type *Projector-Splitting,* similaire à celui présenté dans la Section I.2.3, pour la réduction de ce type d'équations stochastiques. Soit $(\Omega, F, \mathbb{P})$ un espace probabilisé, pour tout $\mu \in \mathcal{P}$, $(X^\mu_t)_{0\leq t\leq T}$ est solution de,

$$dX^\mu_t = b^\mu(t; X^\mu_t)\, dt + \sigma^\mu(t; X^\mu_t)\, dW_t, \tag{I.54}$$

Où $(W_t)_{0\leq t\leq T}$ est un mouvement brownien, et pour tout $\mu \in \mathcal{P}$, $b^\mu : [0,T]\times\mathbb{R} \to \mathbb{R}$ et $\sigma^\mu : [0,T]\times\mathbb{R} \to \mathbb{R}_+$.

En premier lieu, nous proposons des schémas splitting d'ordre 1 pour le cas des EDS avec bruit additif (σ^μ constant). En deuxième lieu, nous utilisons ce schéma, dans le cadre bruit

additif pour construire une variable de contrôle pour approximer l'espérance de la solution de l'EDS à chaque instant, $\mathbb{E}[X_t^\mu]$ pour tout $t \in [0, T]$, d'une manière rapide. En troisième lieu, nous généralisons et proposons des schémas splitting dans le cadre d'un bruit multiplicatif. Ces schémas sont étudiés sur un example d'EDS (le brownien géométrique) intégrant un terme de McKean. Nous obtenons ainsi un gain en temps de résolution de l'ordre de 10 fois plus rapide comparé à ce que donnerait une approximation de X_t par le schéma d'Euler Maruyama (pour le cas étudié). Ce gain peut être augmenté ou diminué dépendemement, principalement, du rang r utilisé.

CHAPTER II

INFLUENCE OF SAMPLING ON THE CONVERGENCE RATES OF GREEDY ALGORITHMS FOR PARAMETER-DEPENDENT RANDOM VARIABLES

Contents

Abstract

The main focus of this chapter is to provide a mathematical study of the algorithm proposed in [6] where the authors proposed a variance reduction technique for the computation of parameter-dependent expectations using a reduced basis paradigm. We study the effect of Monte-Carlo sampling on the theoretical properties of greedy algorithms. In particular, using concentration inequalities for the empirical measure in Wasserstein distance proved in [16], we provide sufficient conditions on the number of samples used for the computation of empirical variances at each iteration of the greedy procedure to guarantee that the resulting method algorithm is a weak greedy algorithm with high probability. These theoretical results are not fully practical and we therefore propose a heuristic procedure to choose the number of Monte-Carlo samples at each iteration, inspired from this theoretical study, which provides satisfactory results on several numerical test cases.

II.1 Introduction

The aim of this chapter is to provide a mathematical study of the algorithm proposed in [6] where the authors proposed a variance reduction technique for the computation of parameter-dependent expectations using a reduced basis paradigm.

More precisely, the problematic we are considering here is the following: let us denote by $\mathcal{P} \subset \mathbb{R}^m$ a set of parameter values. In several applications, it is of significant interest to be able to rapidly compute the expectation of a random variable of the form $f_\mu(Z)$ for a large numbers of values of the parameter $\mu \in \mathcal{P}$, where Z is a random vector and where for all $\mu \in \mathcal{P}$, f_μ is a real-valued function. In practice, such expectations may not be computable analytically and are approximated using empirical means involving a large number of random samples of the random vector Z. Variance reduction methods are commonly used in such contexts in order to reduce the computational cost of approximating these expectations by means of standard Monte-Carlo algorithms. Among these, control variates, which are chosen as approximations of the random variable $f_\mu(Z)$ the expectation of which can be easily computed, can yield to interesting gains in terms of computational cost, provided that the variance of the difference between $f_\mu(Z)$ and its approximation is small. The construction of efficient control variates for a given application is thus fundamental for the variance reduction technique to yield significant computational gains.

In [6], the authors proposed a general algorithm in order to construct a control variate for $f_\mu(Z)$ using a reduced basis paradim. More precisely, the approximation of $f_\mu(Z)$ is constructed as a linear combination of $f_{\mu_1}(Z), \cdots, f_{\mu_n}(Z)$ for some small integer $n \in \mathbb{N}^*$ and well-chosen values $\mu_1, \cdots, \mu_n \in \mathcal{P}$ of the parameters. The choice of n and of the values of the parameters stems from an iterative procedure, called a greedy algorithm, which consists at iteration $n \in \mathbb{N}$ to compute

$$\mu_{n+1} \in \operatorname*{argmax}_{\mu \in \mathcal{P}} \inf_{Z_n \in V_n} \operatorname{Var}\left[f_\mu(Z) - Z_n\right],$$

where $V_n := \operatorname{Span}\{f_{\mu_1}(Z), \cdots, f_{\mu_n}(Z)\}$. In the ideal (unpractical) case where variances can be exactly computed, the procedure boils down to a standard greedy algorithm in a Hilbert space [11]. It is now well-known [11] that such a greedy procedure provides a quasi-optimal set

of parameters $\mu_1, \cdots, \mu_n$ in the sense that the error

$$\sup_{\mu\in\mathcal{P}} \inf_{Z_n\in V_n} \mathrm{Var}\left[f_\mu(Z) - Z_n\right] = \inf_{Z_n\in V_n} \mathrm{Var}\left[f_{\mu_{n+1}}(Z) - Z_n\right]$$

is comparable to the so-called Kolmogorov n-width of the set $\{f_\mu(Z), \mu \in \mathcal{P}\}$, defined by

$$\sup_{\mu\in\mathcal{P}} \inf_{\substack{W_n \text{ vectorial subspace} \\ \dim W_n = n}} \inf_{Z_n\in W_n} \mathrm{Var}\left[f_\mu(Z) - Z_n\right].$$

In other words, the subspace V_n is a quasi-optimal subspace of dimension n for the approximation of random variables $f_\mu(Z)$ for $\mu \in \mathcal{P}$ in an L^2 norm sense.

However, in practice, variances cannot be computed exactly and have to be approximated by empirical means involving a finite number of samples of the random vector Z, which may be different from one iteration of the greedy algorithm to another. The main result of this chapter is to give theoretical lower bounds on the number of samples which have to be taken at each iteration of the greedy algorithm in order to guarantee that the resulting Monte-Carlo greedy algorithm enjoys quasi-optimality properties close to those of an ideal greedy algorithm with high probability.

The mathematical analysis of algorithms which combine randomness and greedy procedures is a quite recent and active field of research among the model-order reduction community. Let us mention here a few works in this direction in which different settings than the one we focus on here are considered. In [10], the authors consider the effect of randomly sampling the set of parameters in order to define random trial sets at each iteration of the greedy algorithm and prove that the obtained procedure enjoys remarkable approximation properties which remain very close to the approximation properties of a greedy algorithm where minimization problems at each iteration are defined over the whole set of parameters. In [26, 25, 1, 2], the authors propose randomized residual-based error estimators for parametrized equations, with a view to using them for the acceleration of greedy algorithms for reduced basis techniques. Let us finally mention that significant research efforts are devoted by many different groups to the improvement of randomized algorithms for Singular Value Decompositions [9], which plays a fundamental role for model-order reduction.

The outline of the chapter is the following. In Section II.2, we motivate the interest of greedy algorithms for the construction of control variates for variance reduction methods and recall some results of [7, 4, 11] on the mathematical analysis of greedy algorithms in Hilbert spaces. In Section II.3, we present the Monte-Carlo greedy algorithm, which is the main focus of this chapter, our main theoretical result and its proof. This theoretical result does not yield a fully practical algorithm. To alleviate this difficulty, we propose in Section II.4 a heuristic algorithm, inspired from the theoretical result, which provides satisfactory results on several test cases.

II.2 Motivation: greedy algorithms for reduced bases and variance reduction

II.2.1 Motivation: reduced basis control variate

The aim of this section is to present the motivation of our work, which aims at constructing control variates for reducing the variance of a Monte-Carlo estimator of the mean of parameter-dependent functions of random vectors.

Let us begin by introducing some notation. Let $d \in \mathbb{N}^*$, $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space and Z a $\mathbb{R}^d$-valued random vector with associated probability measure ν. For all $q \in \mathbb{N}^*$, we denote by

$$L^q_\nu(\mathbb{R}^d) := \left\{ f : \mathbb{R}^d \to \mathbb{R}, \quad \int_{\mathbb{R}^d} |f(x)|^q \, d\nu(x) < +\infty \right\}.$$

Let $\mathcal{C}(\mathbb{R}^d)$ denote the set of continuous real-valued functions defined on $\mathbb{R}^d$. Let $p \in \mathbb{N}^*$, $\mathcal{P} \subset \mathbb{R}^p$ be a set of parameter values, and for all $\mu \in \mathcal{P}$, let f_μ be an element of $\mathcal{C}(\mathbb{R}^d) \cap L^2_\nu(\mathbb{R}^d)$.

For all $f, g \in \mathcal{C}(\mathbb{R}^d)$, any $M \in \mathbb{N}^*$ and any collection $\overline{Z} := (Z_k)_{1 \le k \le M}$ of random vectors of $\mathbb{R}^d$, we define the empirical averages:

$$\begin{aligned} \mathbb{E}_{\overline{Z}}(f) &:= \frac{1}{M} \sum_{k=1}^{M} f\left(Z_k\right), \\ \mathrm{Cov}_{\overline{Z}}(f, g) &:= \mathbb{E}_{\overline{Z}}(fg) - \mathbb{E}_{\overline{Z}}(f)\mathbb{E}_{\overline{Z}}(g), \\ \mathrm{Var}_{\overline{Z}}(f) &:= \mathrm{Cov}_{\overline{Z}}(f, f). \end{aligned}$$

The aim of our work is to propose and analyse from a mathematical point of view a numerical method in order to efficiently construct control variates to reduce the variance of a Monte-Carlo estimator of $\mathbb{E}\left[f_\mu(Z)\right]$ for all $\mu \in \mathcal{P}$ using a Reduced Basis paradigm [17, 3, 22, 13], which was originally proposed in [6].

More precisely, let $M_{\text{small}}, M_{\text{ref}} \in \mathbb{N}^*$ and assume that $M_{\text{ref}} \gg M_{\text{small}}$. Let $\overline{Z}^{\text{ref}} := \left(Z_k^{\text{ref}}\right)_{1 \le k \le M_{\text{ref}}}$ and $\overline{Z}^{\text{small}} := \left(Z_k^{\text{small}}\right)_{1 \le k \le M_{\text{small}}}$ be two independent collections of iid random vectors distributed according to the law of Z and **independent of** Z.

Let us assume that we have selected N values of parameters $(\mu_1, \mu_2, ..., \mu_N) \in \mathcal{P}^N$ for some $N \in \mathbb{N}^*$ and assume that the empirical means $(\mathbb{E}_{\overline{Z}^{\text{ref}}}(f_{\mu_i}))_{1 \le i \le N}$ have been computed in an offline phase.

In an online phase, for all $\mu \in \mathcal{P}$, we can build an approximation of $\mathbb{E}\left[f_\mu(Z)\right]$, using a control variate which reads as $\overline{f}_\mu(Z)$ for some function $\overline{f}_\mu : \mathbb{R}^d \to \mathbb{R}$:

$$\mathbb{E}\left[f_\mu(Z)\right] \approx \mathbb{E}_{\overline{Z}^{\text{ref}}}(\overline{f}_\mu) + \mathbb{E}_{\overline{Z}^{\text{small}}}\left(f_\mu - \overline{f}_\mu\right). \tag{II.1}$$

Remark II.2.1. *Let us point out that the statistical error between* $\mathbb{E}_{\overline{Z}^{\text{ref}}}(\overline{f}_\mu)$ *and* $\mathbb{E}\left[\overline{f}_\mu(Z)\right]$ *is close to*

$$\sqrt{\frac{\mathrm{Var}\left[\overline{f}_\mu(Z)\right]}{M_{\text{ref}}}},$$

whereas the error between $\mathbb{E}_{\overline{Z}^{\text{small}}}\left(f_\mu - \overline{f}_\mu\right)$ *and* $\mathbb{E}\left[(f_\mu - \overline{f}_\mu)(Z)\right]$ *is of the order of*

$$\sqrt{\frac{\mathrm{Var}\left[\left(f_\mu - \overline{f}_\mu\right)(Z)\right]}{M_{\text{small}}}}.$$

The aim of the Monte-Carlo greedy algorithm studied in this chapter is to give an approximation of $\mathbb{E}\left[f_\mu(Z)\right]$ *with an error close to* $\sqrt{\frac{\mathrm{Var}[f_\mu(Z)]}{M_{\text{ref}}}}$ *within a much smaller computational time than the one required by the computation of* $\mathbb{E}_{\overline{Z}^{\text{ref}}}(f_\mu)$.

In the method studied here, the control variate function $\overline{f}_\mu$ is constructed as follows:

$$\overline{f}_\mu = \sum_{i=1}^{N} \lambda_i^\mu f_{\mu_i}$$

where $\lambda^\mu := (\lambda_i^\mu)_{1\le i\le N} \in \mathbb{R}^N$ is a solution of the linear system

$$A\lambda^\mu = b^\mu \tag{II.2}$$

where $A := (A_{ij})_{1\le i,j\le N} \in \mathbb{R}^{N\times N}$ and $b^\mu := (b_i^\mu)_{1\le i\le N} \in \mathbb{R}^N$ are defined as follows: for all $1 \le i, j \le N$,

$$A_{ij} = \mathrm{Cov}_{\overline{Z}^{\text{small}}}(f_{\mu_i}, f_{\mu_j}) \quad \text{and} \quad b_i^\mu = \mathrm{Cov}_{\overline{Z}^{\text{small}}}(f_\mu, f_{\mu_i}). \tag{II.3}$$

Equivalently, the vector λ^μ is a solution of the minimization problem

$$\lambda^\mu \in \underset{\lambda:=(\lambda_i)_{1\le i\le N}\in\mathbb{R}^N}{\mathrm{argmin}} \mathrm{Var}_{\overline{Z}^{\text{small}}}\left(f_\mu - \sum_{i=1}^{N} \lambda_i f_{\mu_i}\right).$$

Let us point out that λ^μ is a random vector which can be written as a deterministic function of $\overline{Z}^{\text{small}}$. In other words, λ^μ is measurable with respect to $\overline{Z}^{\text{small}}$. Remarking that $\mathbb{E}_{\overline{Z}^{\text{ref}}}(\overline{f}_\mu) = \sum_{i=1}^N \lambda_i^\mu \mathbb{E}_{\overline{Z}^{\text{ref}}}(f_{\mu_i})$, the computation of the approximation (II.1) of $\mathbb{E}\left[f_\mu(Z)\right]$ thus requires the following steps:

- **offline phase:** Compute $(\mathbb{E}_{\overline{Z}^{\text{ref}}}(f_{\mu_i}))_{1\le i\le N}$ (N empirical means with M_{ref} samples), $(\mathbb{E}_{\overline{Z}^{\text{small}}}(f_{\mu_i}))_{1\le i\le N}$ (N empirical means with M_{small} samples) and the matrix A (N^2 empirical covariances with M_{small} samples).
- **online phase:** For all $\mu \in \mathcal{P}$, compute b^μ (N empirical covariances with M_{small} samples) and solve the linear system (II.2) to obtain λ^μ. Then, compute the approximation (II.1) of $\mathbb{E}\left[f_\mu(Z)\right]$ as

$$\mathbb{E}\left[f_\mu(Z)\right] \approx \sum_{i=1}^{N} \lambda_i^\mu \mathbb{E}_{\overline{Z}^{\text{ref}}}(f_{\mu_i}) + \mathbb{E}_{\overline{Z}^{\text{small}}}\left(f_\mu\right) - \sum_{i=1}^{N} \lambda_i^\mu \mathbb{E}_{\overline{Z}^{\text{small}}}\left(f_{\mu_i}\right), \tag{II.4}$$

which requires $\mathcal{O}(N)$ elementary operations and the computation of one empirical mean with M_{small} samples.

Naturally, the approximation of $\mathbb{E}\left[f_\mu(Z)\right]$ given by (II.1) can be interesting from a computational point of view in terms of variance reduction only if $\mathrm{Var}\left[f_\mu(Z) - \overline{f}_\mu(Z)\right]$ is much smaller than $\mathrm{Var}\left[f_\mu(Z)\right]$. The following question thus naturally arises: how can the set of parameters $(\mu_1, \mu_2, .., \mu_N) \in \mathcal{P}^N$ be chosen in the offline phase in order to ensure that $\mathrm{Var}\left[f_\mu(Z) - \overline{f}_\mu(Z)\right]$ is as small as possible for any value of $\mu \in \mathcal{P}$?

Greedy algorithms stand as the state-of-the-art technique to construct such sets of snapshot parameters, enjoy very nice mathematical properties and are the backbone of the method proposed in [6] which we wish to analyze here. We present this family of algorithms and related existing theoretical convergence results in the next section.

II.2.2 Greedy algorithms for reduced basis

Let us recall here the results of [7, 4, 11] on the convergence rates of greedy algorithms for reduced bases, adapted to our context. Let us define

$$L^2_{\nu,0}(\mathbb{R}^d) := \left\{ g \in L^2_\nu(\mathbb{R}^d),\ \int_{\mathbb{R}^d} g \, d\nu = 0 \right\}.$$

It holds that $L^2_{\nu,0}(\mathbb{R}^d)$ is a Hilbert space, equipped with the scalar product $\langle \cdot, \cdot \rangle$ defined by

$$\forall g_1, g_2 \in L^2_{\nu,0}(\mathbb{R}^d), \quad \langle g_1, g_2 \rangle = \int_{\mathbb{R}^d} g_1 g_2 \, d\nu = \mathrm{Cov}\left[g_1(Z), g_2(Z)\right].$$

The associated norm is denoted by $\| \cdot \|$ and is given by

$$\forall g \in L^2_{\nu,0}(\mathbb{R}^d), \quad \|g\| = \left(\int_{\mathbb{R}^d} |g|^2 \, d\nu \right)^{1/2} = \sqrt{\mathrm{Var}\left[g(Z)\right]}.$$

For all $\mu \in \mathcal{P}$, let us define

$$g_\mu := f_\mu - \mathbb{E}\left[f_\mu(Z)\right] \tag{II.5}$$

and let us denote by

$$\mathcal{M} := \{ g_\mu,\ \mu \in \mathcal{P} \} \tag{II.6}$$

so that $\mathcal{M} \subset L^2_{\nu,0}(\mathbb{R}^d)$. Let us assume that $\mathcal{M}$ is a compact subset of $L^2_{\nu,0}(\mathbb{R}^d)$. For all $n \in \mathbb{N}^*$, we introduce the Kolmogorov n-width of the set $\mathcal{M}$ in $L^2_{\nu,0}(\mathbb{R}^d)$, defined by

$$\begin{aligned} d_n(\mathcal{M}) &:= \inf_{\substack{V_n \subset L^2_{\nu,0}(\mathbb{R}^d) \text{ subspace}, \\ \dim V_n = n}} \sup_{\mu \in \mathcal{P}} \inf_{g_n \in V_n} \sqrt{\mathrm{Var}\left[g_\mu(Z) - g_n(Z)\right]} \\ &= \inf_{\substack{V_n \subset L^2_{\nu,0}(\mathbb{R}^d) \text{ subspace}, \\ \dim V_n = n}} \sup_{\mu \in \mathcal{P}} \inf_{g_n \in V_n} \|g_\mu - g_n\|. \end{aligned}$$

Let $0 < \gamma < 1$ and consider the following *weak greedy algorithm* with parameter γ.

Weak-Greedy Algorithm

Initialization: Find $\mu_1 \in \mathcal{P}$ such that

$$\|g_{\mu_1}\|^2 \geq \gamma^2 \max_{\mu \in \mathcal{P}} \|g_\mu\|^2 . \tag{II.7}$$

Set $V_1 := \mathrm{Span}\{g_{\mu_1}\}$ and set $n = 2$.

Iteration $n \geq 2$**:** Find $\mu_n \in \mathcal{P}$ such that

$$\inf_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \left\| g_{\mu_n} - \sum_{i=1}^{n-1} \lambda_i g_{\mu_i} \right\|^2 \geq \gamma^2 \max_{\mu\in\mathcal{P}} \inf_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \left\| g_\mu - \sum_{i=1}^{n-1} \lambda_i g_{\mu_i} \right\|^2 , \tag{II.8}$$

Set $V_n := V_{n-1} + \mathrm{Span}\{g_{\mu_n}\} = \mathrm{Span}\{g_{\mu_1}, \cdots, g_{\mu_n}\}$.

For all $n \in \mathbb{N}^*$, the error associated with the n-dimensional subspace V_n given by the weak greedy algorithm is defined by

$$\sigma_n(\mathcal{M}) := \max_{\mu\in\mathcal{P}} \inf_{(\lambda_i)_{1\leq i\leq n}\in\mathbb{R}^n} \left\| g_\mu - \sum_{i=1}^{n} \lambda_i g_{\mu_i} \right\| .$$

The following result is then a direct corollary of the results proved in [11, Corollary 3.3].

Theorem II.2.2. *For all* $n \in \mathbb{N}^*$, $\sigma_n(\mathcal{M}) \leq \sqrt{2}\gamma^{-1} \min_{0\leq m<n} (d_m(\mathcal{M}))^{\frac{n-m}{n}}$. *In particular, for all* $n \in \mathbb{N}^*$, $\sigma_{2n}(\mathcal{M}) \leq \sqrt{2}\gamma^{-1}\sqrt{d_n(\mathcal{M})}$.

This result indicates that the weak greedy algorithm provides a practical way to construct a quasi-optimal sequence $(V_n)_{n\in\mathbb{N}^*}$ of finite dimensional subspaces of $L^2_{\nu,0}(\mathbb{R}^d)$.

Of course, the weak greedy algorithm introduced above cannot be implemented in practice since it requires at the n^{th} iteration of the algorithm the computation of the exact variances of $g_\mu(Z) - \sum_{i=1}^{n-1} \lambda_i g_{\mu_i}(Z)$ for $\mu, \mu_1, \cdots, \mu_{n-1} \in \mathcal{P}$ and $\lambda_1, \cdots, \lambda_{n-1} \in \mathbb{R}$, which is out of reach in our context. In practice, these quantities have to be approximated by Monte-Carlo estimators involving a finite number of samples of the random vector Z. The resulting greedy algorithm with Monte Carlo sampling is presented in Section II.3. The mathematical analysis of this algorithm is the main purpose of the present chapter.

For the sake of simplicity, in the rest of the chapter, we assume that for all $n \in \mathbb{N}^*$, $d_n(\mathcal{M}) > 0$.

II.3 Greedy algorithm with Monte-Carlo sampling

II.3.1 Presentation of the algorithm

Let us begin by presenting the greedy algorithm with Monte Carlo sampling.

Let $(M_n)_{n\in\mathbb{N}^*}$ be a sequence of integers, which represents the number of samples used at iteration n. For all $n \in \mathbb{N}^*$, let $\overline{Z}^n := (Z^n_k)_{1\leq k\leq M_n}$ be a collection of random vectors such that

$(Z_k^n)_{n\geq 1,\, 1\leq k\leq M_n}$ are independent and identically distributed according to the law of Z, and independent of Z. Let $\overline{Z}^{1:n} := \left(\overline{Z}^m\right)_{1\leq m\leq n}$ and $\overline{Z}^{1:\infty} := \left(\overline{Z}^n\right)_{n\in\mathbb{N}^*}$.

For any random functions g_1, g_2 with values in $L^2_{\nu,0}(\mathbb{R}^d)$, we define

$$\langle g_1, g_2\rangle_{\overline{Z}^{1:\infty}} := \mathrm{Cov}\left[g_1(Z), g_2(Z)\,\middle|\,\overline{Z}^{1:\infty}\right] \quad \text{and} \quad \|g_1\|_{\overline{Z}^{1:\infty}} := \sqrt{\mathrm{Var}\left[g_1(Z)\,\middle|\,\overline{Z}^{1:\infty}\right]}.$$

Let us make here an important remark. Since $\overline{Z}^{1:\infty}$ is a collection of random vectors which are all independent of Z, it holds that, for all $f, g \in L^2_{\nu,0}(\mathbb{R}^d)$, almost surely,

$$\begin{aligned}\langle f, g\rangle_{\overline{Z}^{1:\infty}} &= \mathrm{Cov}\left[f(Z), g(Z)\,\middle|\,\overline{Z}^{1:\infty}\right] = \mathrm{Cov}\left[f(Z), g(Z)\right] = \langle f, g\rangle,\\ \|g\|^2_{\overline{Z}^{1:\infty}} &= \mathrm{Var}\left[g(Z)\,\middle|\,\overline{Z}^{1:\infty}\right] = \mathrm{Var}[g(Z)] = \|g\|^2.\end{aligned}$$

Hence, almost surely, $\langle\cdot,\cdot\rangle_{\overline{Z}^{1:\infty}}$ defines a scalar product on $L^2_{\nu,0}(\mathbb{R}^d)$, which is a Hilbert space when equipped with this scalar product, and $\|\cdot\|_{\overline{Z}^{1:\infty}}$ is the associated norm.

The greedy algorithm with Monte-Carlo sampling reads as follows:

MC-Greedy Algorithm

Initialization: Find $\overline{\mu}_1 \in \mathcal{P}$ such that, almost surely,

$$\overline{\mu}_1 \in \operatorname*{argmax}_{\mu\in\mathcal{P}} \mathrm{Var}_{\overline{Z}^1}\left(g_\mu\right) \quad \text{and} \quad g_{\overline{\mu}_1} \neq 0. \tag{II.9}$$

Set $\overline{V}_1 := \mathrm{Span}\{g_{\overline{\mu}_1}\}$ and set $n = 2$.

Iteration $n \geq 2$**:** Find $\overline{\mu}_n \in \mathcal{P}$ such that, almost surely,

$$\overline{\mu}_n \in \operatorname*{argmax}_{\mu\in\mathcal{P}} \inf_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \mathrm{Var}_{\overline{Z}^n}\left(g_\mu - \sum_{i=1}^{n-1}\lambda_i g_{\overline{\mu}_i}\right) \quad \text{and} \quad g_{\overline{\mu}_n} \notin \overline{V}_{n-1}. \tag{II.10}$$

Set $\overline{V}_n := \overline{V}_{n-1} + \mathrm{Span}\{g_{\overline{\mu}_n}\} = \mathrm{Span}\{g_{\overline{\mu}_1}, \cdots, g_{\overline{\mu}_n}\}$.

Naturally, for all $n \in \mathbb{N}^*$, the parameter $\overline{\mu}_n$ and thus the finite-dimensional space $\overline{V}_n$ are $\overline{Z}^{1:n}$-measurable.

Let us first prove an auxiliary lemma.

Lemma II.3.1. *Almost surely, all the iterations of the MC-Greedy Algorithm are well-defined, in the sense that, for all $n \in \mathbb{N}^*$, there always exists at least one element $\overline{\mu}_n \in \mathcal{P}$ such that (II.9) (when $n = 1$) or (II.10) (when $n \geq 2$) is satisfied.*

Proof of Lemma II.3.1. Let us first consider the initialization step corresponding to $n = 1$. Two situations may a priori occur : either $\max_{\mu\in\mathcal{P}} \mathrm{Var}_{\overline{Z}^1}\left(g_\mu\right) > 0$ or $\max_{\mu\in\mathcal{P}} \mathrm{Var}_{\overline{Z}^1}\left(g_\mu\right) = 0$. In the first case, choosing $\overline{\mu}_1 \in \operatorname*{argmax}_{\mu\in\mathcal{P}} \mathrm{Var}_{\overline{Z}^1}\left(g_\mu\right)$ is sufficient to guarantee that $g_{\overline{\mu}_1} \neq 0$. Indeed, since

$\mathcal{M} \subset \mathcal{C}(\mathbb{R}^d)$ (remember that f_μ is continuous for all $\mu \in \mathcal{P}$, and hence so is g_μ), the fact that $\mathrm{Var}_{\overline{Z}^1}\left(g_{\overline{\mu}_1}\right) > 0$ necessarily implies that $\mathrm{Var}\left[g_{\overline{\mu}_1}(Z) \middle| \overline{Z}^{1:\infty}\right] > 0$ almost surely. Since $\overline{Z}^{1:\infty}$ is independent of Z and $\overline{\mu}_1$ is a $\overline{Z}^{1:\infty}$ measurable random variable, this implies that almost surely $g_{\overline{\mu}_1} \neq 0$.

In the second case, it then holds that $\mathrm{Var}_{\overline{Z}^1}(g_\mu) = 0$ for all $\mu \in \mathcal{P}$. Then, the assumption $d_1(\mathcal{M}) > 0$ implies that, almost surely, there exists at least one element $\overline{\mu}_1 \in \mathcal{P}$ such that $g_{\overline{\mu}_1} \neq 0$. In addition, $\overline{\mu}_1 \in \mathrm{argmax}_{\mu\in\mathcal{P}} \mathrm{Var}_{\overline{Z}^1}(g_\mu)$.

Using similar arguments and the fact that $d_n(\mathcal{M}) > 0$ for all $n \in \mathbb{N}^*$, it is easy to see that, almost surely, all the iterations of the MC-Greedy algorithm are well-defined, in particular for $n \geq 2$. □

Remark II.3.2. *We stress on the fact that the practical implementation of the MC-greedy algorithm does not require the knowledge of the value of $\mathbb{E}[f_\mu(Z)]$, even if $g_\mu = f_\mu - \mathbb{E}[f_\mu(Z)]$ for all $\mu \in \mathcal{P}$. Indeed, it holds that for all $g \in \mathcal{C}(\mathbb{R}^d)$, all $n \in \mathbb{N}^*$ and all $C \in \mathbb{R}$, $\mathrm{Var}_{\overline{Z}^n}(g) = \mathrm{Var}_{\overline{Z}^n}(g + C)$. Thus, for all $\mu \in \mathcal{P}$, $n \in \mathbb{N}^*$ and $\lambda := (\lambda_i)_{1\leq i\leq n-1} \in \mathbb{R}^{n-1}$,*

$$\mathrm{Var}_{\overline{Z}^1}(g_\mu) = \mathrm{Var}_{\overline{Z}^1}(f_\mu) \quad \text{and} \quad \mathrm{Var}_{\overline{Z}^n}\left(g_\mu - \sum_{i=1}^{n-1} \lambda_i g_{\overline{\mu}_i}\right) = \mathrm{Var}_{\overline{Z}^n}\left(f_\mu - \sum_{i=1}^{n-1} \lambda_i f_{\overline{\mu}_i}\right).$$

Thus, the MC-greedy algorithm naturally makes sense with a view to the construction of a reduced basis control variate for variance reduction as explained in Section II.2.1.

Remark II.3.3. *In practice, a discrete subset $\mathcal{P}_{\text{trial}} \subset \mathcal{P}$ has to be introduced. The optimization problems (II.9) and (II.10) have to be replaced respectively by*

$$\overline{\mu}_1 \in \operatorname*{argmax}_{\mu\in\mathcal{P}_{\text{trial}}} \mathrm{Var}_{\overline{Z}^1}(g_\mu) \quad \text{and} \quad g_{\overline{\mu}_1} \neq 0,$$

and

$$\overline{\mu}_n \in \operatorname*{argmax}_{\mu\in\mathcal{P}_{\text{trial}}} \inf_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \mathrm{Var}_{\overline{Z}^n}\left(g_\mu - \sum_{i=1}^{n-1} \lambda_i g_{\overline{\mu}_i}\right) \quad \text{and} \quad g_{\overline{\mu}_n} \notin \overline{V}_{n-1}.$$

The influence of the choice of the set $\mathcal{P}_{\text{trial}}$ on the mathematical properties of the MC-greedy algorithm is an important question which we do not address in our analysis for the sake of simplicity. For related discussion, we refer the reader to the work [10], where the authors study the mathematical properties of a greedy algorithm where the set $\mathcal{P}_{\text{trial}}$ depends on the iteration n of the greedy algorithm and is randomly chosen according to appropriate probability distributions defined on the set of parameters $\mathcal{P}$.

For all $n \in \mathbb{N}^*$, we also define

$$\widehat{\sigma}_{n-1}(\mathcal{M}) := \max_{\mu\in\mathcal{P}} \inf_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \sqrt{\mathrm{Var}\left[g_\mu(Z) - \sum_{i=1}^{n-1} \lambda_i g_{\overline{\mu}_i}(Z) \middle| \overline{Z}^{1:\infty}\right]}, \tag{II.11}$$

$$\overline{\sigma}_{n-1}(\mathcal{M}) := \inf_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \sqrt{\mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1} \lambda_i g_{\overline{\mu}_i}(Z) \middle| \overline{Z}^{1:\infty}\right]} \tag{II.12}$$

Let us point out here that $\widehat{\sigma}_{n-1}(\mathcal{M})$ is a random variable which is measurable with respect to $\overline{Z}^{1:(n-1)}$ whereas $\overline{\mu}_n$ and $\overline{\sigma}_{n-1}(\mathcal{M})$ are measurable with respect to $\overline{Z}^{1:n}$.

II.3.2 Main theoretical result

The aim of this section is to study the effect of Monte-Carlo sampling on the convergence of such a greedy algorithm. We consider here the probability space $(\Omega, \mathcal{A}(\overline{Z}^{1:\infty}), \mathbb{P})$ the probabilty space where $\mathcal{A}(\overline{Z}^{1:\infty})$ denotes the set of events that are measurable with respect to $\overline{Z}^{1:\infty}$. We prove, under appropriate assumptions on the probability density ν and on the set of functions $\mathcal{M} = \{g_\mu, \mu \in \mathcal{P}\}$, that for all $0 < \gamma < 1$, there exist explicit conditions on the sequence $(M_n)_{n\in\mathbb{N}^*}$ so that, with high probability, the MC-greedy algorithm is actually a weak greedy algorithm with parameter γ. More precisely, under this set of assumptions, we prove that, with high probability, it holds that for all $n \in \mathbb{N}^*$,

$$\overline{\sigma}_{n-1}(\mathcal{M}) \geq \gamma \widehat{\sigma}_{n-1}(\mathcal{M}).$$

Let us now present the set of assumptions we make on ν and on the set $\mathcal{M} = \{g_\mu,\ \mu \in \mathcal{P}\}$ for our main result to hold.

From now on, we make the following assumption on the probability distribution ν.

Assumption (A): The probability law ν is such that there exist $\alpha > 1$ and $\beta > 0$ such that

$$\int_{\mathbb{R}} e^{\beta|x|^\alpha}\, d\nu(x) < +\infty.$$

Let us denote by $\mathcal{L}$ the set of Lipschitz functions of $\mathbb{R}^d$ and for all $f \in \mathcal{L}$, let us denote by $\|f\|_{\mathcal{L}}$ its Lipschitz constant. In the sequel, we denote by $\phi : \mathbb{R}_+^* \to \mathbb{R}_+^*$ the function defined by

$$\forall \kappa \in \mathbb{R}_+^*, \quad \phi(\kappa) := \begin{cases} \kappa^2 \mathbb{1}_{\kappa\leq 1} + \kappa^\alpha \mathbb{1}_{\kappa>1} & \text{if } d = 1, \\ (\kappa/\log(2+1/\kappa)^2)\mathbb{1}_{\kappa\leq 1} + \kappa^\alpha \mathbb{1}_{\kappa>1} & \text{if } d = 2, \\ \kappa^d \mathbb{1}_{\kappa\leq 1} + \kappa^\alpha \mathbb{1}_{\kappa>1} & \text{if } d \geq 3. \end{cases} \tag{II.13}$$

A key ingredient in our analysis is the use of concentration inequalities in the Wasserstein-1 distance between a probability distribution and its empirical measure proved in [5, 16]. Let us recall here a direct corollary of Theorem 2 of [16], which is the backbone of our analysis.

Corollary II.3.4. *Let us assume that ν satisfies assumption (A). Then, there exist positive constants c, C depending only on ν, d, α and β, such that, for all $M \in \mathbb{N}^*$, all $\overline{Z} := (Z_k)_{1\leq k\leq M}$ iid random vectors distributed according to ν and all $\kappa > 0$, it holds that*

$$\mathbb{P}\left[\mathcal{T}_1\left(\overline{Z}\right) \geq \kappa\right] \leq Ce^{-cM\phi(\kappa)},$$

where

$$\mathcal{T}_1\left(\overline{Z}\right) := \sup_{f\in\mathcal{L};\|f\|_{\mathcal{L}}\leq 1} \left|\mathbb{E}[f(Z)] - \mathbb{E}_{\overline{Z}}(f)\right|.$$

Remark II.3.5. *We would like to mention here that other concentration inequalities are stated in Theorem 2 of [16] under different sets of assumptions than (A) on the probability law ν. In particular, weaker concentration inequalities may be obtained when ν only has some finite polynomial moments. Our analysis can then be easily adapted to these different settings but we restrict ourselves here to a framework where ν satisfies Assumption (A) for the sake of clarity.*

We finally make the following set of assumptions on $\mathcal{M}$ defined in (II.6).

Assumption (B): The set $\mathcal{M}$ satisfies the four conditions:

(B1) $\mathcal{M}$ is a compact subset of $L^2_{\nu,0}(\mathbb{R}^d)$ and let $K_2 := \sup_{\mu\in\mathcal{P}} \|g_\mu\| < \infty$;

(B2) $\mathcal{M} \subset \mathcal{L}$ and $K_{\mathcal{L}} := \sup_{\mu\in\mathcal{P}} \|g_\mu\|_{\mathcal{L}} < +\infty$;

(B3) $\mathcal{M} \subset L^\infty(\mathbb{R}^d)$ and $K_\infty := \sup_{\mu\in\mathcal{P}} \|g_\mu\|_{L^\infty} < +\infty$;

(B4) for all $n \in \mathbb{N}^*$, $d_n(\mathcal{M}) > 0$.

Before presenting our main result, we need to introduce some additional notation. Using Lemma II.3.1, we can almost surely define the sequence $(\overline{g}_n)_{n\in\mathbb{N}^*}$ as the orthonormal family of $L^2_{\nu,0}(\mathbb{R}^d)$ obtained by a Gram-Schmidt orthonormalization procedure (for $\|\cdot\|_{\overline{Z}^{1:\infty}}$) from the family $(g_{\overline{\mu}_n})_{n\in\mathbb{N}^*}$. More precisely, we define

$$\overline{g}_1 := \frac{g_{\overline{\mu}_1}}{\sqrt{\mathrm{Var}\left[g_{\overline{\mu}_1}(Z)\,\middle|\,\overline{Z}^{1:\infty}\right]}}.$$

Moreover, for all $n \geq 2$, et $\overline{\lambda}^n := \left(\overline{\lambda}^n_i\right)_{1\leq i\leq n-1} \in \mathbb{R}^{n-1}$ be a solution to the minimization problem

$$\overline{\lambda}^n \in \underset{\lambda:=(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}}{\mathrm{argmin}} \mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1}\lambda_i\overline{g}_i(Z)\,\middle|\,\overline{Z}^{1:\infty}\right].$$

Then it holds that

$$\overline{g}_n := \frac{g_{\overline{\mu}_n} - \sum_{i=1}^{n-1}\overline{\lambda}^n_i\overline{g}_i}{\sqrt{\mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1}\overline{\lambda}^n_i\overline{g}_i(Z)\,\middle|\,\overline{Z}^{1:\infty}\right]}}.$$

As a consequence, it always holds that $\overline{V}_n = \mathrm{Span}\left\{g_{\overline{\mu}_1},\cdots,g_{\overline{\mu}_n}\right\} = \mathrm{Span}\left\{\overline{g}_1,\cdots,\overline{g}_n\right\}$. Moreover, $\overline{g}_n$ is $\overline{Z}^{1:n}$-measurable.

We are now in position to state our main result, the proof of which is postponed to Section II.3.3.

Theorem II.3.6. *Let $0 < \delta < 1$ and $(\delta_n)_{n\in\mathbb{N}^*} \subset (0,1)^{\mathbb{N}^*}$ be a sequence of numbers satisfying $\Pi_{n\in\mathbb{N}^*}(1-\delta_n) \geq 1-\delta$. Let us assume that $\mathcal{M}$ satisfies assumption (B) and that ν satisfies assumption (A). Let $C, c > 0$ be the constants defined in Corollary II.3.4.*

For all $n \in \mathbb{N}^*$, let

$$K^n_\infty := \max\left(K_\infty, \|\overline{g}_1\|_{L^\infty},\cdots,\|\overline{g}_n\|_{L^\infty}\right) \quad \text{and} \quad K^n_{\mathcal{L}} := \max\left(K_{\mathcal{L}}, \|\overline{g}_1\|_{\mathcal{L}},\cdots,\|\overline{g}_n\|_{\mathcal{L}}\right). \tag{II.14}$$

Let us assume that there exists $0 < \gamma < 1$ such that for all $n \in \mathbb{N}^*$, $M_n \in \mathbb{N}^*$ is a $\overline{Z}^{1:(n-1)}$ measurable random variable which satisfies almost surely the following condition:

$$\forall n \geq 1, \quad M_n \geq -\ln\left(\frac{\delta_n}{C}\right)\frac{1}{c\phi\left(\kappa_{n-1}\right)}, \tag{II.15}$$

where κ_{n-1} is a deterministic function of $\overline{Z}^{1:(n-1)}$, defined by

$$\kappa_0 := \frac{(1-\gamma^2)\,\widehat{\sigma}_0(\mathcal{M})^2}{8K_\infty K_{\mathcal{L}}}; \tag{II.16}$$

and

$$\forall n \geq 2, \quad \kappa_{n-1} := \frac{\min\left(\frac{1}{2(n-1)}, \quad \frac{(1-\gamma^2)\widehat{\sigma}_{n-1}(\mathcal{M})^2}{n\left(9K_2^2+4\right)}\right)}{6K_\infty^{n-1}K_{\mathcal{L}}^{n-1}}. \tag{II.17}$$

Then, for all $n \in \mathbb{N}^*$, it holds that

$$\mathbb{P}\left[\overline{\sigma}_{n-1}(\mathcal{M}) \geq \gamma\widehat{\sigma}_{n-1}(\mathcal{M}) \,\middle|\, \overline{Z}^{1:(n-1)}\right] \geq 1-\delta_n. \tag{II.18}$$

As a consequence, denoting by $\mathcal{G}_n$ the event $\overline{\sigma}_{n-1}(\mathcal{M}) \geq \gamma\widehat{\sigma}_{n-1}(\mathcal{M})$ for all $n \in \mathbb{N}^*$, it holds that

$$\mathbb{P}\left[\bigcap_{n\in\mathbb{N}^*} \mathcal{G}_n\right] \geq 1-\delta. \tag{II.19}$$

Thus, it then holds that the MC-greedy algorithm is a weak greedy algorithm with parameter γ and norm $\|\cdot\|_{\overline{Z}^{1:\infty}}$ with probability at least $1-\delta$.

We state here a direct corollary of Theorem II.3.6, the proof of which is given below.

Corollary II.3.7. *Under the assumptions of Theorem II.3.6, with probability $1-\delta$, it holds that for all $n \in \mathbb{N}^*$,*

$$\widehat{\sigma}_n(\mathcal{M}) \leq \sqrt{2}\gamma^{-1} \min_{1\leq m<n} \left(d_m(\mathcal{M})\right)^{\frac{n-m}{n}}. \tag{II.20}$$

In particular, with probability $1-\delta$, it holds that

$$\forall n \in \mathbb{N}^*, \quad \widehat{\sigma}_{2n}(\mathcal{M}) \leq \sqrt{2}\gamma^{-1}\sqrt{d_n(\mathcal{M})}. \tag{II.21}$$

Proof. With probability $1-\delta$, the MC-greedy algorithm is a weak greedy algorithm with parameter γ and norm $\|\cdot\|_{\overline{Z}^{1:\infty}}$. Thus, since for all $n \in \mathbb{N}^*$, $\overline{\mu}_n$ is a $\overline{Z}^{1:\infty}$ measurable random variable, if such an event is realized, using Theorem II.2.2, it holds that for all $n \in \mathbb{N}^*$

$$\widehat{\sigma}_n(\mathcal{M}) \leq \sqrt{2}\gamma^{-1} \min_{1\leq m<n} \left(d_m^{\overline{Z}^{1:\infty}}(\mathcal{M})\right)^{\frac{n-m}{n}},$$

where for all $n \in \mathbb{N}^*$,

$$\begin{aligned} d_n^{\overline{Z}^{1:\infty}}(\mathcal{M}) &:= \inf_{\substack{V_n \subset L^2_{\nu,0}(\mathbb{R}^d) \text{ subspace,} \\ \dim V_n = n}} \sup_{\mu\in\mathcal{P}} \inf_{g_n\in V_n} \sqrt{\mathrm{Var}\left[g_\mu(Z) - g_n(Z) \,\middle|\, \overline{Z}^{1:\infty}\right]} \\ &= \inf_{\substack{V_n \subset L^2_{\nu,0}(\mathbb{R}^d) \text{ subspace,} \\ \dim V_n = n}} \sup_{\mu\in\mathcal{P}} \inf_{g_n\in V_n} \sqrt{\mathrm{Var}\left[g_\mu(Z) - g_n(Z)\right]} \\ &= d_n(\mathcal{M}). \end{aligned}$$

Hence, we obtain (II.20), and (II.21) as a consequence. □

Some remarks are in order here.

Remark II.3.8. *Note that, since the random variables K_∞^{n-1}, $K_{\mathcal{L}}^{n-1}$ and $\widehat{\sigma}_{n-1}(\mathcal{M})$ are measurable with respect to $\overline{Z}^{1:(n-1)}$, κ_{n-1} is also measurable with respect to $\overline{Z}^{1:(n-1)}$.*

Remark II.3.9. *A natural question is then the following: can Theorem II.3.6 be used (at least in principle) to design a* constructive strategy *to choose a number of samples M_n, so that the MC-greedy algorithm can be guaranteed to be a weak greedy algorithm with parameter γ? This can indeed be done in principle using the following remark: for all $n \in \mathbb{N}^*$, the quantity $\widehat{\sigma}_{n-1}(\mathcal{M})$ defined by (II.11) cannot be computed in practice since variances cannot be computed exactly for any parameter $\mu \in \mathcal{P}$. However, almost surely, it holds that $\overline{\sigma}_{n-1}(\mathcal{M})$ defined by (II.12) satisfies $\overline{\sigma}_{n-1}(\mathcal{M}) \leq \widehat{\sigma}_{n-1}(\mathcal{M})$. Let us recall that $\overline{\sigma}_{n-1}(\mathcal{M})$ depends on $\overline{Z}^{1:n}$, whereas $\widehat{\sigma}_{n-1}(\mathcal{M})$ only depends on $\overline{Z}^{1:(n-1)}$. Since ϕ is an increasing function, this implies that, if the sequence $(M_n)_{n\in\mathbb{N}^*}$ satisfies condition (II.15) where $\widehat{\sigma}_0(\mathcal{M})$ is replaced by $\overline{\sigma}_0(\mathcal{M})$ in (II.16) and $\widehat{\sigma}_{n-1}(\mathcal{M})$ is replaced by $\overline{\sigma}_{n-1}(\mathcal{M})$ in (II.17), the assumptions of Theorem II.3.6 are satisfied. Besides, it is reasonable to expect in this case that $\overline{\sigma}_{n-1}(\mathcal{M})$ should provide a reasonable approximation of $\widehat{\sigma}_{n-1}(\mathcal{M})$ since, from Theorem II.3.6, $\overline{\sigma}_{n-1}(\mathcal{M}) \geq \gamma\widehat{\sigma}_{n-1}(\mathcal{M})$ with high probability.*

Unfortunately, we will see that such an approach is not viable in practice, because it leads to much too large values of M_n for small values of n for the MC-greedy algorithm to be interesting with a view to the variance reduction method explained in Section II.2.1. That is why in Section II.4, we will present numerical results with heuristic ways to choose values of $(M_n)_{n\in\mathbb{N}^*}$ which are not theoretically guaranteeed, but which nevertheless yield satisfactory numerical results in several test cases.

II.3.3 Proof of Theorem II.3.6

The aim of this section is to prove Theorem II.3.6. For all $n \in \mathbb{N}^*$, we denote by $\mathcal{G}_n$ the event $\overline{\sigma}_n(\mathcal{M}) \geq \gamma\widehat{\sigma}_n(\mathcal{M})$.

Let us begin by proving some intermediate results which will be used later. We first need the following auxiliary lemma.

Lemma II.3.10. *Let $n \in \mathbb{N}^*$. Then, almost surely,*

$$\sup_{\substack{f\ \overline{Z}^{1:\infty}\text{-measurable random function} \\ \text{such that } \|f\|_{\mathcal{L}} \leq 1 \text{ almost surely}}} \left|\mathbb{E}\left[f(Z)|\overline{Z}^{1:\infty}\right] - \mathbb{E}_{\overline{Z}^n}(f)\right| = \sup_{f\in\mathcal{L};\|f\|_{\mathcal{L}}\leq 1} \left|\mathbb{E}\left[f(Z)\right] - \mathbb{E}_{\overline{Z}^n}(f)\right|.$$

Proof. On the one hand, it is obvious to check that

$$\sup_{f\in\mathcal{L};\|f\|_{\mathcal{L}}\leq 1} \left|\mathbb{E}\left[f(Z)\right] - \mathbb{E}_{\overline{Z}^n}(f)\right| \leq \sup_{\substack{f\ \overline{Z}^{1:\infty}\text{-measurable random function} \\ \text{such that } \|f\|_{\mathcal{L}} \leq 1 \text{ almost surely}}} \left|\mathbb{E}\left[f(Z)|\overline{Z}^{1:\infty}\right] - \mathbb{E}_{\overline{Z}^n}(f)\right|.$$

On the other hand, for any $\overline{Z}^{1:\infty}$-measurable random function f such that $\|f\|_{\mathcal{L}} \leq 1$ almost surely, it holds that, almost surely, since $\overline{Z}^{1:\infty}$ is independent of Z, $\mathbb{E}[f(Z)|\overline{Z}^{1:\infty}] = \mathbb{E}_Z[f(Z)]$

where the index Z in $\mathbb{E}_Z$ indicates that the expectation is only taken with respect to Z, and thus

$$\left|\mathbb{E}\left[f(Z)|\overline{Z}^{1:\infty}\right] - \mathbb{E}_{\overline{Z}^n}(f)\right| \leq \sup_{f\in\mathcal{L};\|f\|_{\mathcal{L}}\leq 1} \left|\mathbb{E}\left[f(Z)\right] - \mathbb{E}_{\overline{Z}^n}(f)\right|.$$

Hence the result. □

We start by considering the case of the initialization of the MC-greedy algorithm.

Lemma II.3.11. *Let $0 < \gamma < 1$. Then, it holds that almost surely,*

$$\mathbb{P}\left[\mathrm{Var}\left[g_{\overline{\mu}_1}(Z)\,\middle|\,\overline{Z}^{1:\infty}\right] \geq \gamma^2 \max_{\mu\in\mathcal{P}} \mathrm{Var}\left[g_\mu(Z)\,\middle|\,\overline{Z}^{1:\infty}\right]\right] \geq 1-\delta_1. \tag{II.22}$$

As a consequence, $\mathbb{P}\left[\mathcal{G}_1\right] \geq 1-\delta_1$ and (II.18) holds for $n=1$.

Proof. Let $\widehat{\mu}_1 \in \mathcal{P}$ such that

$$\widehat{\sigma}_0(\mathcal{M})^2 = \max_{\mu\in\mathcal{P}} \mathrm{Var}\left[g_\mu(Z)\,\middle|\,\overline{Z}^{1:\infty}\right] = \mathrm{Var}\left[g_{\widehat{\mu}_1}(Z)|\overline{Z}^{1:\infty}\right].$$

Inequality (II.22) holds provided that

$$\mathbb{P}\left[\left(\mathrm{Var}\left[g_{\widehat{\mu}_1}(Z)|\overline{Z}^{1:\infty}\right] - \mathrm{Var}\left[g_{\overline{\mu}_1}(Z)\,\middle|\,\overline{Z}^{1:\infty}\right]\right) > \epsilon\widehat{\sigma}_0(\mathcal{M})^2\right] \leq \delta_1,$$

with $\epsilon := (1-\gamma^2)$. Almost surely, since $\overline{\mu}_1 \in \mathrm{argmax}_{\mu\in\mathcal{P}} \mathrm{Var}_{\overline{Z}^1}(g_\mu)$, it holds that

$$\begin{aligned}
&\mathrm{Var}\left[g_{\widehat{\mu}_1}(Z)|\overline{Z}^{1:\infty}\right] - \mathrm{Var}\left[g_{\overline{\mu}_1}(Z)\,\middle|\,\overline{Z}^{1:\infty}\right] \\
&= \mathrm{Var}\left[g_{\widehat{\mu}_1}(Z)|\overline{Z}^{1:\infty}\right] - \mathrm{Var}_{\overline{Z}^1}\left(g_{\widehat{\mu}_1}\right) + \mathrm{Var}_{\overline{Z}^1}\left(g_{\widehat{\mu}_1}\right) - \mathrm{Var}_{\overline{Z}^1}\left(g_{\overline{\mu}_1}\right) + \mathrm{Var}_{\overline{Z}^1}\left(g_{\overline{\mu}_1}\right) - \mathrm{Var}\left[g_{\overline{\mu}_1}(Z)\,\middle|\,\overline{Z}^{1:\infty}\right] \\
&\leq \mathrm{Var}\left[g_{\widehat{\mu}_1}(Z)|\overline{Z}^{1:\infty}\right] - \mathrm{Var}_{\overline{Z}^1}\left(g_{\widehat{\mu}_1}\right) + \mathrm{Var}_{\overline{Z}^1}\left(g_{\overline{\mu}_1}\right) - \mathrm{Var}\left[g_{\overline{\mu}_1}(Z)\,\middle|\,\overline{Z}^{1:\infty}\right] \\
&= \mathbb{E}\left[|g_{\widehat{\mu}_1}|^2(Z)|\overline{Z}^{1:\infty}\right] - \mathbb{E}_{\overline{Z}^1}\left(|g_{\widehat{\mu}_1}|^2\right) + \mathbb{E}_{\overline{Z}^1}\left(g_{\widehat{\mu}_1}\right)^2 - \mathbb{E}\left[g_{\widehat{\mu}_1}(Z)\,\middle|\,\overline{Z}^{1:\infty}\right]^2 \\
&\quad - \mathbb{E}\left[|g_{\overline{\mu}_1}|^2(Z)|\overline{Z}^{1:\infty}\right] + \mathbb{E}_{\overline{Z}^1}\left(|g_{\overline{\mu}_1}|^2\right) - \mathbb{E}_{\overline{Z}^1}\left(g_{\overline{\mu}_1}\right)^2 + \mathbb{E}\left[g_{\overline{\mu}_1}(Z)|\overline{Z}^{1:\infty}\right]^2 \\
&\leq \left|\mathbb{E}\left[|g_{\widehat{\mu}_1}|^2(Z)|\overline{Z}^{1:\infty}\right] - \mathbb{E}_{\overline{Z}^1}\left(|g_{\widehat{\mu}_1}|^2\right)\right| + 2K_\infty\left|\mathbb{E}_{\overline{Z}^1}\left(g_{\widehat{\mu}_1}\right) - \mathbb{E}\left[g_{\widehat{\mu}_1}(Z)\right]\right| \\
&\quad + \left|\mathbb{E}\left[|g_{\overline{\mu}_1}|^2(Z)|\overline{Z}^{1:\infty}\right] - \mathbb{E}_{\overline{Z}^1}\left(|g_{\overline{\mu}_1}|^2\right)\right| + 2K_\infty\left|\mathbb{E}_{\overline{Z}^1}\left(g_{\overline{\mu}_1}\right) - \mathbb{E}\left[g_{\overline{\mu}_1}(Z)|\overline{Z}^{1:\infty}\right]\right|, \\
&\leq 2K_\infty K_{\mathcal{L}} \\
&\times \left(\left|\mathbb{E}\left[\frac{|g_{\widehat{\mu}_1}|^2}{2K_\infty K_{\mathcal{L}}}(Z)|\overline{Z}^{1:\infty}\right] - \mathbb{E}_{\overline{Z}^1}\left(\frac{|g_{\widehat{\mu}_1}|^2}{2K_\infty K_{\mathcal{L}}}\right)\right| + \left|\mathbb{E}\left[\frac{|g_{\overline{\mu}_1}|^2}{2K_\infty K_{\mathcal{L}}}(Z)|\overline{Z}^{1:\infty}\right] - \mathbb{E}_{\overline{Z}^1}\left(\frac{|g_{\overline{\mu}_1}|^2}{2K_\infty K_{\mathcal{L}}}\right)\right| \right.\\
&\quad \left. + \left|\mathbb{E}_{\overline{Z}^1}\left(\frac{g_{\overline{\mu}_1}}{K_{\mathcal{L}}}\right) - \mathbb{E}\left[\frac{g_{\overline{\mu}_1}}{K_{\mathcal{L}}}(Z)|\overline{Z}^{1:\infty}\right]\right| + \left|\mathbb{E}_{\overline{Z}^1}\left(\frac{g_{\widehat{\mu}_1}}{K_{\mathcal{L}}}\right) - \mathbb{E}\left[\frac{g_{\widehat{\mu}_1}}{K_{\mathcal{L}}}(Z)|\overline{Z}^{1:\infty}\right]\right|\right).
\end{aligned}$$

It holds that for all $\mu\in\mathcal{P}$, $\||g_\mu|^2\|_{\mathcal{L}} \leq 2K_\infty K_{\mathcal{L}}$. Indeed, for all $x, y \in \mathbb{R}^d$, we have

$$||g_\mu|^2(x) - |g_\mu|^2(y)| = |(g_\mu(x)+g_\mu(y))(g_\mu(x)-g_\mu(y))| \leq 2K_\infty K_{\mathcal{L}}|x-y|.$$

Thus, almost surely, it holds that

$$\mathrm{Var}\left[g_{\widehat{\mu}_1}(Z)|\overline{Z}^{1:\infty}\right] - \mathrm{Var}\left[g_{\overline{\mu}_1}(Z)\left|\overline{Z}^{1:\infty}\right.\right] \leq 8K_\infty K_{\mathcal{L}} \sup_{f\in\mathcal{L};\|f\|_{\mathcal{L}}\leq 1} \left|\mathbb{E}\left[f(Z)|\overline{Z}^{1:\infty}\right] - \mathbb{E}_{\overline{Z}^1}(f)\right|.$$

Then, using Lemma II.3.10, we obtain that, almost surely,

$$\mathrm{Var}\left[g_{\widehat{\mu}_1}(Z)|\overline{Z}^{1:\infty}\right] - \mathrm{Var}\left[g_{\overline{\mu}_1}(Z)\left|\overline{Z}^{1:\infty}\right.\right] \leq 8K_\infty K_{\mathcal{L}} \sup_{f\in\mathcal{L};\|f\|_{\mathcal{L}}\leq 1} \left|\mathbb{E}\left[f(Z)\right] - \mathbb{E}_{\overline{Z}^1}(f)\right|.$$

Thus, using Theorem II.3.4, the assumption on M_1 and the definition of κ_0, we obtain that

$$\mathbb{P}\left[\sup_{f\in\mathcal{L};\|f\|_{\mathcal{L}}\leq 1} \left|\mathbb{E}\left[f(Z)\right] - \mathbb{E}_{\overline{Z}^1}(f)\right| \geq \kappa_0\right] \leq Ce^{-c\phi(\kappa_0)} \leq \delta_1.$$

Hence the desired result. □

We now turn to the case of the n^{th} iteration of the algorithm, with $n \geq 2$, that we analyze in the next two lemmas.

Lemma II.3.12. *Let $n \geq 2$. Let us denote by*

$$\mathcal{M}^{n-1} := \mathcal{M} \cup \{\overline{g}_1, \cdots, \overline{g}_{n-1}\}. \tag{II.23}$$

Then, for all $\epsilon > 0$, it holds that, almost surely,

$$\mathbb{P}\left[\sup_{g,h\in\mathcal{M}^{n-1}} \left|\mathrm{Cov}\left[g(Z),h(Z)\left|\overline{Z}^{1:\infty}\right.\right] - \mathrm{Cov}_{\overline{Z}^n}(g,h)\right| \geq \epsilon \middle| \overline{Z}^{1:(n-1)}\right] \leq Ce^{-cM_n\phi\left(\frac{\epsilon}{6K_\infty^{n-1}K_{\mathcal{L}}^{n-1}}\right)},$$

where $K_{\mathcal{L}}^{n-1}$ and K_∞^{n-1} are defined by (II.14).

Proof. For all $g, h \in \mathcal{M}^{n-1}$, it holds that, almost surely,

$$\begin{aligned}
&\left|\mathrm{Cov}\left[g(Z),h(Z)\left|\overline{Z}^{1:\infty}\right.\right] - \mathrm{Cov}_{\overline{Z}^n}(g,h)\right| \\
&\leq \left|\mathbb{E}\left[g(Z)h(Z)\left|\overline{Z}^{1:\infty}\right.\right] - \mathbb{E}_{\overline{Z}^n}(gh)\right| \\
&\quad + K_\infty^{n-1}\left(\left|\mathbb{E}\left[g(Z)\left|\overline{Z}^{1:\infty}\right.\right] - \mathbb{E}_{\overline{Z}^n}(g)\right| + \left|\mathbb{E}\left[h(Z)\left|\overline{Z}^{1:\infty}\right.\right] - \mathbb{E}_{\overline{Z}^n}(h)\right|\right) \\
&\leq 2K_\infty^{n-1}K_{\mathcal{L}}^{n-1}\left(\left|\mathbb{E}\left[\frac{gh}{2K_\infty^{n-1}K_{\mathcal{L}}^{n-1}}(Z)\left|\overline{Z}^{1:\infty}\right.\right] - \mathbb{E}_{\overline{Z}^n}\left(\frac{gh}{2K_\infty^{n-1}K_{\mathcal{L}}^{n-1}}\right)\right|\right. \\
&\quad \left. + \left|\mathbb{E}\left[\frac{g}{2K_{\mathcal{L}}^{n-1}}(Z)\left|\overline{Z}^{1:\infty}\right.\right] - \mathbb{E}_{\overline{Z}^n}\left(\frac{g}{2K_{\mathcal{L}}^{n-1}}\right)\right| + \left|\mathbb{E}\left[\frac{h}{2K_{\mathcal{L}}^{n-1}}(Z)\left|\overline{Z}^{1:\infty}\right.\right] - \mathbb{E}_{\overline{Z}^n}\left(\frac{h}{2K_{\mathcal{L}}^{n-1}}\right)\right|\right).
\end{aligned}$$

For all $g, h \in \mathcal{M}^{n-1}$, it holds that

$$\left\|\frac{gh}{2K_\infty^{n-1}K_{\mathcal{L}}^{n-1}}\right\|_{\mathcal{L}} \leq 1 \quad \text{and} \quad \left\|\frac{g}{2K_{\mathcal{L}}^{n-1}}\right\|_{\mathcal{L}} \leq 1.$$

This implies that, almost surely,

$$\sup_{g,h\in\mathcal{M}^{n-1}}\left|\mathrm{Cov}\left[g(Z),h(Z)\left|\overline{Z}^{1:\infty}\right.\right]-\mathrm{Cov}_{\overline{Z}^n}(g,h)\right|\leq 6K_\infty^{n-1}K_{\mathcal{L}}^{n-1}\sup_{f\in\mathcal{L},\,\|f\|_{\mathcal{L}}\leq 1}\left|\mathbb{E}\left[f(Z)\left|\overline{Z}^{1:\infty}\right.\right]-\mathbb{E}_{\overline{Z}^n}(f)\right|.$$

Using Lemma II.3.10, this yields that, almost surely,

$$\sup_{g,h\in\mathcal{M}^{n-1}}\left|\mathrm{Cov}\left[g(Z),h(Z)\left|\overline{Z}^{1:\infty}\right.\right]-\mathrm{Cov}_{\overline{Z}^n}(g,h)\right|\leq 6K_\infty^{n-1}K_{\mathcal{L}}^{n-1}\sup_{f\in\mathcal{L},\,\|f\|_{\mathcal{L}}\leq 1}\left|\mathbb{E}\left[f(Z)\right]-\mathbb{E}_{\overline{Z}^n}(f)\right|.$$

We finally obtain, using Corollary II.3.4, that

$$\begin{aligned}
&\mathbb{P}\left[\sup_{g,h\in\mathcal{M}^{n-1}}\left|\mathrm{Cov}\left[g(Z),h(Z)\left|\overline{Z}^{1:\infty}\right.\right]-\mathrm{Cov}_{\overline{Z}^n}(g,h)\right|>\epsilon\middle|\overline{Z}^{1:(n-1)}\right]\\
&\leq\mathbb{P}\left[\sup_{f\in\mathcal{L},\,\|f\|_{\mathcal{L}}\leq 1}\left|\mathbb{E}\left[f(Z)\right]-\mathbb{E}_{\overline{Z}^n}(f)\right|>\frac{\epsilon}{6K_\infty^{n-1}K_{\mathcal{L}}^{n-1}}\middle|\overline{Z}^{1:(n-1)}\right]\\
&\leq\mathbb{P}\left[\sup_{f\in\mathcal{L},\,\|f\|_{\mathcal{L}}\leq 1}\left|\mathbb{E}\left[f(Z)\right]-\mathbb{E}_{\overline{Z}^n}(f)\right|>\frac{\epsilon}{6K_\infty^{n-1}K_{\mathcal{L}}^{n-1}}\middle|\overline{Z}^{1:(n-1)}\right]\\
&\leq Ce^{-cM_n\phi\left(\frac{\epsilon}{6K_\infty^{n-1}K_{\mathcal{L}}^{n-1}}\right)}.
\end{aligned}$$

Hence the result. □

Lemma II.3.13. *Let $0<\gamma<1$ and $n\geq 2$. Then, it holds that almost surely*

$$\mathbb{P}\left[\mathcal{G}_n|\overline{Z}^{1:(n-1)}\right]\geq 1-\delta_n.$$

Proof. Since $(\overline{g}_1,\cdots,\overline{g}_n)$ forms a basis of $\overline{V}_{n-1}$, for all $\mu\in\mathcal{P}$, there exists one unique minimizer to

$$\min_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}}\mathrm{Var}\left[g_\mu(Z)-\sum_{i=1}^{n-1}\lambda_i\overline{g}_i(Z)\middle|\overline{Z}^{1:\infty}\right].$$

Let $\overline{\lambda}^n:=\left(\overline{\lambda}_i^n\right)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}$ be the unique minimizer of

$$\overline{\lambda}^n:=\operatorname*{argmin}_{\lambda:=(\lambda_i)_{1\leq i\leq n-1}}\mathrm{Var}\left[g_{\overline{\mu}_n}(Z)-\sum_{i=1}^{n-1}\lambda_i\overline{g}_i(Z)\middle|\overline{Z}^{1:\infty}\right]. \tag{II.24}$$

As a consequence, it holds that

$$\overline{\sigma}_{n-1}(\mathcal{M})=\mathrm{Var}\left[g_{\overline{\mu}_n}(Z)-\sum_{i=1}^{n-1}\overline{\lambda}_i^n\overline{g}_i(Z)\middle|\overline{Z}^{1:\infty}\right]$$

where $\overline{\sigma}_{n-1}(\mathcal{M})$ is defined by (II.12).

Let $\widehat{\mu}_n \in \mathcal{P}$ such that

$$\widehat{\mu}_n \in \operatorname*{argmax}_{\mu\in\mathcal{P}} \min_{(\lambda_i)_{1\le i\le n-1}\in\mathbb{R}^{n-1}} \mathrm{Var}\left[g_\mu(Z) - \sum_{i=1}^{n-1} \lambda_i \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right],$$

so that

$$\widehat{\sigma}_{n-1}(\mathcal{M}) = \min_{(\lambda_i)_{1\le i\le n-1}\in\mathbb{R}^{n-1}} \mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1} \lambda_i \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right],$$

where $\widehat{\sigma}_{n-1}(\mathcal{M})$ is defined in (II.11).

Let $\widehat{\lambda}^n := \left(\widehat{\lambda}_i^n\right)_{1\le i\le n-1} \in \mathbb{R}^{n-1}$ the unique minimizer of

$$\widehat{\lambda}^n := \operatorname*{argmin}_{\lambda:=(\lambda_i)_{1\le i\le n-1}} \mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1} \lambda_i \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right], \tag{II.25}$$

so that

$$\widehat{\sigma}_{n-1}(\mathcal{M}) = \mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1} \widehat{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right].$$

The event $\mathcal{G}_n$ holds if and only if

$$\mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1} \overline{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right] \ge \gamma^2 \mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1} \widehat{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right]. \tag{II.26}$$

Let us begin by pointing out that, since

$$\mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1} \widehat{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right] \ge \mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1} \overline{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right],$$

if the inequality

$$\begin{aligned} &\mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1} \widehat{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right] - \mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1} \overline{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right] \\ &\le (1-\gamma^2)\mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1} \widehat{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right] \end{aligned} \tag{II.27}$$

holds, then (II.26) is necessarily statisfied. The rest of the proof consists in estimating the probability that (II.27) is realized.

To this aim, as a first step, we are going to prove an upper bound on

$$\mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1} \widehat{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right] - \mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1} \overline{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right]$$

as a function of

$$\eta := \sup_{g,h\in\mathcal{M}_{n-1}} \left| \mathrm{Cov}\left[g(Z), h(Z) | \overline{Z}^{1:\infty} \right] - \mathrm{Cov}_{\overline{Z}^n}(g,h) \right|, \tag{II.28}$$

which is the quantity estimated in Lemma II.3.12. More precisely, let us now prove that

$$\begin{aligned} &\mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1} \widehat{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right] - \mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1} \overline{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right] \\ &\leq n \left(2 + \left(\frac{K_2 + \sqrt{n-1}\eta}{1-(n-1)\eta} \right)^2 + K_2^2 \right) \eta. \end{aligned} \tag{II.29}$$

It holds that for all $1 \leq i \leq n-1$, from (II.24) and (II.25),

$$\widehat{\lambda}_i^n = \mathrm{Cov}\left[g_{\widehat{\mu}_n}(Z), \overline{g}_i(Z) | \overline{Z}^{1:\infty} \right] \quad \text{and } \overline{\lambda}_i^n = \mathrm{Cov}\left[g_{\overline{\mu}_n}(Z), \overline{g}_i(Z) | \overline{Z}^{1:\infty} \right],$$

and it then holds that, almost surely,

$$\max\left(\|\widehat{\lambda}^n\|_{\ell^2}, \|\overline{\lambda}^n\|_{\ell^2} \right) \leq \max\left(\|g_{\widehat{\mu}_n}\|, \|g_{\overline{\mu}_n}\| \right) \leq K_2, \tag{II.30}$$

where $\|\cdot\|_{\ell^2}$ denotes the Euclidean norm of $\mathbb{R}^{n-1}$. Let now $\widehat{\lambda}^{n,n} := \left(\widehat{\lambda}_i^{n,n}\right)_{1\leq i\leq n-1} \in \mathbb{R}^{n-1}$ be a minimizer of

$$\widehat{\lambda}^{n,n} := \operatorname*{argmin}_{\lambda:=(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \mathrm{Var}_{\overline{Z}^n}\left(g_{\widehat{\mu}_n} - \sum_{i=1}^{n-1} \lambda_i \overline{g}_i \right),$$

and $\overline{\lambda}^{n,n} := \left(\overline{\lambda}_i^{n,n}\right)_{1\leq i\leq n-1} \in \mathbb{R}^{n-1}$ be a minimizer of

$$\overline{\lambda}^{n,n} := \operatorname*{argmin}_{\lambda:=(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \mathrm{Var}_{\overline{Z}^n}\left(g_{\overline{\mu}_n} - \sum_{i=1}^{n-1} \lambda_i \overline{g}_i \right)$$

It then holds that for all $1 \leq i \leq n-1$, $\widehat{\lambda}^{n,n}$ and $\overline{\lambda}^{n,n}$ are solution to the linear systems

$$A^n \widehat{\lambda}^{n,n} = \widehat{b}^n \quad \text{and } A^n \overline{\lambda}^{n,n} = \overline{b}^n,$$

where $A^n := \left(A_{ij}^n\right)_{1\leq i,j\leq n-1} \in \mathbb{R}^{(n-1)\times(n-1)}$, $\widehat{b}^n := \left(\widehat{b}_i^n\right)_{1\leq i\leq n-1}, \overline{b}^n := \left(\overline{b}_i^n\right)_{1\leq i\leq n-1} \in \mathbb{R}^{n-1}$ are defined as follows: for all $1 \leq i,j \leq n-1$,

$$A_{ij}^n = \mathrm{Cov}_{\overline{Z}^n}\left(\overline{g}_i, \overline{g}_j\right), \ \widehat{b}_i^n = \mathrm{Cov}_{\overline{Z}^n}\left(g_{\widehat{\mu}_n}, \overline{g}_i\right) \ \text{ and } \overline{b}_i^n = \mathrm{Cov}_{\overline{Z}^n}\left(g_{\overline{\mu}_n}, \overline{g}_i\right).$$

Then, it holds that, almost surely,

$$\max_{1\leq i\leq n-1}\left(\left|\widehat{b}_i^n - \widehat{\lambda}_i^n\right|, \left|\overline{b}_i^n - \overline{\lambda}_i^n\right| \right) \leq \eta,$$

which implies that

$$\max\left(\|\widehat{b}_n\|_{\ell^2}, \|\overline{b}_n\|_{\ell^2} \right) \leq K_2 + \sqrt{n-1}\eta.$$

Moreover, we have

$$\max_{1\le i,j\le n-1} \left|A^n_{ij} - \delta_{ij}\right| \le \eta,$$

which yields that for all $\xi \in \mathbb{R}^{n-1}$,

$$(1-(n-1)\eta)\|\xi\|^2_{\ell^2} \le \xi^T A^n \xi \le (1+(n-1)\eta)\|\xi\|^2_{\ell^2}.$$

Assume for now that $\eta(n-1) < 1$, this implies that, for all $\xi \in \mathbb{R}^{n-1}$,

$$\|(A^n)^{-1}\xi\|_{\ell^2} \le \frac{1}{1-(n-1)\eta}\|\xi\|_{\ell^2}. \tag{II.31}$$

Using (II.31), we obtain that

$$\max\left(\|\overline{\lambda}^{n,n}\|_{\ell^2}, \|\widehat{\lambda}^{n,n}\|_{\ell^2}\right) \le \frac{K_2+\sqrt{n-1}\eta}{1-(n-1)\eta}. \tag{II.32}$$

We then have,

$$\begin{aligned}
&\mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1}\widehat{\lambda}^n_i \overline{g}_i(Z)\middle|\overline{Z}^{1:\infty}\right] - \mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1}\overline{\lambda}^n_i \overline{g}_i(Z)\middle|\overline{Z}^{1:\infty}\right] \\
&= \mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1}\widehat{\lambda}^n_i \overline{g}_i(Z)\middle|\overline{Z}^{1:\infty}\right] - \mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1}\widehat{\lambda}^{n,n}_i \overline{g}_i(Z)\middle|\overline{Z}^{1:\infty}\right] \\
&\quad + \mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1}\widehat{\lambda}^{n,n}_i \overline{g}_i(Z)\middle|\overline{Z}^{1:\infty}\right] - \mathrm{Var}_{\overline{Z}^n}\left(g_{\widehat{\mu}_n} - \sum_{i=1}^{n-1}\widehat{\lambda}^{n,n}_i \overline{g}_i\right) \\
&\quad + \mathrm{Var}_{\overline{Z}^n}\left(g_{\widehat{\mu}_n} - \sum_{i=1}^{n-1}\widehat{\lambda}^{n,n}_i \overline{g}_i\right) - \mathrm{Var}_{\overline{Z}^n}\left(g_{\overline{\mu}_n} - \sum_{i=1}^{n-1}\overline{\lambda}^{n,n}_i \overline{g}_i\right) \\
&\quad + \mathrm{Var}_{\overline{Z}^n}\left(g_{\overline{\mu}_n} - \sum_{i=1}^{n-1}\overline{\lambda}^{n,n}_i \overline{g}_i\right) - \mathrm{Var}_{\overline{Z}^n}\left(g_{\overline{\mu}_n} - \sum_{i=1}^{n-1}\overline{\lambda}^{n}_i \overline{g}_i\right) \\
&\quad + \mathrm{Var}_{\overline{Z}^n}\left(g_{\overline{\mu}_n} - \sum_{i=1}^{n-1}\overline{\lambda}^{n}_i \overline{g}_i\right) - \mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1}\overline{\lambda}^n_i \overline{g}_i(Z)\middle|\overline{Z}^{1:\infty}\right].
\end{aligned}$$

Using the fact that

$$\begin{aligned}
&\mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1}\widehat{\lambda}^n_i \overline{g}_i(Z)\middle|\overline{Z}^{1:\infty}\right] - \mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1}\widehat{\lambda}^{n,n}_i \overline{g}_i(Z)\middle|\overline{Z}^{1:\infty}\right] \le 0, \\
&\mathrm{Var}_{\overline{Z}^n}\left(g_{\widehat{\mu}_n} - \sum_{i=1}^{n-1}\widehat{\lambda}^{n,n}_i \overline{g}_i\right) - \mathrm{Var}_{\overline{Z}^n}\left(g_{\overline{\mu}_n} - \sum_{i=1}^{n-1}\overline{\lambda}^{n,n}_i \overline{g}_i\right) \le 0, \\
&\mathrm{Var}_{\overline{Z}^n}\left(g_{\overline{\mu}_n} - \sum_{i=1}^{n-1}\overline{\lambda}^{n,n}_i \overline{g}_i\right) - \mathrm{Var}_{\overline{Z}^n}\left(g_{\overline{\mu}_n} - \sum_{i=1}^{n-1}\overline{\lambda}^{n}_i \overline{g}_i\right) \le 0,
\end{aligned}$$

from the definition of $\widehat{\lambda}^n$, $\widehat{\lambda}^{n,n}$, $\overline{\lambda}^{n,n}$, $\overline{\mu}_n$, we obtain that

$$
\begin{aligned}
&\mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1} \widehat{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty}\right] - \mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1} \overline{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty}\right] \\
&\leq \mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1} \widehat{\lambda}_i^{n,n} \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty}\right] - \mathrm{Var}_{\overline{Z}^n}\left(g_{\widehat{\mu}_n} - \sum_{i=1}^{n-1} \widehat{\lambda}_i^{n,n} \overline{g}_i\right) \\
&\quad + \mathrm{Var}_{\overline{Z}^n}\left(g_{\overline{\mu}_n} - \sum_{i=1}^{n-1} \overline{\lambda}_i^n \overline{g}_i\right) - \mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1} \overline{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty}\right], \\
&= \mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) \middle| \overline{Z}^{1:\infty}\right] - \mathrm{Var}_{\overline{Z}^n}\left(g_{\widehat{\mu}_n}\right) - 2\sum_{i=1}^{n-1} \widehat{\lambda}_i^{n,n} \left(\mathrm{Cov}\left[g_{\widehat{\mu}_n}(Z), \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty}\right] - \mathrm{Cov}_{\overline{Z}^n}\left(g_{\widehat{\mu}_n}, \overline{g}_i\right)\right) \\
&\quad + \sum_{i,j=1}^{n-1} \widehat{\lambda}_i^{n,n} \widehat{\lambda}_j^{n,n} \left(\mathrm{Cov}\left[\overline{g}_i(Z), \overline{g}_j(Z) \middle| \overline{Z}^{1:\infty}\right] - \mathrm{Cov}_{\overline{Z}^n}\left(\overline{g}_i, \overline{g}_j\right)\right) \\
&\quad + \mathrm{Var}_{\overline{Z}^n}\left(g_{\overline{\mu}_n}\right) - \mathrm{Var}\left[g_{\overline{\mu}_n}(Z) \middle| \overline{Z}^{1:\infty}\right] - 2\sum_{i=1}^{n-1} \overline{\lambda}_i^n \left(\mathrm{Cov}_{\overline{Z}^n}\left(g_{\overline{\mu}_n}, \overline{g}_i\right) - \mathrm{Cov}\left[\overline{g}_i(Z), \overline{g}_j(Z) \middle| \overline{Z}^{1:\infty}\right]\right) \\
&\quad + \sum_{i,j=1}^{n-1} \overline{\lambda}_i^n \overline{\lambda}_j^n \left(\mathrm{Cov}_{\overline{Z}^n}\left(\overline{g}_i, \overline{g}_j\right) - \mathrm{Cov}\left[g_{\overline{\mu}_n}(Z), \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty}\right]\right).
\end{aligned}
$$

Now, using the definition of $\mathcal{M}^{n-1}$ given in (II.23), we obtain that

$$
\begin{aligned}
&\mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1} \widehat{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty}\right] - \mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1} \overline{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty}\right] \\
&\leq \left(1 + 2\sum_{i=1}^{n-1} |\widehat{\lambda}_i^{n,n}| + \sum_{i,j=1}^{n-1} |\widehat{\lambda}_i^{n,n}||\widehat{\lambda}_j^{n,n}| + 1 + 2\sum_{i=1}^{n-1} |\overline{\lambda}_i^n| + \sum_{i,j=1}^{n-1} |\overline{\lambda}_i^n||\overline{\lambda}_j^n|\right) \\
&\quad \times \sup_{g,h \in \mathcal{M}_{n-1}} \left|\mathrm{Cov}\left[g(Z), h(Z) \middle| \overline{Z}^{1:\infty}\right] - \mathrm{Cov}_{\overline{Z}^n}\left(g, h\right)\right|.
\end{aligned}
$$

Since $\sup_{g,h \in \mathcal{M}_{n-1}} \left|\mathrm{Cov}\left[g(Z), h(Z) \middle| \overline{Z}^{1:\infty}\right] - \mathrm{Cov}_{\overline{Z}^n}\left(g, h\right)\right| = \eta$, we then have, almost surely,

$$
\begin{aligned}
&\mathrm{Var}\left[g_{\widehat{\mu}_n}(Z) - \sum_{i=1}^{n-1} \widehat{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty}\right] - \mathrm{Var}\left[g_{\overline{\mu}_n}(Z) - \sum_{i=1}^{n-1} \overline{\lambda}_i^n \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty}\right] \\
&\leq \left[\left(1 + \sum_{i=1}^{n-1} |\widehat{\lambda}_i^{n,n}|\right)^2 + \left(1 + \sum_{i=1}^{n-1} |\overline{\lambda}_i^n|\right)^2\right] \sup_{g,h \in \mathcal{M}_{n-1}} \left|\mathrm{Cov}\left[g(Z), h(Z) \middle| \overline{Z}^{1:\infty}\right] - \mathrm{Cov}_{\overline{Z}^n}\left(g, h\right)\right| \\
&\leq n\left(2 + \sum_{i=1}^{n-1} |\widehat{\lambda}_i^{n,n}|^2 + \sum_{i=1}^{n-1} |\overline{\lambda}_i^n|^2\right)\eta \\
&\leq n\left(2 + \|\widehat{\lambda}^{n,n}\|_{\ell^2}^2 + \|\overline{\lambda}^n\|_{\ell^2}^2\right)\eta.
\end{aligned}
$$

Finally, using (II.30) and (II.32), we obtain (II.29), i.e.

$$\widehat{\sigma}_{n-1}(\mathcal{M}) - \overline{\sigma}_{n-1}(\mathcal{M}) \leq n\left(2 + \left(\frac{K_2 + \sqrt{n-1}\eta}{1-(n-1)\eta}\right)^2 + K_2^2\right)\eta.$$

Let us now evaluate the probability, conditioned to the knowledge of $\overline{Z}^{1:\infty}$, that

$$n\left(2 + \left(\frac{K_2 + \sqrt{n-1}\eta}{1-(n-1)\eta}\right)^2 + K_2^2\right)\eta \leq (1-\gamma^2)\widehat{\sigma}_{n-1}(\mathcal{M}). \tag{II.33}$$

If η is chosen to be smaller that $\frac{1}{2(n-1)}$, then it holds that

$$2 + \left(\frac{K_2 + \sqrt{n-1}\eta}{1-(n-1)\eta}\right)^2 + K_2^2 \leq 2 + (2K_2+1)^2 + K_2^2 \leq 9K_2^2 + 4.$$

A sufficient condition for (II.33) to hold is then to ensure that $\eta \leq \epsilon$ with

$$\epsilon := \min\left(\frac{1}{2(n-1)}, \quad \frac{(1-\gamma^2)\widehat{\sigma}_{n-1}^2(\mathcal{M})}{n\,(9K_2^2+4)}\right),$$

Then, it holds that

$$\begin{aligned}
\mathbb{P}\left[\mathcal{G}_n \middle| \overline{Z}^{1:\infty}\right] &= \mathbb{P}\left[\overline{\sigma}_{n-1}(\mathcal{M})^2 \geq \gamma^2 \widehat{\sigma}_{n-1}(\mathcal{M})^2 \middle| \overline{Z}^{1:\infty}\right] \\
&= \mathbb{P}\left[\widehat{\sigma}_{n-1}(\mathcal{M})^2 - \overline{\sigma}_{n-1}(\mathcal{M})^2 \leq (1-\gamma^2)\widehat{\sigma}_{n-1}(\mathcal{M})^2 \middle| \overline{Z}^{1:\infty}\right] \\
&\geq \mathbb{P}\left[n\left(2 + \left(\frac{K_2 + \sqrt{n-1}\eta}{1-(n-1)\eta}\right)^2 + K_2^2\right)\eta \leq (1-\gamma^2)\widehat{\sigma}_{n-1}(\mathcal{M}) \middle| \overline{Z}^{1:\infty}\right] \\
&\geq \mathbb{P}\left[\eta \leq \epsilon \middle| \overline{Z}^{1:\infty}\right].
\end{aligned}$$

Thus, using the definition of η given by (II.28) and applying Lemma II.3.12, we then obtain that

$$\begin{aligned}
\mathbb{P}\left[\mathcal{G}_n | \overline{Z}^{1:(n-1)}\right] &\geq \mathbb{P}\left[\eta \leq \epsilon \middle| \overline{Z}^{1:\infty}\right] \\
&= \mathbb{P}\left[\sup_{g,h \in \mathcal{M}^{n-1}} \left|\mathrm{Cov}\left[g(Z), h(Z) \middle| \overline{Z}^{1:\infty}\right] - \mathrm{Cov}_{\overline{Z}^n}(g,h)\right| \leq \epsilon \middle| \overline{Z}^{1:(n-1)}\right] \\
&\geq 1 - \delta_n,
\end{aligned}$$

since

$$Ce^{-cM_n\phi(\kappa_{n-1})} \leq \delta_n,$$

with

$$\kappa_{n-1} := \frac{\min\left(\frac{1}{2(n-1)}, \quad \frac{(1-\gamma^2)\widehat{\sigma}_{n-1}^2(\mathcal{M})}{n\left(9K_2^2+4\right)}\right)}{6K_\infty^{n-1}K_{\mathcal{L}}^{n-1}},$$

which yields the desired result. □

We are now in position to end the proof of Theorem II.3.6.

Proof of Theorem II.3.6. Collecting Lemma II.3.11 and Lemma II.3.13, we obtain (II.18) for all $n \in \mathbb{N}^*$. Let us now prove (II.19).

Let us first prove by recursion that for all $n \in \mathbb{N}^*$,

$$\mathbb{P}\left[\bigcap_{k=1}^{n} \mathcal{G}_k\right] \geq \Pi_{k=1}^{n}(1-\delta_k). \tag{II.34}$$

Using Lemma II.3.11, it holds that (II.34) is true for $n = 1$. Now we turn to the proof of the recursion. Let $n \in \mathbb{N}^*$. For any event $\mathcal{Z}$, we denote by $\mathbb{1}_{\mathcal{Z}}$ the random variable which is equal to 1 if $\mathcal{Z}$ is realized and 0 if not. Using the fact that $\bigcap_{k=1}^{n} \mathcal{G}_k$ is measurable with respect to $\overline{Z}^{1:n}$, it holds that

$$\begin{aligned}
\mathbb{P}\left[\bigcap_{k=1}^{n+1} \mathcal{G}_k\right] &= \mathbb{E}\left[\mathbb{1}_{\bigcap_{k=1}^{n+1} \mathcal{G}_k}\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[\mathbb{1}_{\mathcal{G}_{n+1}} \mathbb{1}_{\bigcap_{k=1}^{n} \mathcal{G}_k} \middle| \overline{Z}^{1:n}\right]\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[\mathbb{1}_{\mathcal{G}_{n+1}} \middle| \overline{Z}^{1:n}\right] \mathbb{1}_{\bigcap_{k=1}^{n} \mathcal{G}_k}\right] \\
&= \mathbb{E}\left[\mathbb{P}\left[\mathcal{G}_{n+1} \middle| \overline{Z}^{1:n}\right] \mathbb{1}_{\bigcap_{k=1}^{n} \mathcal{G}_k}\right].
\end{aligned}$$

Now using Lemma II.3.13, it holds that almost surely $\mathbb{P}\left[\mathcal{G}_{n+1} \middle| \overline{Z}^{1:n}\right] \geq 1 - \delta_{n+1}$. Hence, it holds that

$$\mathbb{P}\left[\bigcap_{k=1}^{n+1} \mathcal{G}_k\right] \geq (1-\delta_{n+1})\mathbb{E}\left[\mathbb{1}_{\bigcap_{k=1}^{n} \mathcal{G}_k}\right] = (1-\delta_{n+1})\mathbb{P}\left[\bigcap_{k=1}^{n} \mathcal{G}_k\right].$$

The recursion is thus proved, together with (II.34), which implies (II.19).

If $\bigcap_{n\in\mathbb{N}} \mathcal{G}_n$ is realised, it then holds that the MC-greedy algorithm is a weak greedy algorithm with parameter γ and norm $\|\cdot\|_{\overline{Z}^{1:\infty}} = \sqrt{\mathrm{Var}\left[\cdot|\overline{Z}^{1:\infty}\right]}$.

□

II.4 Numerical results

The aim of this section is to compare several procedures to choose the value of the sequence $(M_n)_{n\in\mathbb{N}^*}$ in the MC-greedy algorithm presented in Section II.3.1.

II.4.1 Three numerical procedures

As mentioned in Remark II.3.9, it is possible to design a constructive way to define a sequence of numbers of samples $(M_n)_{n\in\mathbb{N}^*}$ which satisfies assumptions of Theorem II.3.6, and thus which guarantees that the corresponding MC-greedy algorithm is a weak-greedy algorithm with high

probability. Unfortunately, it appears that such a procedure cannot be used in practice to design a variance reduction method since the values of the sequence $(M_n)_{n\in\mathbb{N}^*}$ increases too sharply. The objective of this section is to propose a *heuristic* procedure to choose a sequence $(M_n)_{n\in\mathbb{N}^*}$ for an MC-greedy algorithm. This heuristic procedure appears to yield a reduced basis approximation $\overline{f}_\mu$ of f_μ which provides very satisfactory results in terms of variance reduction, at least on the different test cases presented below.

We use here the same notation as in Section II.2.1 and consider $M_{\text{ref}} \in \mathbb{N}^*$ such that $M_{\text{ref}} \gg 1$. The idea of this heuristic method is the following: assume that the sequence $(M_n)_{n\in\mathbb{N}^*}$ can be chosen so that for all $n \in \mathbb{N}^*$, the inequality

$$\left| \inf_{(\lambda_i)_{1\le i\le n-1}\in\mathbb{R}^{n-1}} \text{Var}\left[g_\mu(Z) - \sum_{i=1}^{n-1} \lambda_i \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right] - \inf_{(\lambda_i)_{1\le i\le n-1}\in\mathbb{R}^{n-1}} \text{Var}_{\overline{Z}^n}\left(g_\mu - \sum_{i=1}^{n-1} \lambda_i \overline{g}_i \right) \right|$$
$$\le (1-\gamma^2) \inf_{(\lambda_i)_{1\le i\le n-1}\in\mathbb{R}^{n-1}} \text{Var}\left[g_\mu(Z) - \sum_{i=1}^{n-1} \lambda_i \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right] \tag{II.35}$$

holds for all $\mu \in \mathcal{P}$. Then, it can easily be checked that such an MC-greedy algorithm is a weak greedy algorithm with parameter γ. Of course, such an algorithm could not be of any use for variance reduction since it would imply the computation of $\inf_{(\lambda_i)_{1\le i\le n-1}\in\mathbb{R}^{n-1}} \text{Var}\left[g_\mu(Z) - \sum_{i=1}^{n-1} \lambda_i \overline{g}_i(Z) \middle| \overline{Z}^{1:\infty} \right]$ (or an approximation of this quantity of the form $\inf_{(\lambda_i)_{1\le i\le n-1}\in\mathbb{R}^{n-1}} \text{Var}_{\overline{Z}^{\text{ref}}}\left(g_\mu - \sum_{i=1}^{n-1} \lambda_i \overline{g}_i \right)$ with $\overline{Z}^{\text{ref}} = \left(Z_k^{\text{ref}}\right)_{1\le k\le M^{\text{ref}}}$ a collection of iid random variables with the same law as Z and independent of Z) for all $\mu \in \mathcal{P}$.

The idea of the heuristic procedure is then to check if the inequality (II.35) holds, *only for the value* $\mu = \overline{\mu}_n$. If the inequality holds, the value of M_{n+1} is chosen to be equal to M_n for the next iteration. Otherwise, the value of M_n is increased and the n^{th} iteration of the MC-greedy algorithm is performed again.

This heuristic procedure leads to the Heuristic MC-greedy algorithm (or HMC-greedy algorithm), see Algorithm 4. Notice that we introduce here $\mathcal{P}_{\text{trial}}$ a finite subset of $\mathcal{P}$, which is classically called the trial set of parameters in reduced basis methods.

For the sake of comparison, we introduce two other algorithms, which cannot be implemented in practice, but which will allow us to compare the performance of the HMC-greedy algorithm with ideal procedures. The first method, called SHMC-greedy algorithm and also presented in Algorithm 4 as a variant, consists in designing the sequence $(M_n)_{n\in\mathbb{N}^*}$ in order to ensure that the inequality (II.35) is satisfied for all $\mu \in \mathcal{P}_{\text{trial}}$ (and not only for $\overline{\mu}_n$). The second one consists in performing an *ideal* MC-greedy algorithm, called herefater IMC-greedy algorithm, see Algorithm 5, where all the variances and expectations are evaluated using M_{ref} number of samples of the random variable Z at each iteration of the MC-greedy algorithm.

Let us comment on the termination criterion

$$\frac{\text{Var}_{\overline{Z}^{\text{ref}}}\left(\overline{f}_{\mu_{n-1}^{(S)H}}\right)}{M_{\text{ref}}} > \frac{\text{Var}_{\overline{Z}^{\text{ref}}}\left(f_{\mu_{n-1}^{(S)H}} - \overline{f}_{\mu_{n-1}^{(S)H}}\right)}{M_{n-1}}$$

introduced in line 11 of the (S)HMC-greedy algorithm. Recall that, for $\mu = \mu_{n-1}^{(S)H}$, the expectation $\mathbb{E}\left[f_{\mu_{n-1}^{(S)H}}(Z)\right]$ is approximated after $n-1$ iterations of the greedy algorithm by the control variate formula (see (II.1))

$$\mathbb{E}_{\overline{Z}^{\mathrm{ref}}}(\overline{f}_{\mu_{n-1}^{(S)H}}) + \mathbb{E}_{\overline{Z}^{n-1}}\left(f_{\mu_{n-1}^{(S)H}} - \overline{f}_{\mu_{n-1}^{(S)H}}\right). \tag{II.36}$$

This criterion ensures that the iterative scheme ends when the statistical error associated with the second term in (II.36) becomes smaller than the statistical error of the first term (see Remark II.2.1).

II.4.2 Definitions of quantities of interest

For each of the test cases presented below, we plot different quantities of interest which we define here.

For all $n \in \mathbb{N}^*$, we denote by $\mu_1^H, \cdots, \mu_n^H$ (respectively $\mu_1^{SH}, \cdots, \mu_n^{SH}$ and $\mu_1^I, \cdots, \mu_n^I$) the set of parameter values selected after n iterations of the HMC-greedy (respectively SHMC-greedy and IMC-greedy) algorithm. We also denote by $\overline{V}_n^H := \mathrm{Span}\left\{g_{\mu_1^H}, \cdots, g_{\mu_n^H}\right\}$, $\overline{V}_n^{SH} := \mathrm{Span}\left\{g_{\mu_1^{SH}}, \cdots, g_{\mu_n^{SH}}\right\}$ and $\overline{V}_n^I := \mathrm{Span}\left\{g_{\mu_1^I}, \cdots, g_{\mu_n^I}\right\}$.

For all $\mu \in \mathcal{P}$ and $n \in \mathbb{N}^*$, we define for the three algorithms presented in Section II.4.1,

$$\theta_n(\mu) := \inf_{(\lambda_i)_{1\le i\le n-1}\in\mathbb{R}^{n-1}} \sqrt{\mathrm{Var}_{\overline{Z}^{\mathrm{ref}}}\left(g_\mu - \sum_{i=1}^{n-1}\lambda_i \overline{g}_i\right)} \tag{II.37}$$

and

$$\theta_n := \sup_{\mu\in\mathcal{P}_{\mathrm{trial}}} \theta_n(\mu). \tag{II.38}$$

In what follows, we denote by $\theta_n^H(\mu)$ and θ_n^H (respectively by $\theta_n^{SH}(\mu)$, θ_n^{SH}, $\theta_n^I(\mu)$ and θ_n^I) the quantities defined by (II.37) and (II.38) obtained with the HMC-greedy (respectively the SHMC-greedy and IMC-greedy) algorithm. Note that, by definition of the IMC-greedy algorithm, $\theta_n^I = \theta_n^I(\mu_n^I)$.

A second quantity of interest for the HMC-greedy and the SHMC-greedy algorithms is given, for all $n \in \mathbb{N}^*$ and $\mu \in \mathcal{P}$, by

$$\beta_n(\mu) := \inf_{(\lambda_i)_{1\le i\le n-1}\in\mathbb{R}^{n-1}} \sqrt{\mathrm{Var}_{\overline{Z}^{n}}\left(g_\mu - \sum_{i=1}^{n-1}\lambda_i \overline{g}_i\right)}. \tag{II.39}$$

and

$$\beta_n := \sup_{\mu\in\mathcal{P}_{\mathrm{trial}}} \beta_n(\mu). \tag{II.40}$$

In the sequel, we denote by β_n^H (respectively by β_n^{SH}) the quantity defined by (II.40) obtained with the HMC-greedy (respectively the SHMC-greedy) algorithm.

Let us point out that when $\mathcal{P}_{\mathrm{trial}} = \mathcal{P}$ and when $M_{\mathrm{ref}} = \infty$, it holds that $\theta_n = \widehat{\sigma}_{n-1}(\mathcal{M})$ and $\theta_n(\overline{\mu}_n) = \overline{\sigma}_{n-1}(\mathcal{M})$ where $\widehat{\sigma}_{n-1}(\mathcal{M})$ and $\overline{\sigma}_{n-1}(\mathcal{M})$ are defined respectively in (II.11) and (II.12).

Algorithm 4 (S)HMC-greedy algorithm

input : $\gamma > 0$, $\epsilon > 0$, $M_1 \in \mathbb{N}^*$, $\mathcal{P}_{\text{trial}}$ trial set of parameters (finite subset of $\mathcal{P}$), $M_{\text{ref}} \in \mathbb{N}^*$ (high fidelity sampling number, which has a vocation to satisfy $M_{\text{ref}} \gg M_1$.

output: $N \in \mathbb{N}^*$ size of the reduced basis, $\mu_1^{(S)H}, \mu_2^{(S)H}, \cdots, \mu_N^{(S)H} \in \mathcal{P}_{\text{trial}}$, $\left(\mathbb{E}_{\overline{Z}^{\text{ref}}}(f_{\mu_n^{(S)H}})\right)_{1\leq n\leq N}$.

Let $\overline{Z}^{\text{ref}} := \left(Z_k^{\text{ref}}\right)_{1\leq k\leq M_{\text{ref}}}$ be a collection of M_{ref} iid random variables with the same law as Z and independent of Z.

Set $R^1 = 1$.

while $R^1 \geq 1-\gamma^2$ **do**

  Let $\overline{Z}^1 := \left(Z_k^1\right)_{1\leq k\leq M_1}$ be a collection of M_1 iid random variables with the same law as Z and independent of Z and $\overline{Z}^{\text{ref}}$.

  Compute $\mu_1^{(S)H} \in \underset{\mu\in\mathcal{P}_{\text{trial}}}{\operatorname{argmax}} \operatorname{Var}_{\overline{Z}^1}(f_\mu)$.

  Compute $\overline{f}_{\mu_1^{(S)H}} = 0$.

  Compute $\mathbb{E}_{\overline{Z}^{\text{ref}}}(f_{\mu_1^{(S)H}})$.

- **HMC case:** Set $\theta_1^H(\mu_1^H) = \sqrt{\operatorname{Var}_{\overline{Z}^{\text{ref}}}\left(f_{\mu_1^H}\right)}$ and $\beta_1^H(\mu_1^H) = \sqrt{\operatorname{Var}_{\overline{Z}^1}\left(f_{\mu_1^H}\right)}$. Set $R^1 = \frac{|\theta_1^H(\mu_1^H)^2 - \beta_1^H(\mu_1^H)^2|}{\theta_1^H(\mu_1^H)^2}$.

- **SHMC case:** Set $\theta_1^{SH}(\mu) = \sqrt{\operatorname{Var}_{\overline{Z}^{\text{ref}}}(f_\mu)}$ and $\beta_1^{SH}(\mu) = \sqrt{\operatorname{Var}_{\overline{Z}^1}(f_\mu)}$ for all $\mu \in \mathcal{P}_{\text{trial}}$. Set $R^1 = \sup_{\mu\in\mathcal{P}_{\text{trial}}} \frac{|\theta_1^{SH}(\mu)^2 - \beta_1^{SH}(\mu)^2|}{\theta_1^{SH}(\mu)^2}$.

  if $R^1 \geq 1-\gamma^2$ **then**

    Set $b_1 := 1.1$ and $M_1 = \lceil b_1 M_1 + 1\rceil$.

  end

end

Compute $\overline{g}^1 = \frac{f_{\mu_1^{(S)H}} - \mathbb{E}_{\overline{Z}^{\text{ref}}}\left(f_{\mu_1^{(S)H}}\right)}{\theta_1^{(S)H}(\mu_1^{(S)H})}$. Set $n = 2$ and $M_n = M_1$.

while $\frac{\operatorname{Var}_{\overline{Z}^{\text{ref}}}\left(\overline{f}_{\mu_{n-1}^{(S)H}}\right)}{M_{\text{ref}}} \leq \frac{\operatorname{Var}_{\overline{Z}^{\text{ref}}}\left(f_{\mu_{n-1}^{(S)H}} - \overline{f}_{\mu_{n-1}^{(S)H}}\right)}{M_{n-1}}$ **do**

  Set $R^n = 1$.

  while $R^n \geq 1-\gamma^2$ **do**

    Let $\overline{Z}^n := \left(Z_k^n\right)_{1\leq k\leq M_n}$ be a collection of M_n iid random variables with the same law as Z and independent of Z and $\overline{Z}^{\text{ref}}$.

    Compute $\mu_n^{(S)H} \in \underset{\mu\in\mathcal{P}_{\text{trial}}}{\operatorname{argmax}} \min_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \operatorname{Var}_{\overline{Z}^n}\left(f_\mu - \sum_{i=1}^{n-1}\lambda_i\overline{g}_i\right)$

    Compute $(\overline{\lambda}_i^n)_{1\leq i\leq n-1} = \operatorname{argmin}_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \sqrt{\operatorname{Var}_{\overline{Z}^{\text{ref}}}\left(f_{\mu_n^{(S)H}} - \sum_{i=1}^{n-1}\lambda_i\overline{g}_i\right)}$.

    Compute $\overline{f}_{\mu_n^{(S)H}} = \sum_{i=1}^{n-1}\overline{\lambda}_i^n\overline{g}_i$.

- **HMC case:** Compute $\theta_n^H(\mu_n^H) = \min_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \sqrt{\operatorname{Var}_{\overline{Z}^{\text{ref}}}\left(f_{\mu_n^{(S)H}} - \sum_{i=1}^{n-1}\lambda_i\overline{g}_i\right)} = \sqrt{\operatorname{Var}_{\overline{Z}^{\text{ref}}}\left(f_{\mu_n^{(S)H}} - \overline{f}_{\mu_n^{(S)H}}\right)}$ and $\beta_n^H(\mu_n^H) = \min_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \sqrt{\operatorname{Var}_{\overline{Z}^n}\left(f_{\mu_n^H} - \sum_{i=1}^{n-1}\lambda_i\overline{g}_i\right)}$. Set $R^n = \left|\frac{\theta_n^H(\mu_n^H)^2 - \beta_n^H(\mu_n^H)^2}{\theta_n^H(\mu_n^H)^2}\right|$.

- **SHMC case:** For all $\mu \in \mathcal{P}_{\text{trial}}$, compute $\theta_n^{SH}(\mu) = \min_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \sqrt{\operatorname{Var}_{\overline{Z}^{\text{ref}}}\left(f_\mu - \sum_{i=1}^{n-1}\lambda_i\overline{g}_i\right)}$ and $\beta_n^{SH}(\mu) = \min_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \sqrt{\operatorname{Var}_{\overline{Z}^n}\left(f_\mu - \sum_{i=1}^{n-1}\lambda_i\overline{g}_i\right)}$. Set $R^n = \sup_{\mu\in\mathcal{P}_{\text{trial}}} \left|\frac{\theta_n^{SH}(\mu)^2 - \beta_n^{SH}(\mu)^2}{\theta_n^{SH}(\mu)^2}\right|$.

    if $R^n \geq 1-\gamma^2$ **then**

      Compute $r_n := \frac{\phi(\theta_{n-1}^{(S)H}(\mu_{n-1}^{(S)H})^2)}{\phi\left(\theta_n^{(S)H}(\mu_n^{(S)H})^2\right)}$, $b_n := \max(1.1, r_n)$ and set $M_n = \lceil b_n M_n + 1\rceil$.

    end

  end

  Compute $\overline{g}_n = \frac{f_{\mu_n^{(S)H}} - \sum_{i=1}^{n-1}\overline{\lambda}_i^n\overline{g}_i}{\theta_n^{(S)H}(\mu_n^{(S)H})} = \frac{f_{\mu_n^{(S)H}} - \overline{f}_{\mu_n^{(S)H}}}{\theta_n^{(S)H}(\mu_n^{(S)H})}$ and $\mathbb{E}_{\overline{Z}^{\text{ref}}}(f_{\mu_n^{(S)H}})$.

  Set $M_{n+1} = M_n$ and $n = n+1$.

end

Set $N = n$, $M_N = M_n$.

Algorithm 5 IMC-greedy algorithm

input : $\epsilon > 0$, $\mathcal{P}_{\text{trial}}$ trial set of parameters (finite subset of $\mathcal{P}$), $M_{\text{ref}} \in \mathbb{N}^*$ (high fidelity sampling number).

output: $N \in \mathbb{N}^*$ size of the reduced basis, $\mu_1^I, \mu_2^I, \cdots, \mu_N^I \in \mathcal{P}_{\text{trial}}$, $\left(\mathbb{E}_{\overline{Z}^{\text{ref}}}(f_{\mu_n^I})\right)_{1\leq n\leq N}$.

Let $\overline{Z}^{\text{ref}} := \left(Z_k^{\text{ref}}\right)_{1\leq k\leq M_{\text{ref}}}$ be a collection of M_{ref} iid random variables with the same law as Z and independent of Z.

Compute $\mu_1^I \in \underset{\mu\in\mathcal{P}_{\text{trial}}}{\operatorname{argmax}} \operatorname{Var}_{\overline{Z}^{\text{ref}}}(f_\mu)$.

Set $\theta_1^I(\mu_1^I) := \operatorname{Var}_{\overline{Z}^{\text{ref}}}\left(f_{\mu_1^I}\right)$ and compute $\mathbb{E}_{\overline{Z}^{\text{ref}}}\left(f_{\mu_1^I}\right)$.

Set $n = 2$, $M_n = M_1$, $\overline{g}^1 = \dfrac{f_{\mu_1^I}}{\sqrt{\operatorname{Var}_{\overline{Z}^{\text{ref}}}\left(f_{\mu_1^I}\right)}}$

while $\theta_{n-1}^I(\mu_{n-1}^I) \geq \epsilon$ **do**

Compute $\mu_n^I \in \underset{\mu\in\mathcal{P}_{\text{trial}}}{\operatorname{argmax}} \min_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \operatorname{Var}_{\overline{Z}^{\text{ref}}}\left(f_\mu - \sum_{i=1}^{n-1}\lambda_i\overline{g}_i\right)$

Compute $\theta_n^I(\mu_n^I) = \min_{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}} \sqrt{\operatorname{Var}_{\overline{Z}^{\text{ref}}}\left(f_{\mu_n^I} - \sum_{i=1}^{n-1}\lambda_i\overline{g}_i\right)}$

Compute $(\overline{\lambda}_i^n)_{1\leq i\leq n-1} = \underset{(\lambda_i)_{1\leq i\leq n-1}\in\mathbb{R}^{n-1}}{\operatorname{argmin}} \operatorname{Var}_{\overline{Z}^{\text{ref}}}\left(f_{\mu_n^I} - \sum_{i=1}^{n-1}\lambda_i\overline{g}_i\right)$

Compute $\overline{g}_n = \dfrac{f_{\mu_n^I} - \sum_{i=1}^{n-1}\overline{\lambda}_i^n\overline{g}_i}{\theta_n^I(\mu_n^I)}$ and $\mathbb{E}_{\overline{Z}^{\text{ref}}}(f_{\mu_n^I})$. Set $n = n+1$.

end

Set $N = n$.

We finally wish to evaluate the error made on the approximation of $\mathbb{E}[f_\mu(Z)]$ obtained by using the variance reduction method based on these MC-greedy algorithm. More precisely, this approximation is computed as

$$\sum_{i=1}^{n-1} \lambda_i^{n,\mu} \mathbb{E}_{\overline{Z}^{\mathrm{ref}}}(f_{\overline{\mu}_i}) + \mathbb{E}_{\overline{Z}^n}\left(f_\mu - \sum_{i=1}^{n-1} \lambda_i^{n,\mu} f_{\overline{\mu}_i}\right)$$

where

$$(\lambda_i^{n,\mu})_{1\le i\le n-1} = \underset{(\lambda_i)_{1\le i\le n-1}\in\mathbb{R}^{n-1}}{\operatorname{argmin}} \operatorname{Var}_{\overline{Z}^n}\left(f_\mu - \sum_{i=1}^{n-1} \lambda_i f_{\overline{\mu}_i}\right)$$

and $\overline{\mu}_i$ is equal to μ_i^H, μ_i^{SH} or μ_i^I depending on the chosen algorithm (remember formula (II.1)). This quantity has to be compared with the approximation obtained with a standard Monte-Carlo with M_{ref} samples, i.e. $\mathbb{E}_{\overline{Z}^{\mathrm{ref}}}(f_\mu)$. To this aim, for all $n \in \mathbb{N}^*$ and $\mu \in \mathcal{P}$, we define

$$e_n(\mu) := \frac{\left|\mathbb{E}_{\overline{Z}^{\mathrm{ref}}}(f_\mu) - \left[\sum_{i=1}^{n-1} \lambda_i^{n,\mu} \mathbb{E}_{\overline{Z}^{\mathrm{ref}}}(f_{\overline{\mu}_i}) + \mathbb{E}_{\overline{Z}^n}\left(f_\mu - \sum_{i=1}^{n-1} \lambda_i^{n,\mu} f_{\overline{\mu}_i}\right)\right]\right|}{\left|\mathbb{E}_{\overline{Z}^{\mathrm{ref}}}(f_\mu)\right|}. \tag{II.41}$$

In what follows, we denote by $e_n^H(\mu)$ (respectively $e_n^{SH}(\mu)$) the quantity defined by (II.41) obtained by the HMC-greedy (respectively the SHMC-greedy) algorithm.

II.4.3 Explicit one-dimensional functions

We consider in this section two sets of one-dimensional explicit functions. The motivation for considering these two simple test cases is that the decays of the Kolmogorov n-widths of the associated sets $\mathcal{M}$ are known.

First test case

Let $f : \mathbb{R} \to \mathbb{R}$ be the function defined such that

$$\forall x \in \mathbb{R}, \quad f(x) := \begin{cases} 2\,x & \text{if} \quad 0 \le x \le 0.5, \\ 1 & \text{if} \quad 0.5 \le x \le 1.5, \\ 4 - 2\,x & \text{if} \quad 1.5 \le x \le 2, \\ 0 & \text{otherwise.} \end{cases} \tag{II.42}$$

Let $\mathcal{P} = [0, 3]$ be the set of parameter values. We consider in this first test case the family of functions $(f_\mu)_{\mu\in\mathcal{P}}$ such that $f_\mu(x) = f(x - \mu)$ for all μ in $\mathcal{P}$ and $x \in \mathbb{R}$. Let Z be a real-valued random variable with probability measure $\nu = \mathcal{U}(0, 5)$. In this case, it is known [12] that there exists a constant $c > 0$ such that $d_n(\mathcal{M}) \ge cn^{-1/2}$ for all $n \in \mathbb{N}^*$.

In this example, $M_1 = 10$, $M_{\mathrm{ref}} = 10^5$, $\gamma = 0.9$ and the trial set $\mathcal{P}_{\mathrm{trial}}$ is chosen to be a set of 300 random parameter values which were uniformly sampled in $\mathcal{P}$.

Figure II.1 illustrates the evolution of the values of M_n as a function of n for the HMC and SHMC algorithms.

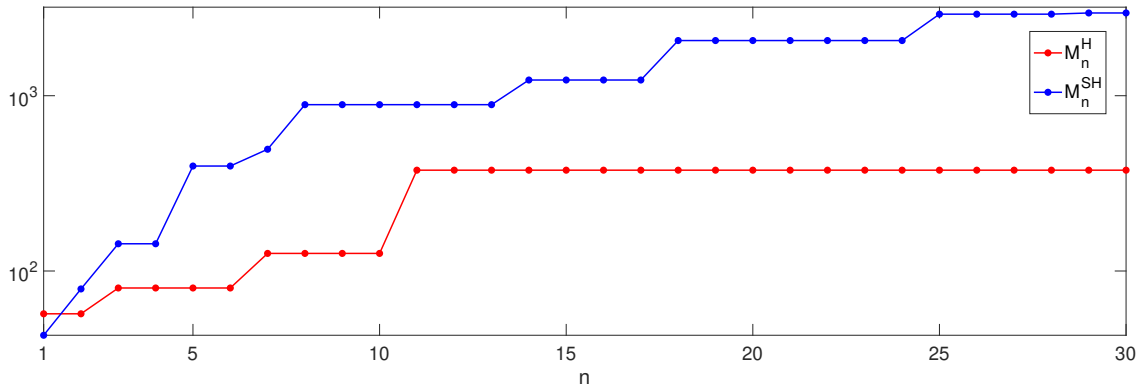


Figure II.1: Evolution of M_n as a function of n for the HMC and SHMC-greedy algorithms in test case 1.

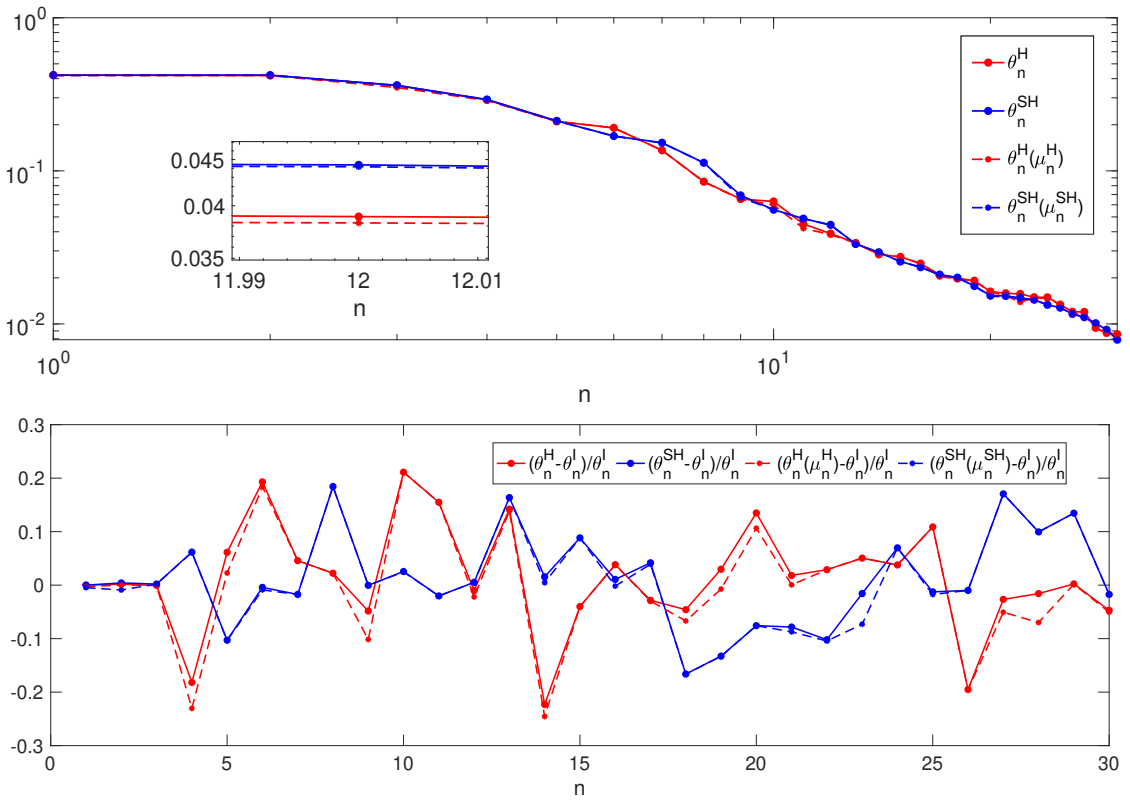


Figure II.2: Evolution of $\theta_n^H(\mu_n^H)$, $\theta_n^{SH}(\mu_n^{SH})$, θ_n^H, θ_n^{SH} as a function of n in test case 1.

Figure II.2 illustrates the fact that at each iteration $n \in \mathbb{N}^*$, for the (S)HMC-algorithm, the value of the selected parameter $\mu_n^{(S)H}$ is relevant since we observe numerically that $\theta^{(S)H}(\mu_n^{(S)H})$ is very close to $\theta_n^{(S)H} = \sup_{\mu \in \mathcal{P}_{\text{trial}}} \theta_n^{(S)H}(\mu)$. In addition, we observe that the resulting reduced spaces $\overline{V}_n^{(S)H}$ have very good approximability properties with respect to the set $\mathcal{M}$, in the sense that the values of $\theta_n^{(S)H}$ and $\theta_n^{(S)H}(\mu_n^{(S)H})$ are very close to θ_n^I, which is computed with the reference IMC algorithm.

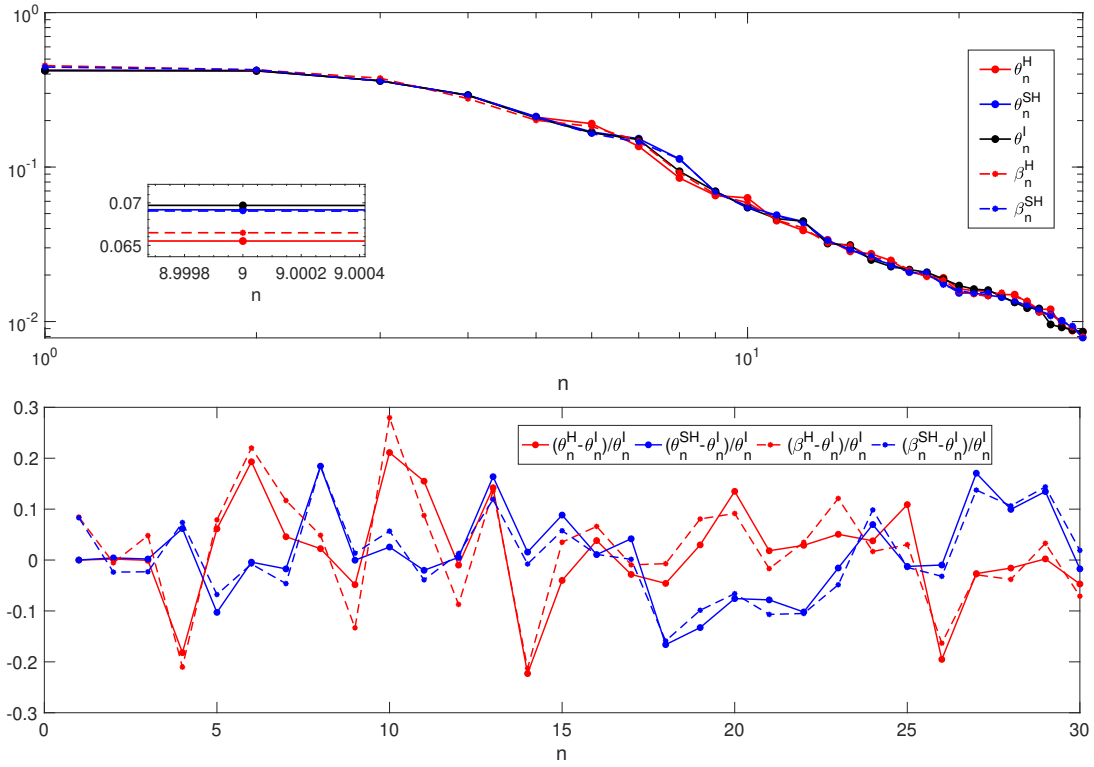


Figure II.3: Evolution of β_n^H, β_n^{SH}, θ_n^H, θ_n^{SH}, θ_n^I as a function of n in test case 1.

Figure II.3 illustrates the fact that the value of the number of samples M_n chosen at each iteration $n \in \mathbb{N}^*$ enables to compute empirical variances that are close to exact variances since the values of $\beta_n^{(S)H}$ are very close to the $\theta_n^{(S)H}$ for the (S)HMC-algorithm.

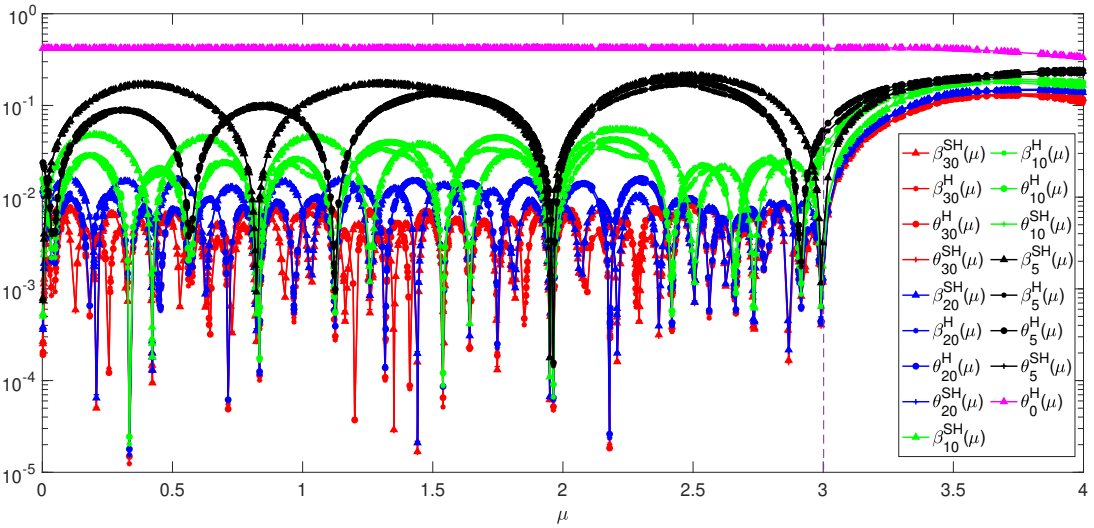


Figure II.4: $\theta_n^H(\mu)$, $\theta_n^{SH}(\mu)$, $\beta_n^H(\mu)$, $\beta_n^{SH}(\mu)$ as a function of μ for $n = 0, 5, 10, 20, 30$ on $\mathcal{P}_{test} = [0, 4]$.

In Figure II.4, the values of $\theta_n^{(S)H}(\mu)$ and $\beta_n^{(S)H}(\mu)$ are plotted as a function of $\mu \in \mathcal{P}_{test} = [0, 4]$ for different values of n ($n = 0, 5, 10, 20, 30$).

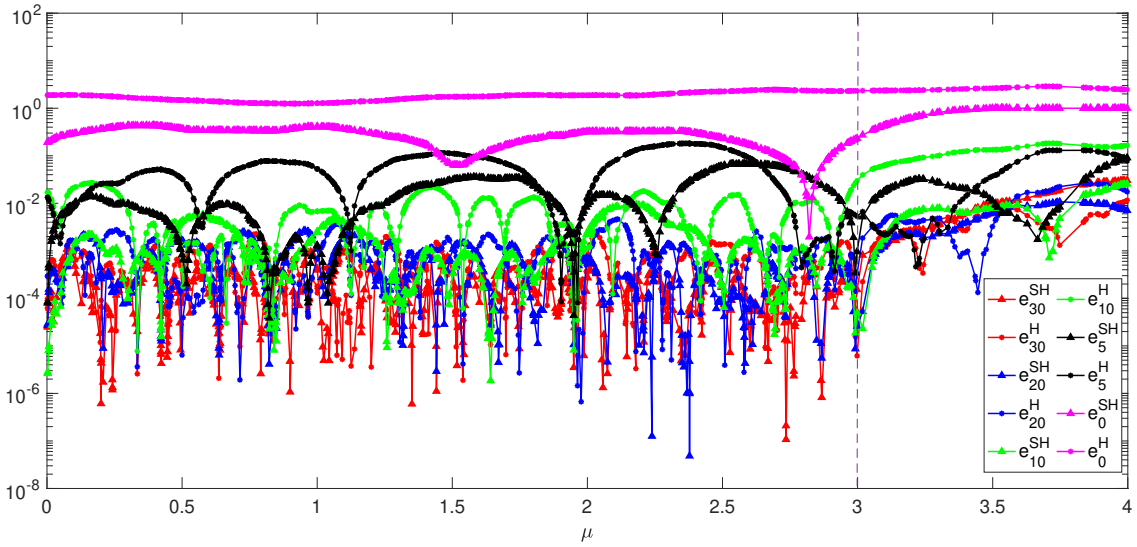


Figure II.5: $e_n^H(\mu)$ and $e_n^{SH}(\mu)$ as a function of μ for $n = 0, 5, 10, 20, 30$ on $\mathcal{P}_{test} = [0, 4]$.

In comparison, in Figure II.5, the relative error $e_n^{(S)H}(\mu)$ is plotted as a function of μ for $n = 0, 5, 10, 20, 30$. In particular, we observe that this error remains lower than 1% as soon as $n \geq 10$ on $\mathcal{P}$. Naturally, this error is larger for $\mu \in \mathcal{P}_{test} \setminus \mathcal{P}$.

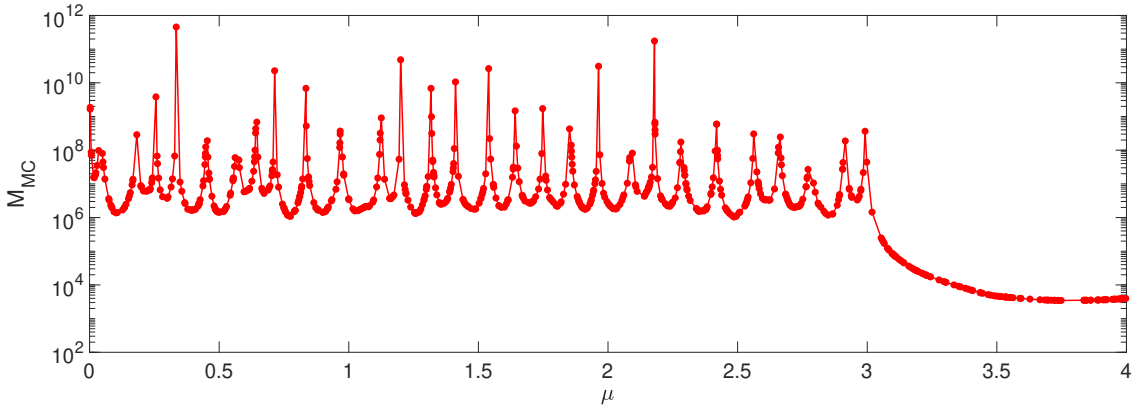


Figure II.6: $M_{MC}(\mu)$ as a function of $\mu \in \mathcal{P}_{test} = [0, 4]$.

Finally, to illustrate the gain of our proposed method in terms of variance reduction, we plot on Figure II.6 the value of the number of random Monte-Carlo samples $M_{MC}(\mu)$ that would have been necessary to compute an approximation of the mean of $f_\mu(Z)$ with a standard Monte-Carlo method with the same level of accuracy than the one given by the HMC-algorithm after $n = 30$ iterations. In this case, let us point out that $M_n = 349$. More precisely, we compute $M_{MC}(\mu)$ by the follwoing formula:

$$M_{MC}(\mu) = \frac{\mathrm{Var}_{\overline{Z}^{ref}}\left(f_\mu\right) \times M_n}{\mathrm{Var}_{\overline{Z}^n}\left(f_\mu - \sum_{i=1}^n \lambda_i^\mu f_{\overline{\mu}_i}\right)}. \tag{II.43}$$

Figure II.6 illustrates that, for all $\mu \in \mathcal{P}$, the classical Monte Carlo method would have required a number of samples $M_{MC}(\mu)$ in the range $10^6 \leq M_{MC}(\mu) \leq 10^{12}$ in order to obtain the same

level of statistical error. Thus, we see that the HMC-algorithm significantly improves the efficiency of the computation of the expectation of $f_\mu(Z)$ with respect to a standard Monte-Carlo algorithm.

Second test case

In this example, we consider a second family of one-dimensional functions where $\mathcal{P} = [0,1]$ is the set of parameter values. More precisely, we consider the family of functions $(f_\mu)_{\mu\in\mathcal{P}}$ such that, for all μ in $\mathcal{P}$:

$$\forall x \in [0,1], \quad f_\mu(x) := \begin{cases} \sqrt{x+0.1} & \text{if} \quad x \in [0,\mu] \\ \frac{1}{2}(\mu+0.1)^{-\frac{1}{2}}x - \frac{1}{2}(\mu+0.1)^{-\frac{1}{2}}\mu + (\mu+0.1)^{\frac{1}{2}} & \text{if} \quad x \in [\mu,1] \end{cases} \tag{II.44}$$

Let us point out that for all $\mu \in \mathcal{P}$, f_μ is a $\mathcal{C}^1$ function on $[0,1]$. In this case, it is known [14] that there exists a constant $c > 0$ such that $d_n(\mathcal{M}) \leq cn^{-2}$ for all $n \in \mathbb{N}^*$.

Let Z be a random variable with probability measure $\nu = \mathcal{U}(0,1)$.

In this example, $M_1 = 10$, $M_{\text{ref}} = 10^5$, $\gamma = 0.9$ and the trial set $\mathcal{P}_{\text{trial}}$ is chosen to be a set of 300 random parameter values which were uniformly sampled in $\mathcal{P}$. In this test case, we osbserve a similar behaviour of the (S)HMC-algorithm as in the first test case.

Figure II.11 illustrates the computational gain brought by the HMC algorithm after $n = 70$ iterations (so that $M_n = 3109$) with respect to the classical Monte Carlo method. Indeed, the quantity $M_{MC}(\mu)$ defined in (II.43) is observed to vary in this case between 10^{12} and 10^{18}.

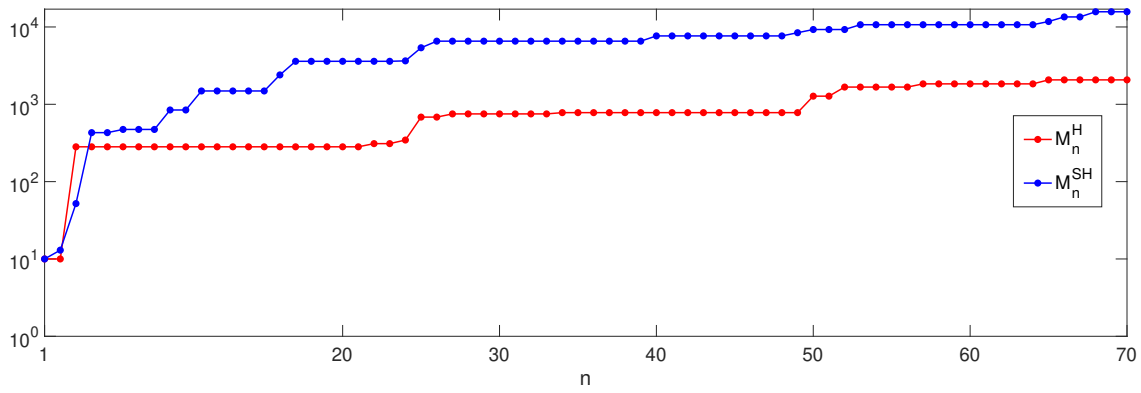


Figure II.7: Evolution of M_n as a function of n for the HMC and SHMC-greedy algorithms.

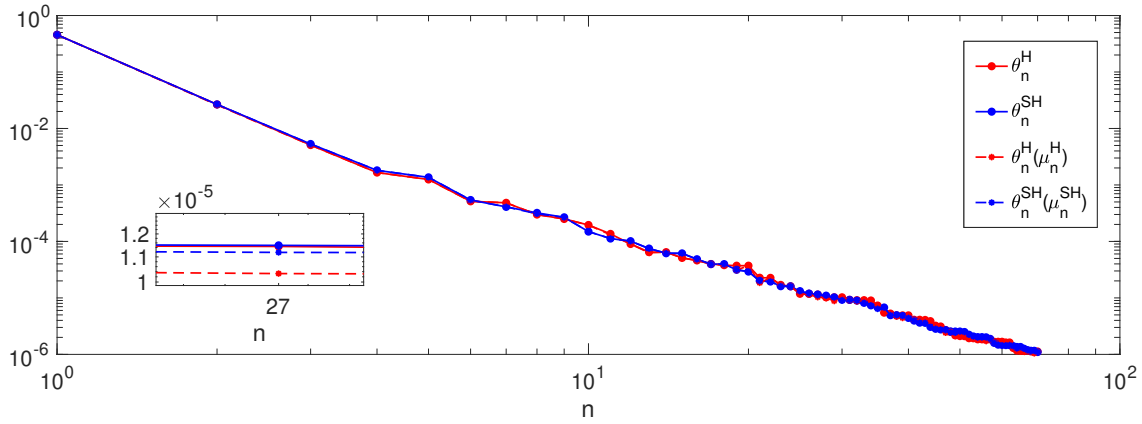


Figure II.8: Evolution of $\theta_n^H(\mu_n^H)$, $\theta_n^{SH}(\mu_n^{SH})$, θ_n^H, θ_n^{SH} as a function of n in test case 2.

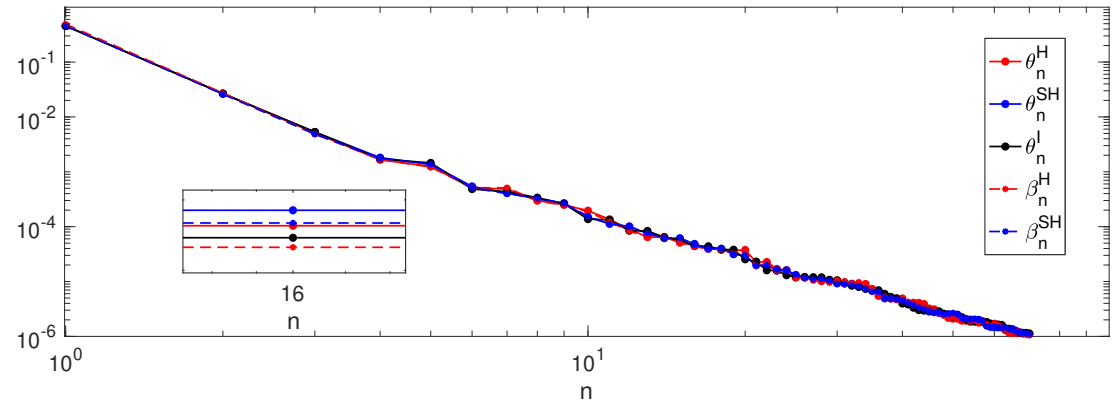


Figure II.9: Evolution of β_n^H, β_n^{SH}, θ_n^H, θ_n^{SH}, θ_n^I as a function of n in test case 2.

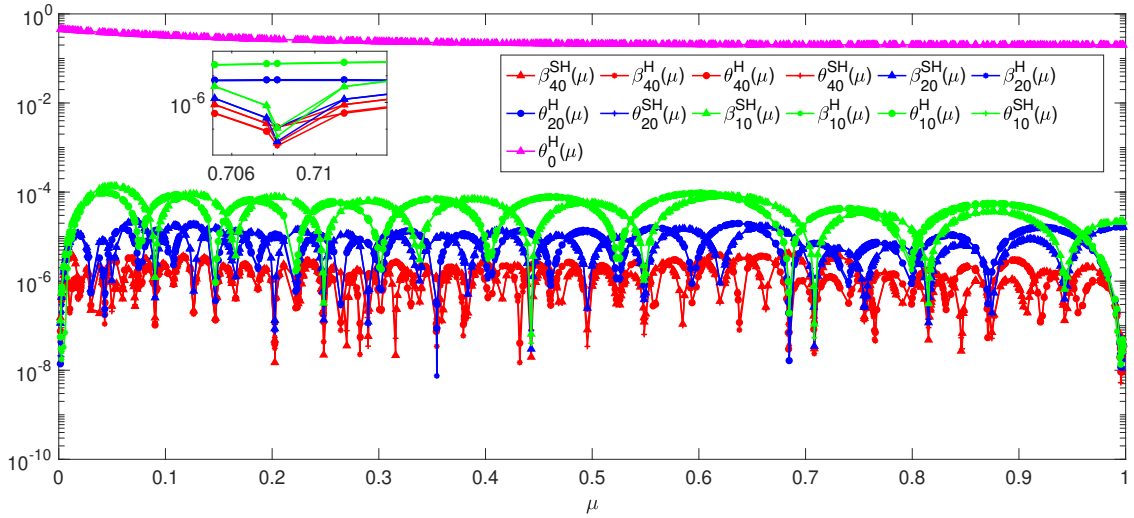


Figure II.10: $\theta_n^H(\mu)$, $\theta_n^{SH}(\mu)$, $\beta_n^H(\mu)$, $\beta_n^{SH}(\mu)$ as a function of μ for $N = 0, 10, 20, 40$ on $\mathcal{P}_{test} = [0, 1]$.

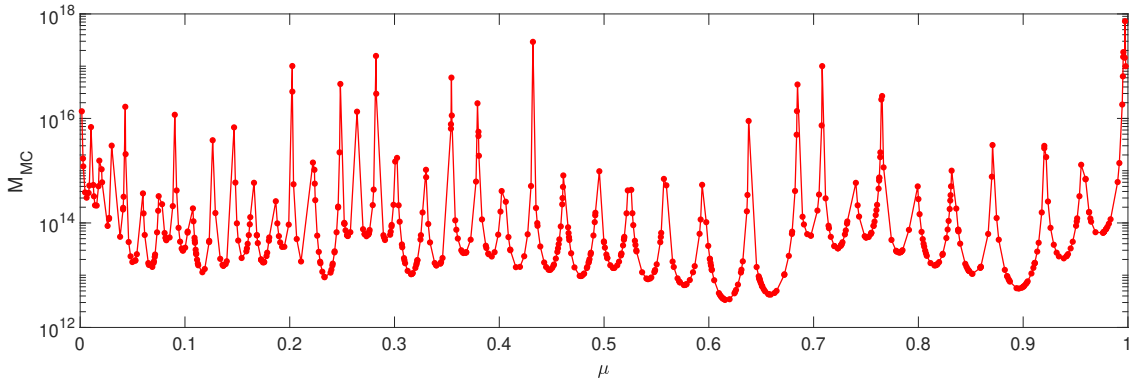


Figure II.11: $M_{MC}(\mu)$ as a function of $\mu \in \mathcal{P}_{test} = [0, 1]$.

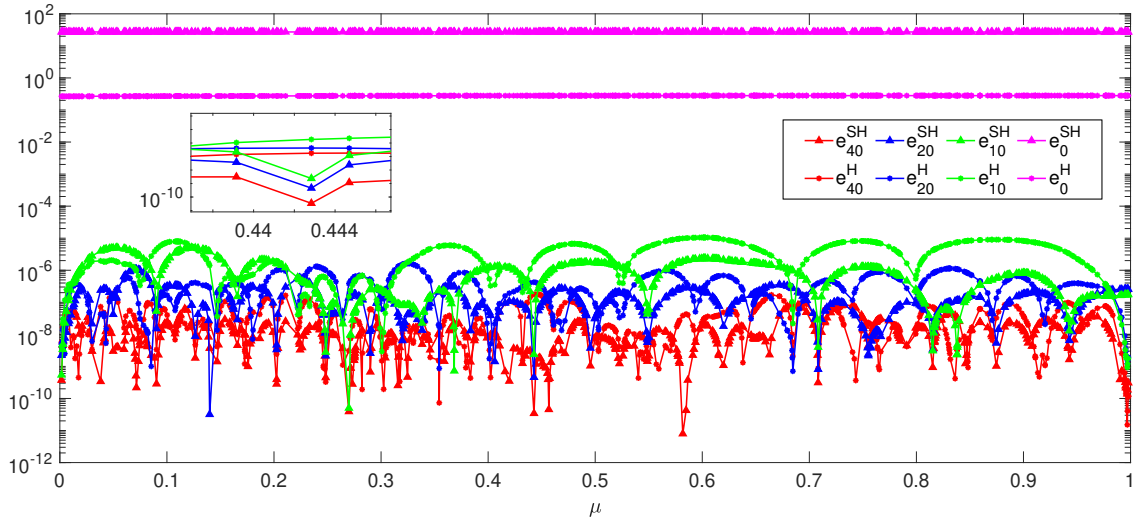


Figure II.12: $e_N^H(\mu)$ and $e_N^{SH}(\mu)$ as a function of μ for $n = 0, 10, 20, 40$ on $\mathcal{P}_{test} = [0, 1]$.

II.4.4 Two-dimensional heat equation

Let Z_1 and Z_2 be two independent real-valued random varibales with probability laws respectively $\mathcal{U}(0.5, 2)$ and $\mathcal{N}(0, 1)$) and let $Z = (Z_1, Z_2)$. Let $\mathcal{D} = (0, 2)^2$, $\mathcal{P} := [0, 10]$. The trial set $\mathcal{P}_{\text{trial}}$ is constructed by selecting 50 random values uniformly distributed in $\mathcal{P}$.

For all $\mu \in \mathcal{P}$ and $z := (z_1, z_2) \in (0, 5, 2) \times \mathbb{R}$, we introduce

$$D^{\mu,z} : \begin{cases} \mathcal{D} & \to & \mathbb{R}^{2\times 2} \\ (x, y) & \mapsto & \begin{bmatrix} D_{11}^{\mu,z}(x, y) & 0 \\ 0 & D_{22}^{\mu,z}(x, y) \end{bmatrix} \end{cases}$$

where

$$\forall (x, y) \in \mathcal{D}, \quad D_{11}^{\mu,z}(x, y) = 13 + \mu \sin(2\pi x/z_1) + 0.5 z_2 \quad \text{and } D_{22}^{\mu,z}(x, y) = 13 + \mu \sin(2\pi y/z_1) + 0.5 z_2.$$

We introduce a conform triangular mesh $\mathcal{T}$ of the domain $\mathcal{D}$ as represented on the left-hand side plot of Figure II.13 and denote by

$$V_h := \{u \in \mathcal{C}(\mathcal{D}), \quad u|_T \in \mathbb{P}_1 \ \forall T \in \mathcal{T}, \quad u|_{\partial\mathcal{D}} = 0\},$$

the standard $\mathbb{P}_1$ finite element space associated to this mesh.

For $\mu \in \mathcal{P}$ and $z \in (0.5, 2) \times \mathbb{R}$, we define $u_h^{\mu,z} \in V_h$ the unique solution to

$$a_{\mu,z}\left(u_h^{\mu,z}, v\right) = b(v), \quad \forall v \in V_h, \tag{II.45}$$

where

$$\forall v, w \in H_0^1(\mathcal{D}), \quad a_{\mu,z} = \int_{\mathcal{D}} \nabla v \cdot D^{\mu,z} \nabla w, \quad b(v) = \int_{\mathcal{D}} rv,$$

and where $r \in L^2(\mathcal{D})$ is defined by

$$r(x, y) = \exp\left(-(x-1)^2 - (y-1)^2\right), \quad \forall (x, y) \in \mathcal{D}.$$

The function $u_h^{\mu,z}$ is thus the standard $\mathbb{P}_1$ finite element approximation of the unique solution $u^{\mu,z} \in H_0^1(\mathcal{D})$ to

$$\begin{cases} -\mathrm{div}\left(D^{\mu,z} \nabla u^{\mu,z}\right) = r, & \text{in } \mathcal{D}, \\ u^{\mu,z} = 0 & \text{on } \partial\mathcal{D}. \end{cases} \tag{II.46}$$

Let $T_1 \in \mathcal{T}$ be the triangle colored in red in the left-hand side plot of Figure II.13. For all $\mu \in \mathcal{P}$ and $z \in (0,5, 2) \times \mathbb{R}$, we define by

$$f_\mu(z) := \frac{1}{|T_1|} \int_{T_1} u_h^{\mu,z}.$$

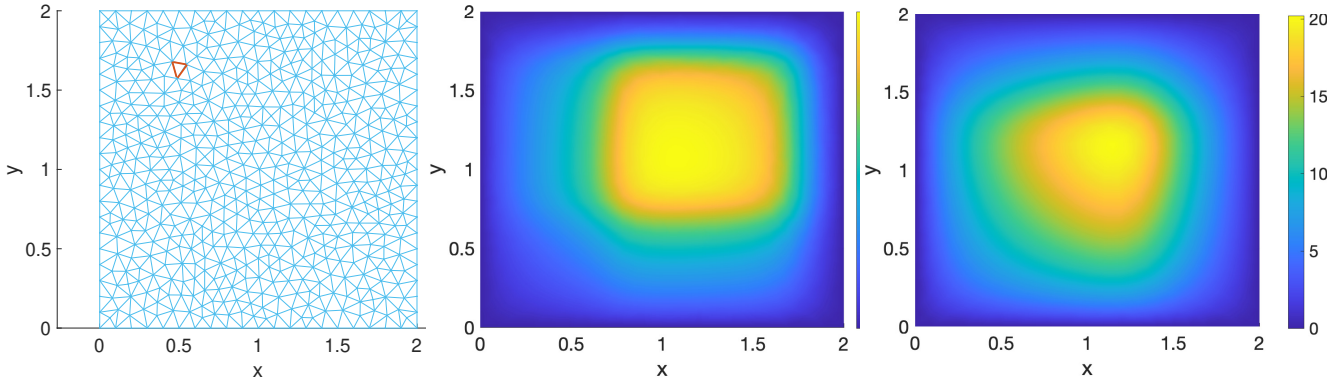


Figure II.13: Left: mesh $\mathcal{T}$ (the triangle T_1 is highlighted in red color); Center: $u_h^{\mu,z}$ for $\mu = 9$ and $z = (1, 0)$; Right: $u_h^{\mu,z}$ for $\mu = 9$ and $z = (1.777, 0.2062)$.

In this example, $M_{\mathrm{ref}} = 10^5$, $M_1 = 800$ and $\gamma = 0.9$. Figure II.14 illustrates the evolution of the values of M_n as a function of n for the HMC and SHMC algorithms.

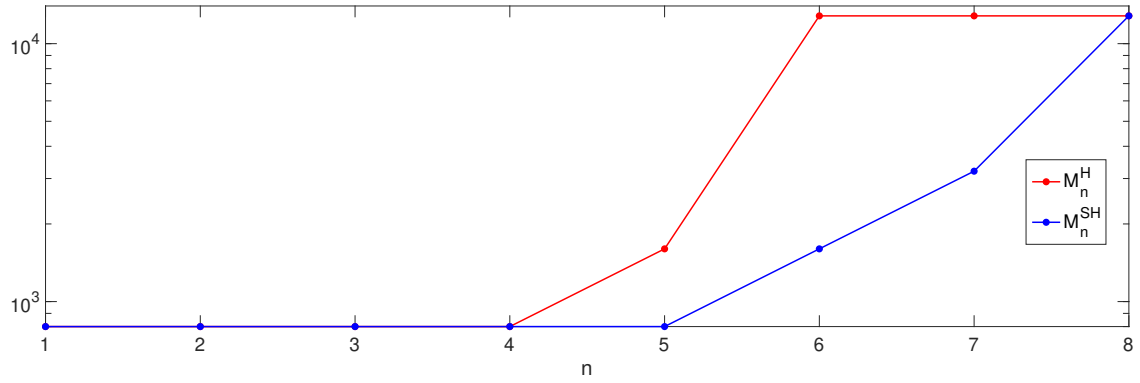


Figure II.14: Evolution of M_n as a function of n for the HMC and SHMC-greedy algorithms in test case 3.

It is to be noted here, from Figure II.15 and Figure II.16 that the quantities θ_n^H, θ_n^{SH} and θ_n^I are very close: the quality of approximation of the reduced spaces V_n^H or V_n^{SH} is very close to the quality of approximation of the reduced space V_n^I given by an ideal greedy algorithm.

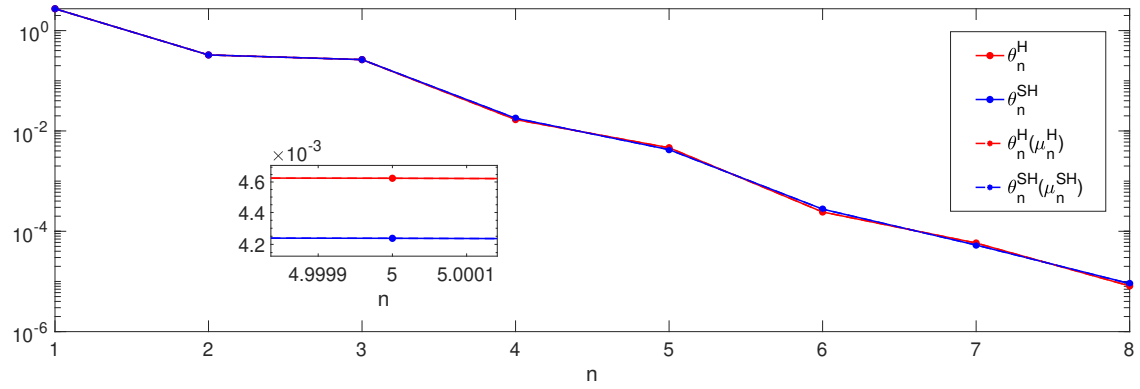


Figure II.15: Evolution of $\theta_n^H(\mu_n^H)$, $\theta_n^{SH}(\mu_n^{SH})$, θ_n^H, θ_n^{SH}as a function of n in test case 3.

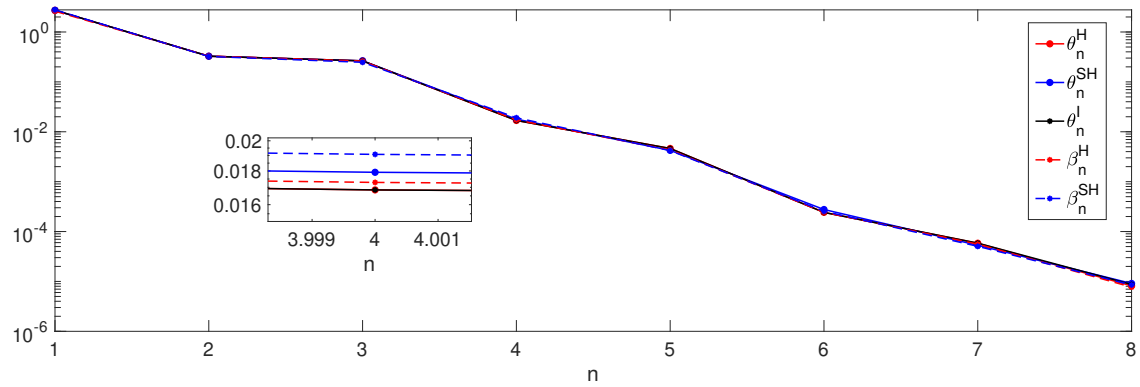


Figure II.16: Evolution of β_n^H, β_n^{SH}, θ_n^H, θ_n^{SH}, θ_n^I as a function of n in test case 3.

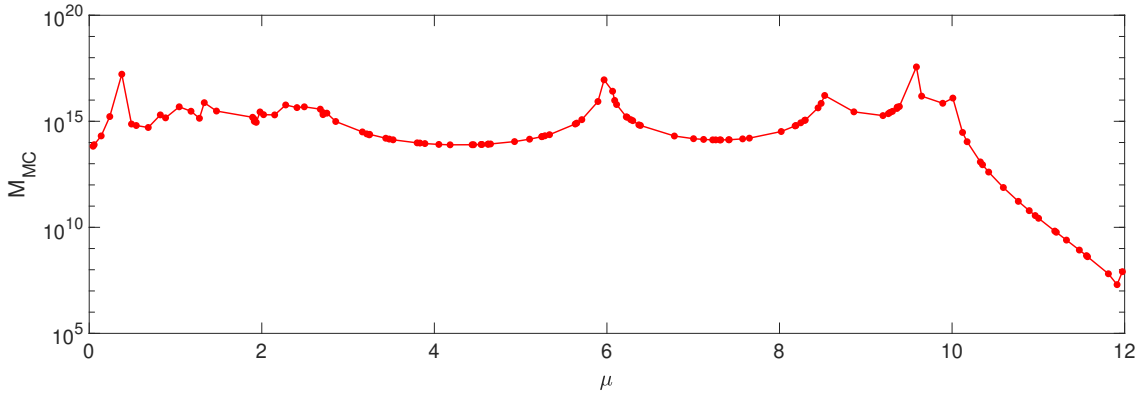


Figure II.17: $M_{MC}(\mu)$ as a function of $\mu \in \mathcal{P}_{test} = [0, 12]$.

Figure II.17 shows the value of $M_{MC}(\mu)$ given by (II.43), knowing that $M_n = 12800$ after $n = 7$ iterations of the HMC algorithm. We observe that in this case $10^{14} \leq M_{MC}(\mu) \leq 10^{20}$, which shows the huge computational gain brought by the HMC algorithm with respect to a standard Monte-Carlo method in this test case.

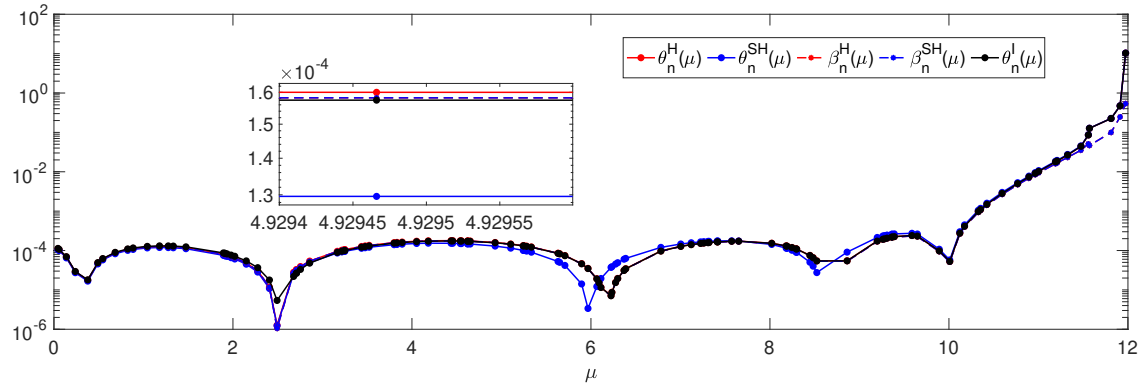


Figure II.18: $\theta_n^H(\mu)$, $\theta_n^{SH}(\mu)$, $\theta_n^I(\mu)$, $\beta_n^H(\mu)$, $\beta_n^{SH}(\mu)$ as a function of μ for $n = 5$ in test case 3.

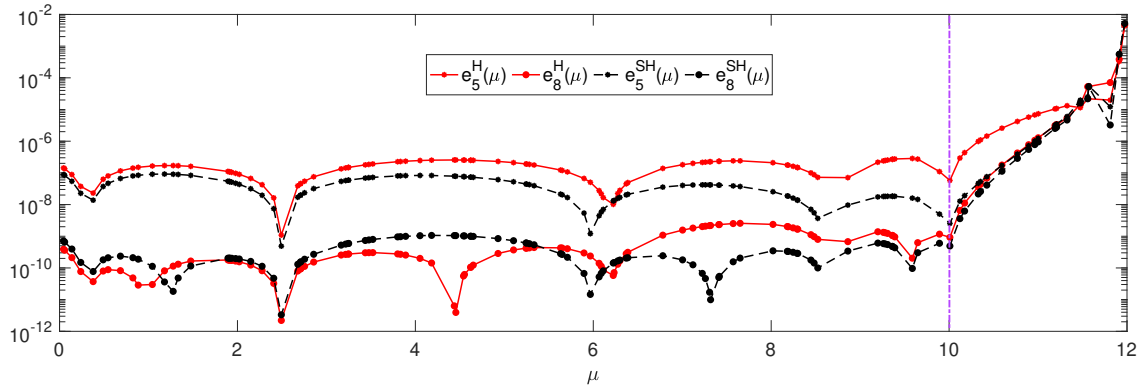


Figure II.19: $e_n^H(\mu)$ and $e_n^{SH}(\mu)$ as a function of μ for $n = 5$ and $n = 8$ in test case 3.

CHAPTER III

DYNAMICAL ORTHOGONAL APPROXIMATION OF PARAMETRIC STOCHASTIC DIFFERENTIAL EQUATIONS

Contents

Abstract

In this work we aim to decompose a stochastic process solution of a parameteric stochastic differential equation using a dynamical low rank approximation.
We look for a decomposition with two kinds of dynamical modes. The first modes operate on the parameter space, while the second ones operate on the stochastic space. For this, we adapt for the parametric SDE problems an explicit scheme developed by Lubich and Oseledets ([19]) used for the ODE cases. Indeed, in the later case, the method shows high numerical robustness as producing quasi optimal low rank approximation compared to the best approximation in the Frobenuis norm.
The method is applied to a matrix $X(t) \in \mathbb{R}^{d\times p}$ solution to a parametric stochastic differential equation at each time $t \in [0, T]$ and leads to a significant gain in memory and in computational time.
We propose splitting schemes for additive and multiplicative stochastic differential equation, and we use this scheme (in the additive case) to construct a control variate in order to calculate very efficently the quantities of interest. An example on an SDE non linear in the sense of McKean illustrates the efficency of the method.

III.1 Introduction

The aim of this work is to study a low-rank approximation technique for the approximation of parametric Stochastic Differential Equations, namely the so-called dynamical low-rank approximation method which was introduced for the approximation of the solution of Ordinary Differential Equations in [21]. The principle of the method is the following: let $(X_t^\mu)_{t\geq 0}$ be the solution of a Stochastic Differential Equation depending on a parameter $\mu \in \mathcal{P} \subset \mathbb{R}^p$ for some $p \in \mathbb{N}^*$. The dynamical low-rank method then consists in approximating X_t^μ, for all $t \geq 0$ and all $\mu \in \mathcal{P}$, under the following form:

$$X_t^\mu \approx \sum_{k=1}^{r} b_k(t;\mu) Y_t^k,$$

where $r \in \mathbb{N}^*$ and for all $1 \leq k \leq r$, $(Y_t^k)_{t\geq 0}$ is a parameter-independent stochastic process and $b_k : \mathbb{R}_+ \times \mathcal{P} \to \mathbb{R}$ is a deterministic function.

The processes $(Y_t^k)_{t\geq 0}$ and functions b_k are obtained as solutions of a coupled time-dependent system, which is solved numerically by means of a projector-splitting scheme, similar to the scheme proposed by Oseledets and Lubich in [19]. The aim of this chapter is to provide some numerical studies of the behaviour of such splitting schemes for the low-rank approximation of the solutions of parametric Stochastic Differential Equations.

The outline of the chapter is the following. In Section III.2, we recall some well-known results about the Dynamical Orthogonal method for the reduction of systems of Ordinary Differential equations. In Section III.3, we present the various splitting schemes we propose in order to compute dynamical low-rank approximations of parametric Stochastic Differential Equations with additive noise. In Section III.4 we illustrate the numerical behaviour of these different schemes on the case of an Overdamped Langevin process. In Section III.5 we generelize the scheme to the case of an SDE with multiplicative noise and we apply them on a non linear SDE in the sense of McKean. In Section III.6 we use the projector splitting algorithm to construct an algorithm that gives a control variate in order to caculate fast some quantities of interest.

III.2 Dynamical low-rank method for Ordinary Differential Equations

III.2.1 Parametric Ordinary Differential Equations

Let us briefly recall the principle of the dynamical low-rank approximation method, for the approximation of parametric Ordinary Differential Equations. Let $\mathcal{P} \subset \mathbb{R}^m$ denote a set of parameter values for some $m \in \mathbb{N}^*$ and consider the solution of the following parametric Cauchy-Lipschitz problem. For all $\mu \in \mathcal{P}$ and all $X_0^\mu \in \mathbb{R}^d$, let $(X_t^\mu)_{t\geq 0}$ be the unique solution of

$$\left\{ \begin{array}{l} \frac{d}{dt} X_t^\mu = \mathcal{F}^\mu(t; X_t^\mu), \\ X_{t=0}^\mu = X_0^\mu, \end{array} \right\} \tag{III.1}$$

where for all $\mu \in \mathcal{P}$, $\mathcal{F}^\mu : \mathbb{R}_+ \times \mathbb{R}^d$ is a Lispchitz function.

Let us assume here that the set of parameter values $\mathcal{P}$ is a finite set of cardinality $p \in \mathbb{N}^*$. Let us denote by $\mu_1, \cdots, \mu_p$ the elements of $\mathcal{P}$. For all $t \in [0, T]$, let us denote by $X(t) \in \mathbb{R}^{d\times p}$ the matrix defined such that its q^{th} column is equal to $X^{\mu_q}(t)$ for all $1 \leq q \leq p$. Let us also denote by $\mathcal{F} : [0, T] \times \mathbb{R}^{d\times p} \to \mathbb{R}^{d\times p}$ the function defined by: for all $X = (X_1, \cdots, X_p) \in \mathbb{R}^{d\times p}$ and for all $1 \leq q \leq p$,

$$\left(\mathcal{F}(t, X)\right)_q := \mathcal{F}^{\mu_q}(t; X_q),$$

where $\left(\mathcal{F}(t, X)\right)_q$ is the q^{th} column of $\mathcal{F}(t, X)$.

Problem (III.1) can then be rewritten equivalently under the following form:

$$\left\{ \begin{array}{l} \dot{X}(t) = \mathcal{F}(t; X(t)), \ \forall t \in [0, T], \\ X(0) = X_0, \end{array} \right. \tag{III.2}$$

where $X_0 = (X_0^{\mu_1}, \cdots, X_0^{\mu_p}) \in \mathbb{R}^{d\times p}$.

III.2.2 Principle of the Dynamical Orthogonal method

The principle of the dynamical low-rank approximation method introduced in [21] is the following: for a given value of $r \in \mathbb{N}^*$ and for all $t \in [0,T]$, the matrix $X(t)$ is approximated by an element of the low-rank manifold of matrices of $\mathbb{R}^{d\times p}$ of rank r:

$$\mathcal{R}_r = \{X_r \in \mathbb{R}^{d\times p},\, rg(X_r) = r\},.$$

It is well-known that for all $t \in [0,T]$, a best rank-r approximation of the matrix $X(t)$, solution of the following minimisation problem:

$$X_r(t) \in \underset{X_r\in\mathcal{R}_r}{\arg\min} \|X(t) - X_r\|_F\,, \tag{III.3}$$

can be obtained as a truncated rank-r Singular Value Decomposition of the matrix $X(t)$.

For all $t \in [0,T]$, there exists a unitary matrix $U(t) = (U_1(t), \cdots, U_d(t)) \in \mathbb{R}^{d\times d}$, a unitary matrix $V(t) = (V_1(t), \cdots, V_p(t)) \in \mathbb{R}^{p\times p}$ and a diagonal matrix $S(t) := (S_{ij}(t))_{1\le i\le d,\, 1\le j\le p} \in \mathbb{R}^{d\times p}$ with non-negative coefficients such that

$$X(t) = U(t)S(t)V(t)^T.$$

Assuming that $r \le \min(p,d)$, one best rank-r approximation of the matrix $X(t)$ is then given by

$$X_r(t) = \overline{U}_r(t)\overline{S}_r(t)\overline{V}_r(t)^T$$

where $\overline{U}_r(t) := (U_1(t), \cdots, U_r(t)) \in \mathbb{R}^{d\times r}$, $\overline{V}_r(t) := (V_1(t), \cdots, V_r(t)) \in \mathbb{R}^{p\times r}$ et $\overline{S}_r(t) := (S_{ij}(t))_{1\le i,j\le r} \in \mathbb{R}^{r\times r}$.

The Dynamical Orthogonal method was developed by Lubich and Koch [21] in order to compute a low-rank approximation of large systems of ordinary differential equations. It consists in computing at each time $t > 0$ an approximation $Y(t) \in \mathcal{R}_r$ to the matrix $X(t) \in \mathbb{R}^{d\times p}$ such that the a priori knowledge of the solution $X(t)$ is not necessary. Dynamical orthogonal equations are derived as follows: an approximation $Y(t) \in \mathcal{R}_r$ is computed such that for all $t > 0$

$$\begin{cases} \dot{Y}(t) \in \operatorname{argmin}_{Z\in\mathcal{T}_{Y(t)}\mathcal{R}_r} \|Z - \mathcal{F}(t;Y(t))\|_F \\ Y(0) = X_r(0), \end{cases} \tag{III.4}$$

where for all $Y \in \mathcal{R}_r$, $\mathcal{T}_Y\mathcal{R}_r$ denotes the tangent space to the manifold $\mathcal{R}_r$ at point $Y \in \mathcal{R}_r$, and where $\|\cdot\|_F$ denotes the Frobenius norm.

If $Y(t)$ solves (III.4), it holds that for all $t > 0$, since $Y(t) \in \mathcal{R}_r$, there exists a unitary matrix $U_t \in \mathbb{R}^{d\times r}$, a unitary matrix $V_t \in \mathbb{R}^{p\times r}$ and a non-singular matrix $S_t \in \mathbb{R}^{r\times r}$ such that

$$Y(t) = U_t S_t V_t^T. \tag{III.5}$$

More precisely, the following relationships hold for the matrices U_t and V_t:

$$U_t^T U_t = I_r \quad \text{and} \quad V_t^T V_t = I_r \tag{III.6}$$

Naturally, the decomposition (III.5) is not unique, which is an issue in order to compute $Y(t)$ in practice.

Denoting by $\mathcal{V}_{d\times r}$ (respectively $\mathcal{V}_{p\times r}$) the Stiefel manifold of unitary matrices of size $d \times r$ (respectively $p \times r$), it holds that:

$$\forall U \in \mathcal{V}_{d\times r}, \quad \mathcal{T}_U\mathcal{V}_{d\times r} = \left\{\partial U \in \mathbb{R}^{d\times r},\ \partial U^T U + U^T \partial U = 0\right\}.$$

For all $Y \in \mathcal{R}_r$ which can be written under the form (III.5) i.e $Y = USV$, it then holds that

$$\mathcal{T}_Y\mathcal{R}_r = \left\{\partial Y = \partial U S V^T + U \partial S V^T + U S \partial V^T,\ \partial U \in \mathcal{T}_U\mathcal{V}_{d\times r},\ \partial V \in \mathcal{T}_V\mathcal{V}_{p\times r},\ \partial S \in \mathbb{R}^{r\times r}\right\}. \tag{III.7}$$

It then holds that for all $\partial Y \in \mathcal{T}_Y\mathcal{R}_r$, the matrices ∂S, ∂U and ∂V are uniquely determined if the following additional orthogonality condition is required:

$$U^T \partial U = 0 \quad \text{and} \quad V^T \partial V = 0. \tag{III.8}$$

The low-rank approximation $Y(t)$ is then defined as the solution of the following dynamical system on the space $\mathcal{R}_r$:

$$\begin{cases} \dot{Y}(t) = \Pi_{\mathcal{T}_{Y(t)}\mathcal{R}_r} \mathcal{F}(t; Y(t)), \\ Y(0) = X_r(0), \end{cases} \tag{III.9}$$

where for all $Y \in \mathcal{R}_r$, $\Pi_{\mathcal{T}_Y\mathcal{R}_r}$ denotes the orthogonal projector on the tangent space $\mathcal{T}_Y\mathcal{R}_r$ at the point Y.

III.2.3 Theoretical results on the Dynamical Orthogonal method for Ordinary Differential Equations

In this section, we recall some theoretical results on the analysis of the dynamical orthogonal method for the reduction of ODE problems.

The following result holds on $X(t)$ solution of (III.2) and $Y(t)$ solution of (III.9).

Theorem III.2.1. *[21] Let us assume that for all $t \geq 0$ there exists a best rank-r approximation $X_r(t)$ of $X(t)$ in $\mathcal{R}_r$ such that the mapping $t \mapsto X_r(t)$ is continuously differentiable on $[0, T]$. Let $\rho > 0$. For all $t \in [0, T]$, let $\lambda_r(X(t))$ denote the r^{th} singular value of $X(t)$ and let us assume that*

$$\forall t \in [0, T], \quad \lambda_r(X(t)) \geq \rho > 0.$$

Let us also assume that there exists $\mu > 0$ such that for all $t \in [0, T]$,

$$\left\|\dot{X}(t)\right\|_F \leq \mu.$$

Lastly, let us assume that

$$\|X_r(t) - X(t)\|_F < \frac{1}{16}\rho \quad , \forall t \in [0, T]. \tag{III.10}$$

Then, it holds that

$$\forall t \in [0, T], \quad \|Y(t) - X_r(t)\|_F \leq 2\beta e^{\beta t} \int_0^t \|X_r(s) - X(s)\|_F \, ds,$$

with $\beta := 8\mu\rho^{-1}$.

Note that this result was later improved by Feppon and Lermusiaux in [15].

III.2.4 Numerical schemes for the resolution of the Dynamical Orthogonal system

The aim of this section is to present two numerical schemes for the resolution of the Dynamical Orthogonal system (III.9).

SVD scheme

We first consider a numerical scheme proposed in [15], which requires the computation of an SVD at each time step (and thus is quite costly from a computational point of view).

Let $\Delta t > 0$, $t_n := n\Delta t$ for all $n \in \mathbb{N}$ and let us denote by Y^n the numerical approximation of $Y(t_n)$ given by the numerical scheme.

The numerical SVD scheme proposed in [15] consists in computing, for all $n \in \mathbb{N}$,

$$\begin{cases} Y^{n+1} := \Pi_{\mathcal{R}_r}(Y^n + \Delta t \mathcal{F}(t_n, Y^n)), \\ Y(0) = X_r(0) = \Pi_{\mathcal{R}_r} X(0), \end{cases} \tag{III.11}$$

where, for all $M \in \mathbb{R}^{d\times p}$, $\Pi_{\mathcal{R}_r} M$ denotes one best rank-r approximation of the matrix M, which is typically obtained by a truncated SVD decomposition of the matrix M.

Then, the following result holds.

Theorem III.2.2. *[15] Let $T > 0$ and $N \in \mathbb{N}^*$. Let $\Delta t := \frac{T}{N}$ and for all $0 \leq n \leq N$, let $t_n = n\Delta t$. Let $(Y^n)_{0\leq n\leq N}$ be the sequence obtained by the discretized system (III.11) and assume that $\mathcal{F}$ is Lipschitz continuous. Then, $(Y^n)_{0\leq n\leq N}$ uniformly converges to the dynamical orthogonal solution $(Y(t))_{t\in[0,T]}$ of (III.9) in the following sense:*

$$\sup_{0\leq n\leq \frac{T}{\Delta t}} \|Y^n - Y(t_n)\|_T \underset{\Delta t\longrightarrow 0}{\longrightarrow} 0. \tag{III.12}$$

Splitting scheme

As mentioned above, the SVD scheme is quite expensive from a computational point of view. This is the reason why a (cheaper) splitting scheme has been introduced in [19] in order to solve (III.9). We present this splitting scheme in this section. The main objective of this chapter is to study from a numerical point of view the behaviour of this splitting scheme (and variants) for the resolution of Dynamical Orthogonal equations for the approximation of the solutions of parametric SDEs.

For any $Y \in \mathcal{R}_r$, there exists $U \in \mathcal{V}_{d,r}$, $V \in \mathcal{V}_{p,r}$ and $S \in \mathbb{R}^{r\times r}$ such that $Y = USV^T$. Hence, using the explicit characterization of $\mathcal{T}_Y\mathcal{R}_r$ given in (III.7), it holds that the orthogonal projector $\Pi_{\mathcal{T}_Y\mathcal{R}_r}$ onto the tangent space to $\mathcal{R}_r$ at Y has the following expression:

$$\forall Z \in \mathbb{R}^{d\times p}, \quad \Pi_{\mathcal{T}_Y\mathcal{R}_r} Z = ZVV^T - UU^TZVV^T + UU^TZ.$$

Introducing the orthogonal projector $P_U := UU^T$ and $P_V := VV^T$, we thus obtain that

$$\forall Z \in \mathbb{R}^{d\times p}, \quad \Pi_{\mathcal{T}_Y\mathcal{R}_r} Z = ZP_V - P_UZP_V + P_UZ.$$

The splitting scheme introduced in [19] for the resolution of (III.9) is a three-step scheme. For all $n \in \mathbb{N}^*$, Y^{n+1} is computed from Y^n as follows: assume that $Y^n := U^n S^n (V^n)^T$ for some $U^n \in \mathcal{V}_{d,r}$, $V^n \in \mathcal{V}_{p,r}$ and $S^n \in \mathbb{R}^{r\times r}$,

1) $Y_1^n := Y^n + \Delta t \mathcal{F}(t, Y^n) P_{V^n}$; compute $Y_1^n := U^{n+1} S_1^n (V^n)^T$ with $U^{n+1} \in \mathcal{V}_{d,r}$ and $S_1^n \in \mathbb{R}^{r\times r}$; to do so, compute a QR decomposition of the matrix $Y_1^n V^n$;

2) $Y_2^n = Y_1^n - \Delta t P_{U^{n+1}} \mathcal{F}(t, Y_1^n) P_{V^n}$; compute $Y_2^n := U^{n+1} S_2^n (V^n)^T$ with $S_2^n \in \mathbb{R}^{r\times r}$; actually, $S_2^n := (U^{n+1})^T Y_2^n V^n$;

3) $Y^{n+1} = Y_2^n + \Delta t P_{U^{n+1}} \mathcal{F}(t, Y_2^n)$; compute $Y^{n+1} := U^{n+1} S^{n+1} (V^{n+1})^T$ with $V^{n+1} \in \mathcal{V}_{p,r}$ and $S^{n+1} \in \mathbb{R}^{r\times r}$; to do so, compute a QR decomposition of the matrix $(U^{n+1})^T Y^{n+1}$.

It is proved in [19] that this splitting scheme is of order 1 for the approximation of the Dynamical Orthogonal equations for the reduction of an ODE system of the form (III.2).

The following result gives an upper bound for the dynamical orthogonal approximation error when the matrix $X(t)$ is solution of problem of the type (III.2).

Theorem III.2.3. *[18] Let $X(t)$ be the solution of the problem (III.2) on the interval $[0, T]$, assume that there exist $B > 0$, $L > 0$ and $\epsilon > 0$, such that for all $Y \in \mathbb{R}^{d\times p}$ and $\tilde{Y} \in \mathbb{R}^{d\times p}$ we have,*

- $||\mathcal{F}(t, Y) - \mathcal{F}(t, \tilde{Y})||_F \leq L||Y - \tilde{Y}||_F$ *et* $||\mathcal{F}(t, Y)||_F \leq B$
- *The non-tangential part of $\mathcal{F}(t, Y)$ is ϵ-small,*

$$||(I - \mathcal{T}_Y \mathcal{M}_r)\mathcal{F}(t, Y)||_F \leq \epsilon$$

  for all $Y \in \mathcal{R}_r$ and $0 \leq t \leq T$,

- *The error in the intial value is δ-small,*

$$||Y^0 - X(0)||_F \leq \delta$$

Let Y^n be the rank r approximation of $X(t_n)$ at $t_n = n \times \Delta t$ obtained after n steps of the Projector-Splitting with step size Δt. Then the approximation error satisfies, for all n such that $t_n \leq T$:

$$||Y^n - X(t_n)||_F \leq c_0\epsilon + c_1\Delta t + c_2\delta \tag{III.13}$$

Where c_0, c_1 et c_2 are constants that only depend on B, L and T.

Remarks:

- This result proves that the Projector-Splitting is a first order scheme, note that one can use a higher order scheme, which is done in [19].
- Theorem (III.2.3) shows that the approximation error is bounded by a term independantly from the singular values of $X(t)$, this shows that the Projector-Splitting integrator is insensible to the low singular values. Hence an over-ranking approximation does not affect the error.
- The approximation error of the projector splitting is proportional to the magnitude of the non-tangential part of $\mathcal{F}(t, Y(t))$ at each time t.

III.3 Splitting schemes for the resolution of Dynamical Orthogonal equations for parametric stochastic differential equations with additive noise

The aim of this chapter is to study from a numerical point of view the behaviour of an adaptation of the splitting scheme described above and an adaptative variant for the resolution of Dynamical Orthogonal Equations for parametric Stochastic Differential Equations.

Let us first introduce some notation. Let $(\Omega, F, \mathbb{P})$ be a probability space. Let us consider the parametric stochastic differential equation of the following form: for all $\mu \in \mathcal{P}$, find $(X_t^\mu)_{0\leq t\leq T}$ solution to

$$dX_t^\mu = b^\mu(t; X_t^\mu)\, dt + \sigma^\mu(t; X_t^\mu)\, dW_t, \tag{III.14}$$

where $(W_t)_{0\leq t\leq T}$ is a Brownian motion, and for all $\mu \in \mathcal{P}$, $b^\mu : [0,T]\times\mathbb{R} \to \mathbb{R}$ and $\sigma^\mu : [0,T]\times\mathbb{R} \to \mathbb{R}_+$ are smooth functions. Let us assume that for all $\mu \in \mathcal{P}$, there exists a unique strong solution to (III.14).

Assuming now that we wish to build a reduced-order model for the approximation of the solution of (III.14) for values of parameters μ belonging to a finite subset $\mathcal{P}_{\text{train}} \subset \mathcal{P}$ of cardinality $p \in \mathbb{N}^*$. Let us denote by $\mu_1, \cdots, \mu_p$ the elements of $\mathcal{P}_{\text{train}}$. Then, denoting by $X_t := (X_t^{\mu_1}, \cdots, X_t^{\mu_p})$ for all $0 \leq t \leq T$, we have,

$$dX_t = B(t, X_t)\, dt + \Sigma(t, X_t)\, dW_t \tag{III.15}$$

where for all $X := (X_1, \cdots, X_p) \in \mathbb{R}^p$ and all $t \in [0,T]$, $B(t,X) := (B_i(t,X))_{1\leq i\leq p} \in \mathbb{R}^p$ and $\Sigma(t,X) := (\Sigma_i(t,X))_{1\leq i\leq p} \in \mathbb{R}^p$ are defined such that

$$\forall 1 \leq i \leq p, \quad B_i(t,X) = b^{\mu_i}(t, X_i) \quad \text{ and } \Sigma_i(t,X) = \sigma^{\mu_i}(t, X_i).$$

Equation (III.15) is supplemented with the initial condition $X_{t=0} = X_0$ for some p-dimensional random vector X_0.

The aim of a dynamical orthogonal method will be to compute an approximation of X_t under the form

$$X_t \approx Y_t := \sum_{k,l=1}^{r} U_t^k S^{k,l}(t) V^l(t)$$

where for all $1 \leq k, l \leq r$, $V^k : [0,T] \to \mathbb{R}^p$ and $S^{k,l} : [0,T] \to \mathbb{R}$ are deterministic functions and where $(U_t^k)_{0\leq t\leq T}$ are real-valued stochastic processes.

Before presenting the splitting scheme, for the computation of a low-rank approximation to $(X_t)_{0\leq t\leq T}$, let us first recall the classical Euler-Maruyama scheme for the time discretization of (III.15). Let $\Delta t > 0$ and for all $n \in \mathbb{N}$, $t_n := \Delta t$. Let $(G_n)_{n\in\mathbb{N}}$ be a sequence of independent identically distributed random variables with normal law. The Euler-Maruyama scheme then consists in computing, for all $n \in \mathbb{N}$,

$$X^{n+1} = X^n + \Delta t B(t_n, X^n) + \sqrt{\Delta t}\Sigma(t_n, X^n) G_n. \tag{III.16}$$

Let us now present different variants of splitting schemes to compute a low-rank approximation $(Y_t)_{0\leq t\leq T}$ of the stochastic process $(X_t)_{0\leq t\leq T}$.

We first restrict our presentation to the case where for all $t \in [0,T]$ and all $X \in \mathbb{R}^p$,

$$\Sigma(t,X) = \sigma(1,1,\cdots,1)^T$$

for some constant $\sigma > 0$. In the sequel, we denote by $\Sigma := \sigma(1,1,\cdots,1)^T$. In other words, we first restrict our presentation here to the case of an additive noise, i.e. when $\Sigma(t,X) = \Sigma$ for all $t \in [0,T]$ and $X \in \mathbb{R}^p$.

III.3.1 A splitting scheme without projection

First, let us consider a so called *splitting scheme without projection* which has a full rank. Then, a natural extension of the splitting scheme used for ODEs would read as follows. Let $\Delta t > 0$ and for all $n \in \mathbb{N}$, $t_n := \Delta t$. Let $(G_n)_{n\in\mathbb{N}}$ be a sequence of independent identically distributed random variables with normal law. Then, for all $n \in \mathbb{N}$, compute

1) $Y_1^n := Y^n + \Delta t B(t_n, Y^n) + \sqrt{\Delta t}\Sigma G_n$;

2) $Y_2^n = Y_1^n - \Delta t B(t_n, Y_1^n) - \sqrt{\Delta t}\Sigma G_n$;

3) $Y^{n+1} = Y_2^n + \Delta t B(t_n, Y_2^n) + \sqrt{\Delta t}\Sigma G_n$.

It can be checked that this splitting scheme is consistent up to order 1 in Δt with the Euler-Maruyama discretization scheme. This is the reason why we consider a rank-truncated version of this scheme in the case of an additive noise in Section III.3.2 below.

Remark III.3.1. *Note that a naive extension of this splitting scheme too a multiplicative noise is not consistent with an Euler-Maruyama scheme. Indeed, consider the following splitting scheme: for all $n \in \mathbb{N}$,*

1) $Y_1^n = Y^n + B(t_n, Y^n)\Delta t + \Sigma(t_n, Y^n)\sqrt{\Delta t}G_n$

2) $Y_2^n = Y_1^n - B(t_n, Y_1^n)\Delta t - \Sigma(t_n, Y_1^n)\sqrt{\Delta t}G_n$

3) $Y_{n+1} = Y_2^n + B(t_n, Y_2^n)\Delta t + \Sigma(t_n, Y_2^n)\sqrt{\Delta t}G_n$

It then holds that

$$\begin{aligned}
\Sigma(t_n, Y_1^n) &= \Sigma(t_n, Y^n) + \sqrt{\Delta t}\nabla_X\Sigma(t_n, Y^n)\Sigma(t_n, Y^n)G_n \\
&\quad + \Delta t\left(\nabla_X\Sigma(Y^n)b(Y^n) + \frac{1}{2}\langle\langle\nabla_X^2\Sigma(Y^n, t_n), \Sigma(t_n, Y^n)\rangle\Sigma(t_n, Y^n)G_n^2\right) + o(\Delta t) \\
B(t_n, Y_1^n) &= B(t_n, Y^n) + \sqrt{\Delta t}\nabla_X B(t_n, Y^n)\Sigma(t_n, Y^n)G_n \\
&\quad + \Delta t\left(\nabla_X B(t_n, Y^n)B(t_n, Y^n) + \frac{1}{2}\langle\nabla_X^2 B(t_n, Y^n), \Sigma(t_n, Y^n)\rangle\Sigma(t_n, Y^n)G_n^2\right) + o(\Delta t)
\end{aligned}$$

This implies that

$$\begin{aligned}
Y_2^n &= Y^n + \sqrt{\Delta t}\left[\Sigma(t_n, Y^n)G_n - \Sigma(t_n, Y^n)G_n\right] \\
&\quad + \Delta t\left[B(t_n, Y^n) - B(t_n, Y^n) - \nabla_X\Sigma(t_n, Y^n)\Sigma(t_n, Y^n)G_n^2\right] + o(\Delta t) \\
&= Y^n - \Delta t\nabla_X\Sigma(t_n, Y^n)\Sigma(t_n, Y^n)G_n^2 + o(\Delta t).
\end{aligned}$$

We then obtain

$$\Sigma(t_n, Y_2^n) = \Sigma(t_n, Y^n) + \Delta t \nabla_X \sigma(t_n, Y^n)^2 \Sigma(t_n, Y^n) G_n^2 + o(\Delta t),$$
$$B(t_n, Y_2^n) = B(t_n, Y^n) + \Delta t \nabla_X (t_n, Y^n) \nabla_X \Sigma(t_n, Y^n) \Sigma(t_n, Y^n) G_n^2 + o(\Delta t).$$

As a consequence, it holds that

$$Y^{n+1} = Y^n + \sqrt{\Delta t}\,[\Sigma(t_n, Y^n) G_n] + \Delta t\left[-\nabla_X \Sigma(t_n, Y^n)\Sigma(t_n, Y^n) G_n^2 + B(t_n, Y^n)\right] + o(\Delta t). \tag{III.17}$$

Since the term $-\nabla_X \Sigma(t_n, Y^n)\Sigma(t_n, Y^n) G_n^2$ may not be zero in general, we clearly see from (III.17) that the splitting scheme written above cannot be consistent of order 1 with an Euler-Maruyama scheme. This is however the case when Σ is a constant function. One way to fix this issue would be to consider, for instance, the following corrected splitting scheme:

1) $Y_1^n = Y^n + [B(t_n, Y^n) + \nabla_X \Sigma(t_n, Y^n)\Sigma(t_n, Y^n) G_n^2]\,\Delta t + \Sigma(t_n, Y^n)\sqrt{\Delta t} G_n$

2) $Y_2^n = Y_1^n - [B(t_n, Y_1^n) + \nabla_X \Sigma(t_n, Y_1^n)\Sigma(t_n, Y_1^n) G_n^2]\,\Delta t - \Sigma(t_n, Y_1^n)\sqrt{\Delta t} G_n$

3) $Y^{n+1} = Y_2^n + [B(t_n, Y_2^n) + \nabla_X \Sigma(t_n, Y_2^n)\Sigma(t_n, Y_2^n) G_n^2]\,\Delta t + \Sigma(t_n, Y_2^n)\sqrt{\Delta t} G_n.$

The study of such corrected consistent splitting schemes together with rank truncation for a dynamical low-rank approximation of the solution of parametric SDEs will be discussed in section III.5, and we restrict our presentation for the moment to the case of a additive noise.

III.3.2 A fixed-rank splitting scheme

In this section, we present the rank-truncated splitting scheme we implemented in our numerical tests. The objective is actually to get an ensemble of realizations of the solution in order to be able to run a Monte Carlo method to get empirical averages.

We assume that d random realizations of the noise are considered. More precisely, let $(G_n^j)_{1\leq j\leq d, n\in\mathbb{N}}$ be a family of independent and identically distributed normal random variables.

Let us introduce a few notation. For all $1 \leq j \leq d$, we denote by $(\overline{X}_j^n)_{n\in\mathbb{N}}$ the approximation of the random process X_t obtained via an Euler-Maruyama scheme (III.16) with $G_n = G_n^j$ for all $n \in \mathbb{N}$. More precisely, for all $n \in \mathbb{N}$ and all $1 \leq j \leq d$,

$$\overline{X}_j^{n+1} = \overline{X}_j^n + \Delta t B(t_n, \overline{X}_j^n) + \sqrt{\Delta t}\Sigma G_n^j, \tag{III.18}$$

and for all $n \in \mathbb{N}$, we denote by $\overline{X}^n := \left(\overline{X}_1^n, \cdots, \overline{X}_d^n\right)^T \in \mathbb{R}^{d\times p}$.

For all $\overline{X} = (X_1, \cdots, X_d) \in \mathbb{R}^{d\times p}$, we denote by $\overline{B}(t, \overline{X}) := (B(t, X_1), \cdots, B(t, X_d)) \in \mathbb{R}^{d\times p}$. In addition, we denote by $\overline{G}_n := (G_n^1, \cdots, G_n^d) \in \mathbb{R}^d$. Then, (III.18) can be rewritten in the more compact form

$$\overline{X}^{n+1} = \overline{X}^n + \Delta t \overline{B}(t_n, \overline{X}^n) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma. \tag{III.19}$$

For all $1 \leq j \leq d$, we denote by $(\overline{Y}_j^{SP,n})_{n\in\mathbb{N}}$ the approximation of the random process X_t obtained via a full rank splitting see section (III.3.1) with $G_n = G_n^j$ for all $n \in \mathbb{N}$. Denoting by $\overline{Y}^{SP,n} := \left(\overline{Y}_1^{SP,n}, \cdots, \overline{Y}_d^{SP,n}\right) \in \mathbb{R}^{d\times p}$, the full rank splitting scheme can be rewritten in the compact form:

1) $\overline{Y}_1^{SP,n} := \overline{Y}^{SP,n} + \Delta t \overline{B}(t_n, \overline{Y}^n) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma$;

2) $\overline{Y}_2^{SP,n} = \overline{Y}_1^{SP,n} - \Delta t \overline{B}(t_n, \overline{Y}_1^{SP,n}) - \sqrt{\Delta t}\overline{G}_n \otimes \Sigma$;

3) $\overline{Y}^{SP,n+1} = Y_2^{SP,n} + \Delta t \overline{B}(t_n, \overline{Y}_2^{SP,n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma$.

Let now $r \in \mathbb{N}^*$ such that $r \leq \min(p, d)$. We denote by $\overline{Y}_r^{DO,n}$ the approximation of $\overline{Y}^{SP,n}$ given by the rank-r truncated dynamical orthogonal splitting scheme that we now introduce. Assuming that $\overline{Y}_r^{DO,n} = \overline{U}^n\overline{S}^n(\overline{V}^n)^T$ with $\overline{U}^n \in \mathbb{R}^{d\times r}$, $\overline{V}^n \in \mathbb{R}^{p\times r}$ and $\overline{S}^n \in \mathbb{R}^{r\times r}$, we obtain $\overline{Y}_r^{DO,n+1}$ as defined in algorithm (6).

Algorithm 6 Projector Splitting algorithm

Input: Let $r \in \mathbb{N}^*$, $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let us be given at the step $n = 0$, the approximation $\overline{Y}_r^{DO,0} = \overline{U}^0\overline{S}^0(\overline{V}^0)^T$.

Output: $(\overline{Y}_r^{DO,n})_{0\leq n\leq N}$

While $0 \leq n \leq N-1$ do,

- **1)** $\overline{Y}_1^{DO,n} := \overline{Y}_r^{DO,n} + \left[\Delta t \overline{B}(t_n, \overline{Y}^{DO,n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right] P_{\overline{V}^n}$; compute $\overline{Y}_1^{DO,n} = \overline{U}^{n+1}\overline{S}_1^n(\overline{V}^n)^T$ with $\overline{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\overline{S}_1^n \in \mathbb{R}^{r\times r}$;
- **2)** $\overline{Y}_2^{DO,n} = \overline{Y}_1^{DO,n} + P_{\overline{U}^{n+1}}\left[-\Delta t \overline{B}(t_n, \overline{Y}_1^{DO,n}) - \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right] P_{\overline{V}^n}$; compute $\overline{Y}_2^{DO,n} = \overline{U}^{n+1}\overline{S}_2^n(\overline{V}^n)^T$ with $\overline{S}_2^n \in \mathbb{R}^{r\times r}$;
- **3)** $\overline{Y}^{DO,n+1} = \overline{Y}_2^{DO,n} + P_{\overline{U}^{n+1}}\left[\Delta t \overline{B}(t_n, \overline{Y}_2^{DO,n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right]$; compute $\overline{Y}_r^{DO,n+1} = \overline{U}^{n+1}\overline{S}^{n+1}(\overline{V}^{n+1})^T$ with $\overline{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\overline{S}^{n+1} \in \mathbb{R}^{r\times r}$.
  $n = n + 1$.

III.3.3 An adaptive-rank splitting scheme

We introduce in this section a variant of Algorithm (6) where the value of the rank-truncation r is allowed to evolve with respect to time. This rank-adaptative splitting scheme is analogous to the scheme proposed by Ceruti, Kush and Lubich in [8].

Let $\zeta > 0$. We denote by $\overline{Y}_\zeta^{ADO,n}$ the approximation of $\overline{Y}^{SP,n}$ given by the adaptive truncated dynamical orthogonal splitting scheme with an error tolerance ζ. Assuming that $\overline{Y}_\zeta^{ADO,n} = \overline{U}^n\overline{S}^n(\overline{V}^n)^T$ with $\overline{U}^n \in \mathcal{V}_{d,r_n}$, $\overline{V}^n \in \mathcal{V}_{p,r_n}$ and $\overline{S}^n \in \mathbb{R}^{r_n\times r_n}$ for some $r_n \in \mathbb{N}^*$, $\overline{Y}_\zeta^{ADO,n+1}$ is computed as defined in algorithm (7).

As seen from theorem (III.2.3), an over-ranking approximation by the Projector-Splitting method does not affect the approximation error, but in the case where the matrix $X(t)$ has an increasing rank over the time, the approximation with fixed rank will lose its accuracy. This problem is solved using an adaptative rank to the splitting approximation as shown by Ceruti, Kush and Lubich in [8]. The autors present an algorithm, Adaptative-Splitting, that allows rank adaptation through iterations. It is also proved that this algorithm presents the same properties as the Projector-Splitting. Hence, we adapt this for the SDE (see algorithm (7)).

Algorithm 7 Adaptive Projector Splitting algorithm

Input: Let $r \in \mathbb{N}^*$, $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let the tolerance $\zeta > 0$. Let us be given at the step $n = 0$, the approximation $\overline{Y}_\zeta^{ADO,0}$ and $r_0 = r$.

Output: $(\overline{Y}_\zeta^{ADO,n})_{0\leq n\leq N}$ and r_n for $0 \leq n \leq N$.

While $0 \leq n \leq N-1$ do,

- **1)** $\overline{Y}_1^{ADO,n} := \overline{Y}_\zeta^{ADO,n} + \left[\Delta t \overline{B}(t_n, \overline{Y}_\zeta^{ADO,n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right] P_{\overline{V}^n}$; compute $\overline{Y}_1^{ADO,n} = \overline{U}_1^n \overline{S}_1^n (\overline{V}^n)^T$ with $\overline{U}_1^n \in \mathcal{V}_{d,r_n}$ and $\overline{S}_1^n \in \mathbb{R}^{r_n \times r_n}$; compute $\widetilde{U}_1^n \in \mathcal{V}_{d,2r_n}$ obtained from a QR decomposition of the matrix $(\overline{U}_1^n \overline{S}_1^n, \overline{U}^n) \in \mathbb{R}^{d\times 2r_n}$; define $\overline{M} := \left(\widetilde{U}_1^n\right)^T \overline{U}^n \in \mathbb{R}^{2r_n \times r_n}$.

- **2)** $\overline{Y}_2^{ADO,n} = \overline{Y}_\zeta^{ADO,n} + P_{\overline{U}^n}\left[\Delta t \overline{B}(t_n, \overline{Y}_\zeta^{ADO,n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right]$; compute $\overline{Y}_2^{ADO,n} = \overline{U}^n \overline{S}_2^n (\overline{V}_2^n)^T$ with $\overline{V}_2^n \in \mathcal{V}_{d,r_n}$ and $\overline{S}_2^n \in \mathbb{R}^{r_n \times r_n}$; compute $\widetilde{V}_2^n \in \mathcal{V}_{p,2r_n}$ obtained from a QR decomposition of the matrix $(\overline{V}_2^n (\overline{S}_2^n)^T, \overline{V}^n) \in \mathbb{R}^{p\times 2r_n}$; define $\overline{N} := \left(\widetilde{V}_2^n\right)^T \overline{V}^n \in \mathbb{R}^{2r_n \times r_n}$;

- **3)** Let $\widetilde{S}^n := \overline{MS}^n (\overline{N})^T$ and let $\widetilde{S}_3^n := \widetilde{S}^n + \left(\widetilde{U}_1^n\right)^T \left[\Delta t \overline{B}(t_n, \overline{Y}_\zeta^{ADO,n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right] \widetilde{V}_2^n \in \mathbb{R}^{2r_n \times 2r_n}$;

- **4)** Compute a rank r_{n+1}-truncated SVD decomposition of $\widetilde{S}_3^n$ of the form $\overline{PS}^{n+1}(\overline{Q})^T$ with $\overline{P} \in \mathcal{V}_{2r_n, r_{n+1}}$, $\overline{Q} \in \mathcal{V}_{2r_n, r_{n+1}}$ and $\overline{S}^{n+1} \in \mathbb{R}^{r_{n+1}\times r_{n+1}}$. The rank r_{n+1} is chosen so that $r_{n+1} \geq r_n$ and
$$\|\widetilde{S}_3^n - \overline{PS}^{n+1}(\overline{Q})^T\|_F^2 \leq \zeta,$$
for some error tolerance $\zeta > 0$. Then, compute $\overline{U}^{n+1} := \widetilde{U}_1^n \overline{P} \in \mathcal{V}_{d,r_{n+1}}$, $\overline{V}^{n+1} := \widetilde{V}_2^n \overline{Q} \in \mathcal{V}_{p,r_{n+1}}$ and $\overline{Y}_\zeta^{ADO,n+1} := \overline{U}^{n+1}\overline{S}^{n+1}\left(\overline{V}^{n+1}\right)$.
$n = n+1$ and $r_n = r_{n+1}$.

The Adaptative-Splitting for the SDE reads as follow, for each iteration n, given $\overline{Y}_\zeta^{ADO,n}$ at time t_n we look for $\overline{Y}_\zeta^{ADO,n+1}$ solution of algorithm (7) at the time step $t_{n+1} = t_n + \Delta t$. The first sub-step of this algorithm is the same as the Projector-Splitting algorithm (6), we just look for $\tilde{U}_1^n \in \mathbb{R}^{d\times 2r_n}$ as an extension of $\overline{U}^n \in \mathbb{R}^{d\times r_n}$, hence we obtain a new vector space of dimension $2r_n$ as a subspace of $\mathbb{R}^d$, this is done also for the deterministic modes in step 2, we look for a new vector space of dimension $2r_n$ as a subspace of $\mathbb{R}^p$. Then we assume that the evolution from the point $\overline{Y}_\zeta^{ADO,n}$ to $\overline{Y}_\zeta^{ADO,n+1}$ is done on these vector spaces. Thus we look for $\tilde{S}^n$ as the

Galerkin projection of the point $\overline{U}^n\overline{S}^n\left(\overline{V}^n\right)$ on these spaces, which is described by the substep 3. Then we make an SVD truncation with rank r_{n+1} to $\tilde{S}_3^n$ where r_{n+1} is defined by a chosen condition. Finally we update the modes by coming back on the vector space of $\tilde{U}^n$ and $\tilde{V}^n$.

III.4 Numerical tests for the additive noise

We present in this section some numerical results obtained with the splitting schemes presented in the previous section in order to compute a dynamical low-rank approximation of the solution of a parametrized stochastic differential equation.

III.4.1 Overdamped Langevin process and initialization

For our numerical experiments, we chose to compute a dynamical low-rank approximation for a parametrized overdamped langevin process, which reads as the solution of

$$dX_t^\mu = -\nabla V^\mu(X_t^\mu)dt + \sqrt{2\beta^{-1}}dW_t \tag{III.20}$$

where for all μ in the set of parameter values $\mathcal{P}$, $V^\mu : \mathbb{R} \to \mathbb{R}$ is a smooth parametrized potential function, and where $\beta > 0$. The overdamped Langevin dynamics is often used in molecular dynamics simulations in order to compute statistical averages of observables of interest related to some molecular systems.

In our numerical experiments, the set of parameter values $\mathcal{P}$ is chosen to be $\mathcal{P} := [0.1, 1] \times [0.5, 5]$ and we consider V^μ to be given as the so-called double-well potential defined as

$$\forall x \in \mathbb{R},\ \forall \mu = (a, b) \in \mathcal{P}, \qquad V^\mu(x) := a\left(\left(\frac{x}{b}\right)^2 - 1\right)^2.$$

For a given value $p_1, p_2 \in \mathbb{N}^*$, the train parameter set $\mathcal{P}_{\text{train}}$ is chosen as the cartesian product of two sets $\{a_1, \cdots, a_{p_1}\} \times \{b_1, \cdots, b_{p_2}\}$, where $(a_i)_{1\leq i\leq p_1}$ (respectively $(b_i)_{1\leq i\leq p_2}$) are uniformly distributed in $[0.1, 1]$ (respectively $[0.5, 5]$). As a consequence, the cardinality p of $\mathcal{P}_{\text{train}}$ is equal to $p := p_1 p_2$.

III.4.2 Initialization step

The DO scheme is initialized in the following way. A full Euler-Maruyama scheme (III.19) is computed during a time $t_0 > 0$ so that $t_0 = n_0\Delta t$ from an initial condition

$$\overline{X}_0 := 0.$$

Given $r \in \mathbb{N}$, the DO low-rank scheme is then initialized by choosing $\overline{Y}^{DO,n_0} = \overline{X}_r^{n_0}$ where $\overline{X}_r^{n_0}$ is a rank-r truncated SVD decomposition of $\overline{X}^{n_0}$.

The different splitting algorithms detailed in the previous section are then run from the starting time t_0 up to a final time $T = N\Delta t$ for some $N \in \mathbb{N}^*$.

III.4.3 Low-rank approximation of the solution of the full-rank splitting scheme

In this section, numerical parameters are chosen as follows: $p_1 = 25$, $p_2 = 20$ (so that $p = 500$), $d = 800$, $\beta = 1$, $\Delta t = 1e-3$, $t_0 = 3$ and $T = 10$.

The aim of this section is to illustrate the low-rank approximability properties of the solution $(\overline{Y}_{\overline{r}}^{DO,n})_{0\leq n\leq N}$, where $\overline{r} = \min(p, d)$ computed by a maximal-rank splitting scheme. For all $1 \leq i \leq \overline{r} = 500$, we denote by λ_i^n the i^{th} singular value of the matrix $\overline{Y}_{\overline{r}}^{DO,n}$. We then define for all $1 \leq i \leq \overline{r}$,

$$\lambda_i^{\min} := \min_{0\leq n\leq N} \lambda_i^n \quad \text{and } \lambda_i^{\max} := \max_{0\leq n\leq N} \lambda_i^n.$$

For all $1 \leq r \leq \overline{r} = 500$ and $0 \leq n \leq N$, we denote by $\overline{Y}_r^{SVD,n}$ a rank-r truncated SVD decomposition of $\overline{Y}_{\overline{r}}^{DO,n}$ and by

$$e_n(r) := \left\| \overline{Y}_{\overline{r}}^{DO,n} - \overline{Y}_r^{SVD,n} \right\|_F.$$

We also introduce the quantities,

$$e_{\min}(r) := \min_{0\leq n\leq N} e_n(r) \quad \text{and } e_{\max}(r) := \max_{0\leq n\leq N} e_n(r).$$

In Figure III.1 are plotted $\lambda_i^{\min}$ and $\lambda_i^{\max}$ as a function of i on the left-hand side, it presents the decrease of the singular values between t_0 and T. On the right-hand side, are plotted $e_{\min}(r)$ and $e_{\max}(r)$ as a function of r. We remark from both figures that a low rank solution of the problem (III.20) may exist as we have a decrease of the singular values at each time step with a best error that is at least between 10^{-2} and 10^{-6} starting from the rank $r = 50$.

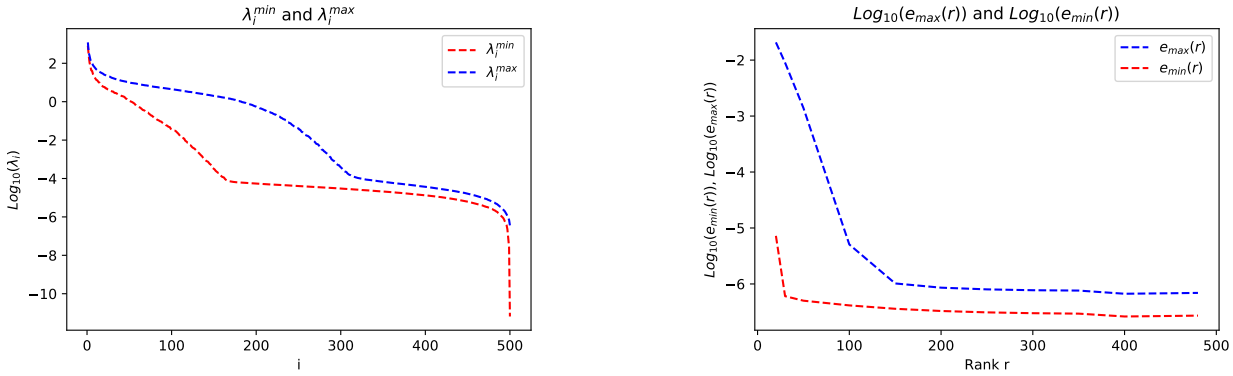


Figure III.1: Left: $\lambda_i^{\min}$ and $\lambda_i^{\max}$ as a function of i. Right: $e_{\min}(r)$ and $e_{\max}(r)$ as a function of r.

III.4.4 Splitting scheme for the DO method

The aim of this section is to illustrate the approximability properties of the DO method, used in conjunction with the splitting scheme described in the preceding sections.

Here, the numerical parameters are identical to those of the preceding section. For all $n \in \mathbb{N}$ and $1 \leq r \leq \overline{r} = 500$, we define by

$$\epsilon_n(r) := \left\| \overline{Y}_{\overline{r}}^{DO,n} - \overline{Y}_{r}^{DO,n} \right\|_F .$$

On the left-hand side of Figure III.2 are plotted the quantities $e_n(r)$ for $r = 50$ (red curve) and $r = 150$ (blue curve) and $\epsilon_n(r)$ for $r = 50$ (magenta curve) and $r = 150$ (black curve) as a function of the time t_n.

On the right-hand side of Figure III.2, we plot a realisation of the trajectory of the stochastic process X_t^μ for $\mu = (1.58, 1.33)$ computed with an Euler-Maruyama scheme (blue curve) and its approximation computed by a DO splitting scheme with rank $r = 50$. We remark in this plot that the transition of the stochastic process from one well to another of the double-well potential is well-recovered by the DO approximation.

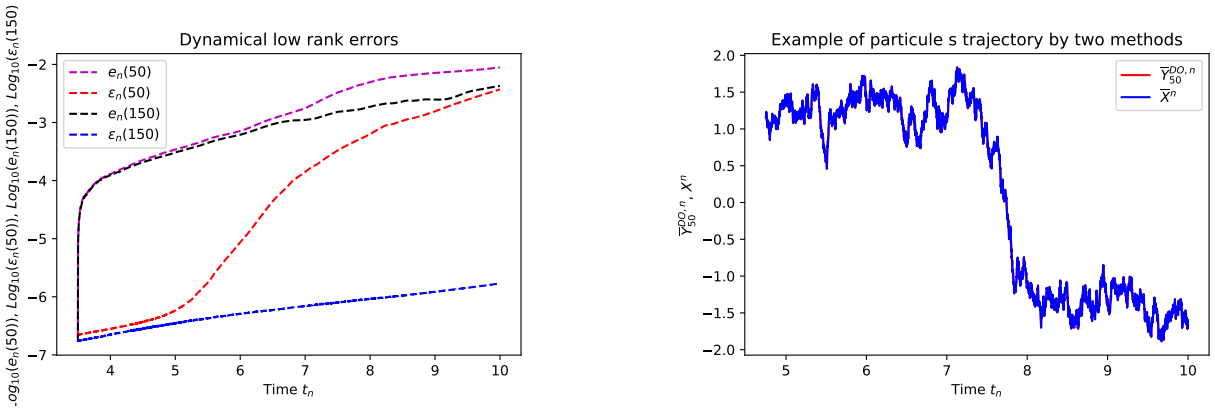


Figure III.2: Left: $e_n(50)$, $e_n(150)$, $\epsilon_n(50)$ and $\epsilon_n(150)$ as a function of t_n. Right: Particular realisation of the stochastic process for $\mu = (1.58, 1.33)$ computed with the Euler-Maruyama scheme (blue curve) or the DO approximation with $r = 50$ (red curve).

Let us define

$$\epsilon_{\max}(r) := \max_{0 \leq n \leq N} \epsilon_n(r).$$

In Figure III.3, are plotted the quantities $\epsilon_{\max}(r)$ and $e_{\max}(r)$ as a function of r. The result goes along with what we obtained in Figure III.2, we have the same order of decrease of the error between the projector splitting approximation and the SVD approximation for rank r less than $r_0 = 70$. Then when r is higher than r_0 we remark that the projector splitting method error remains constant and we lose the match with the best approximation error. This is observed for many test cases, we present only one of them.

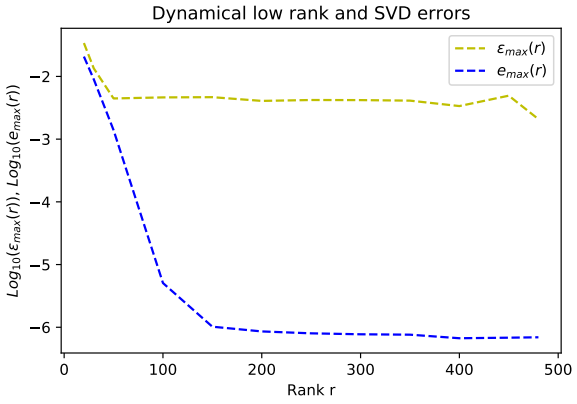


Figure III.3: $\epsilon_{\max}(r)$ and $e_{\max}(r)$ as a function of r.

We also consider here the behaviour of the error in another norm, which we refer hereafter as the trajectorial error, defined by

$$\eta_1(r) = \frac{1}{dp}\sum_{i=1}^{d}\sum_{j=1}^{p}\left[\sup_{0\leq n\leq N}\left|\left(\overline{Y}_{\overline{r}}^{DO,n}\right)_{ij} - \left(\overline{Y}_{r}^{DO,n}\right)_{ij}\right|^2\right]$$

And

$$\eta_2(r) = \frac{1}{dp}\sum_{i=1}^{d}\sum_{j=1}^{p}\left[\sup_{0\leq n\leq N}\left|\left(\overline{Y}_{\overline{r}}^{DO,n}\right)_{ij} - \left(\overline{Y}_{r}^{SVD,n}\right)_{ij}\right|^2\right]$$

In Figure III.4, we plot $\eta(r)$ as a function of r when $N \in \mathbb{N}$ is chosen so that $3 \leq t_n \leq 5$. We remark that the L^∞ in time error between trajectories of the full and reduced-order model are quite small, which is a very good feature of the method.

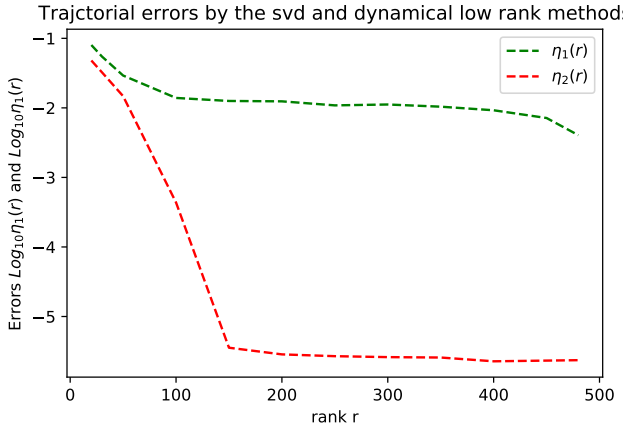


Figure III.4: Evolution of the trajectorial error $\eta(r)$ as a function of the rank r.

III.4.5 Influence of the time step

In Figure III.5, are plotted six curves corresponding to $\log_{10}\epsilon_{\max}(r)$ as a function of r, for different values of the time step Δt, namely $\Delta t = 10^{-2}$, $\Delta t = 2.10^{-3}$, $\Delta t = 4.10^{-4}$, $\Delta t = 2.10^{-4}$,

$\Delta t = 4.10^{-5}$ and $\Delta t = 10^{-5}$.

We numerically observe that, the smaller the time step Δt, the more accurate is the reduced-order model with respect to the full-order model.

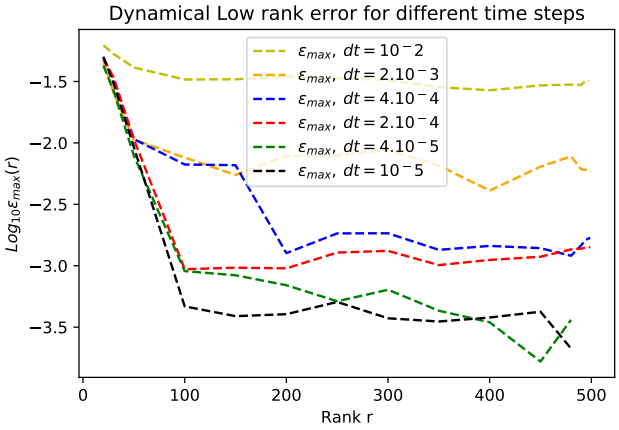


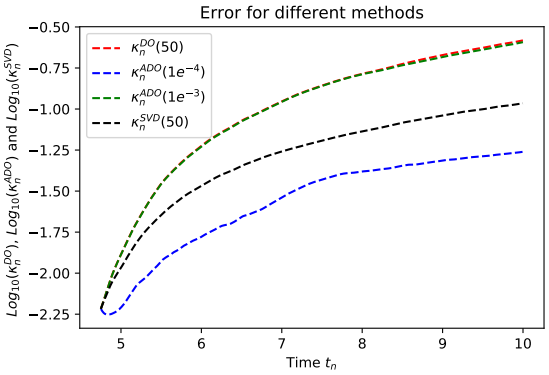
Figure III.5: Evolution of $\log_{10} \epsilon_{\max}(r)$ as a function the rank r for different time step Δt.

III.4.6 Comparison between different schemes

In this section, we compare different time integrators for the resolution of the DO equations, namely the splitting and adaptive splitting schemes. To this aim, we plot the errors produced by the different schemes, with respect to the solution $\left(\overline{X}^n\right)_{0\leq n\leq N}$ given by the standard Euler-Maruyama scheme. For all $0 \leq n \leq N$, we thus denote by

$$\kappa_n^{SVD}(r) := \left\|\overline{X}^n - \overline{X}_r^n\right\|_F, \quad \kappa_n^{DO}(r) := \left\|\overline{X}^n - \overline{Y}_r^{DO,n}\right\|_F \quad \text{and} \quad \kappa_n^{ADO}(\zeta) := \left\|\overline{X}^n - \overline{Y}_\zeta^{ADO,n}\right\|_F,$$

where $\overline{X}_r^n$ denotes a rank-r truncated SVD decomposition of $\overline{X}^n$.



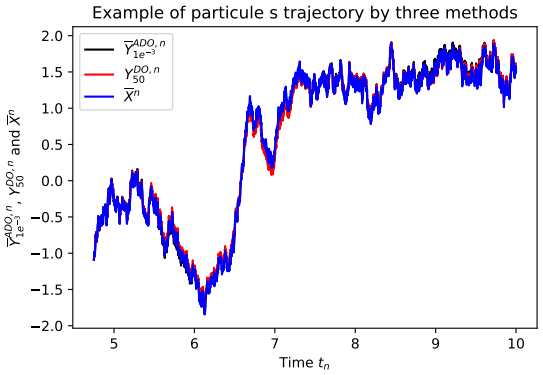


Figure III.6: Left: $\log_{10} \kappa_n^{SVD}(50)$, $\log_{10} \kappa_n^{DO}(50)$, $\log_{10} \kappa_n^{ADO}(1e-3)$, $\log_{10} \kappa_n^{ADO}(1e-4)$ as a function of t_n. Right: Comparison of some stochastic trajectories for $\mu = (1.15, 2.02)$ as a function of time: $\overline{X}^n$, $\overline{Y}_{50}^{DO,n}$ and $\overline{Y}_{1e^{-3}}^{ADO,n}$.

In figure III.6, are plotted four curves. The green curve represents the error obtained with $\overline{Y}^{ADO,n}_{1e^{-3}}$, this approximation have shown a conserved rank $r_0 = r_N = 50$ during the simulation, while the blue curve represents the error obtained with $\overline{Y}^{ADO,n}_{1e^{-4}}$ that have shown a rank evolving from $r_0 = 50$ to $r_N = 194$. The black curve represents the error obtained with the best approximation with rank $r = 50$, $\overline{X}^n_r$, and the red curve represents the error obtained with the dynamical orthogonal scheme using a rank $r = 50$, $\overline{Y}^{DO,n}_{50}$. We observe that the error between $\overline{X}^n$ and any of the two DO solutions is quite small. First, we remark that we have quite the same order of approximation between the best approximation of rank r, $\overline{X}^n_r$, and the dynamical orthogonal approximation of rank r along the interval $[t_0, T]$, thus the dynamical orthogonal approximation is comparable to the best approximation for $\overline{X}^n$. Second, obviously the adaptive method shows better approximation as the rank r increases, but remark when the adaptive method does not increase the rank r then $\overline{Y}^{ADO,n}_{\zeta} = \overline{Y}^{DO}_{r}$ for all $0 \leq n \leq N$, which implies coherence between methods.
The right plot on Figure III.6 gives an example of a trajectories obtained by either the Euler-Maruyama scheme $\overline{X}^n$, the Projector-Splitting $\overline{Y}^{DO,n}_r$ or the Adaptive-Projector-Splitting $\overline{Y}^{ADO,n}_{\zeta}$. We can observe the good matching between methods especially the ability to catch the transition of the particule from a well to another.

Comparaison of the stochastic and deterministic modes of different schemes

Let us assume that for all $r \in \mathbb{N}^*$ and $\zeta > 0$, we have

$$\overline{Y}^{DO,n}_r = U^{DO,n}_r S^{DO,n}_r \left(V^{DO,n}_r\right)^T, \quad \overline{Y}^{ADO,n}_{\zeta} = U^{ADO,n}_{\zeta} S^{ADO,n}_{\zeta} \left(V^{ADO,n}_{\zeta}\right)^T$$

and

$$\overline{X}^n_r = U^{SVD,n}_r S^{SVD,n}_r \left(V^{SVD,n}_r\right)^T,$$

where the matrices $U^{DO,n}_r$,$S^{DO,n}_r$,$V^{DO,n}_r$ (respectively $U^{ADO,n}_{\zeta}$,$S^{ADO,n}_{\zeta}$,$V^{ADO,n}_{\zeta}$) are obtained with the projector splitting (respectively the adaptive projector splitting) algorithm and $U^{SVD,n}_r \in \mathcal{V}_{d,r}$, $S^{SVD,n}_r \in \mathbb{R}^{r\times r}$ and $V^{SVD,n}_r \in \mathcal{V}_{p,r}$ are obtained via a truncated SVD-decomposition of $\overline{X}^n$.

In the numerical results highlighted below, the projector splitting algorithm was run with $r = 50$ and the adaptive projector splitting with $\zeta = 1e^{-4}$.

In Figure III.8, we plot values of $\left(V^n_{ij}\right)_{0\leq n\leq N}$ as a function of the time t_n for different methods for some $1 \leq i \leq p$, corresponding to a parameter value $\mu_i = (1.25, 1.8)$ and for different values of $j = 1, 5, 10, 20$. The higher the value of j the smaller the associated singular value in the SVD decomposition of the matrix $\overline{X}^n$. We thus plot the evolution of the parametric modes of the DO decomposition for the particular value of the parameter $\mu_i = (1.25, 1.8)$.

We also plot in Figure III.7 values of $\left(U^n_{ij}\right)_{0\leq n\leq N}$ as a function of the time t_n for different methods for some $1 \leq i \leq d$, and for different values of $j = 1, 5, 10, 20$. The higher the value of j the smaller the associated singular value in the SVD decomposition of the matrix $\overline{X}^n$. We thus plot the evolution of one particular random realisation of the stochastic modes of the DO decomposition.

We observe that these modes are very close to one another for small values of j and differ when j gets larger.

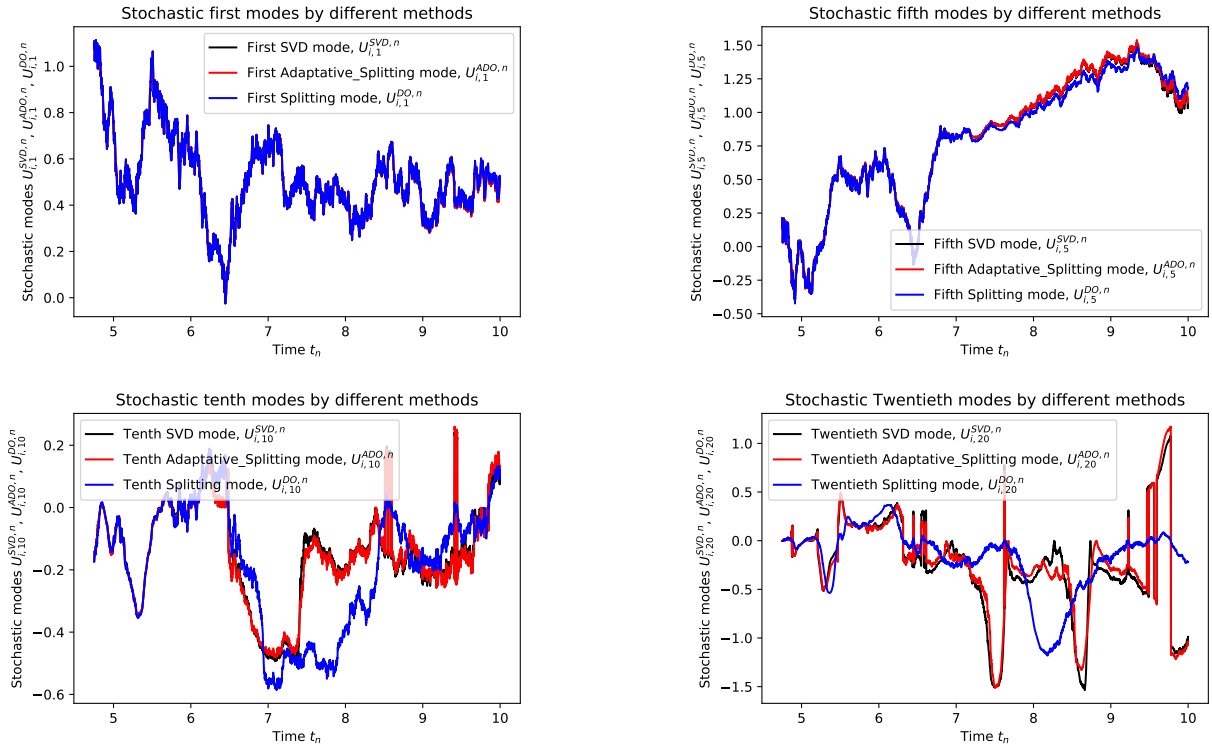


Figure III.7: Stochastic Modes 1,5, 10 and 20 for different methods

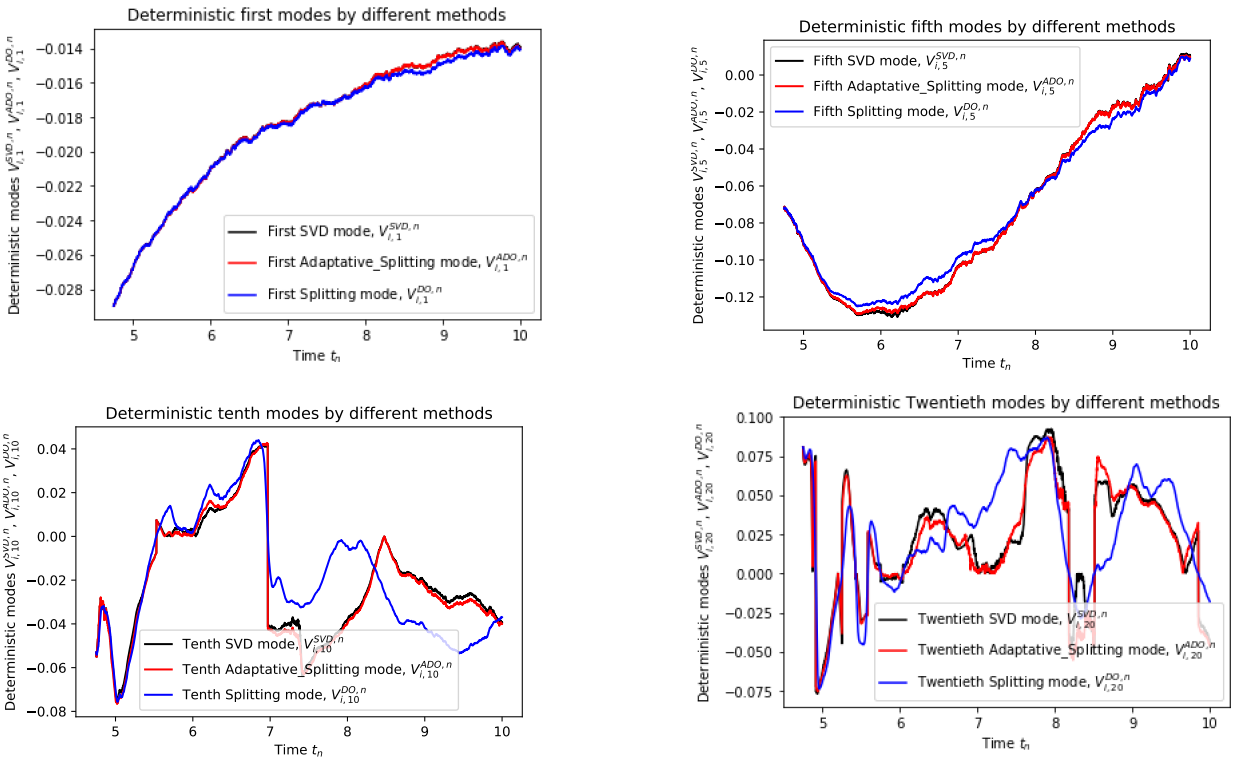


Figure III.8: Deterministic Modes 1,5, 10 and 20 for different methods

III.5 Generalization to multiplicative noise and to McKean nonlinearity

III.5.1 An SDE with multiplicative noise

In this part we present 4 schemes to solve the problem (III.14) in the multiplicative noise case. We prove that these schemes are consistent with the initial EDS at the order 1. We first prove numerically that these schemes converge to the Euler Maruyama scheme in the full rank case, and then we compare their efficiencies a function of the rank r. The numerical tests are done on the following overdamped Langevin equation,

$$dX_t^\mu = -\nabla V^\mu(X_t^\mu)dt + \sqrt{2\beta^{-1}}X_t^\mu dW_t. \tag{III.21}$$

We use the same notations as in (III.18), then, (III.21) can be rewritten using the Hadamard product $\overline{\otimes}$,

$$\overline{X}^{n+1} = \overline{X}^n + \Delta t\overline{B}(t_n, \overline{X}^n) + \sqrt{\Delta t}\overline{\overline{G}}_n\overline{\otimes}\overline{\Sigma}(t_n, \overline{X}^n), \tag{III.22}$$

such that,

$$\overline{\Sigma}(t, \overline{X}^n) = \sqrt{2\beta^{-1}}\overline{X}^n \in \mathbb{R}^{d,p} \qquad \text{and} \qquad \overline{\overline{G}}_n = \overline{G}_n \otimes L^T \in \mathbb{R}^{d,p}, \quad \text{where} \quad L = (1, \cdots, 1) \in \mathbb{R}^p.$$

The algorithm (8) below is the Splitting full rank scheme with the correction term $\overline{C}(t_n, \overline{Y}^n)$ such that $\overline{C}(t_n, \overline{Y}^n) = 2\beta^{-1}\overline{Y}^n\overline{\otimes}(\overline{\overline{G}}_n\overline{\otimes}\overline{\overline{G}}_n)$ as proven in (III.17). This scheme is of order one compared to the Euler Maruyama scheme (III.22). We introduce this scheme only because it does not include the depends on r and thus there no projection to do. Hence there is no gain to use this scheme (in term of speedup) we only take it for reference as it is obvious that it converges at the order one to Euler Mauryama scheme. We denote by $\overline{Y}^{SP,n} \in \mathbb{R}^{d\times p}$ the solution obtained via the algorithm (8) as the approximation of the random process $\overline{X}^n$ for all $n \in \mathbb{N}$.

Algorithm 8 Splitting without projection

Input: Let $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let us be given at the step $n = 0$, the approximation $\overline{Y}^{SP,0}$.
Output: $(\overline{Y}^{SP,n})_{0\le n\le N}$.
While $0 \le n \le N-1$ do:

- **1)** compute $\overline{Y}_1^{SP,n} := \overline{Y}^{SP,n} + \Delta t\left[\overline{B}(t_n, \overline{Y}^{SP,n}) + \overline{C}(t_n, \overline{Y}^{SP,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n\overline{\otimes}\overline{\Sigma}(t_n, \overline{Y}^{SP,n})$; compute $\overline{Y}_1^{SP,n} = \overline{U}^{n+1}\overline{S}_1^n(\overline{V}^n)^T$ with $\overline{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\overline{S}_1^n \in \mathbb{R}^{r\times r}$;
- **2)** $\overline{Y}_2^{SP,n} = \overline{Y}_1^{SP,n} - \Delta t\left[\overline{B}(t_n, \overline{Y}_1^{SP,n}) + \overline{C}(t_n, \overline{Y}_1^{SP,n})\right] - \sqrt{\Delta t}\overline{\overline{G}}_n\overline{\otimes}\overline{\Sigma}(t_n, \overline{Y}_1^{SP,n})$; compute $\overline{Y}_2^{SP,n} = \overline{U}^{n+1}\overline{S}_2^n(\overline{V}^n)^T$ with $\overline{S}_2^n \in \mathbb{R}^{r\times r}$;
- **3)** $\overline{Y}^{SP,n+1} = \overline{Y}_2^{SP,n} + \Delta t\left[\overline{B}(t_n, \overline{Y}_2^{SP,n}) + \overline{C}(t_n, \overline{Y}_2^{SP,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n\overline{\otimes}\overline{\Sigma}(t_n, \overline{Y}_2^{SP,n})$; compute $\overline{Y}^{SP,n+1} = \overline{U}^{n+1}\overline{S}^{n+1}(\overline{V}^{n+1})^T$ with $\overline{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\overline{S}^{n+1} \in \mathbb{R}^{r\times r}$.
  $n = n + 1$.

The algorithm (9) that we propose for each rank r is an implicit scheme for both the drift term and the multiplicative noise term, but we introduce the correction term $\overline{C}(t_n, \overline{Y}^n)$ only in the second equation.

Let $r \in \mathbb{N}^*$ such that $r \leq \min(p, d)$. We denote by $\tilde{Y}_r^{DO,n}$ the approximation of $\overline{Y}^{SP,n}$ given by the rank-r truncated dynamical orthogonal splitting algorithm (9) defined below. For each iteration n given $\tilde{Y}_r^{DO,n} = \tilde{U}^n \tilde{S}^n (\tilde{V}^n)^T$ with $\tilde{U}^n \in \mathcal{V}_{d,r}$, $\tilde{V}^n \in \mathcal{V}_{p,r}$ and $\tilde{S}^n \in \mathbb{R}^{r\times r}$, we obtain $\tilde{Y}_r^{DO,n+1}$ by the algorithm (9).

Algorithm 9 Splitting DO using implicit scheme for the drift and the noise

---

Input: Let $r \in \mathbb{N}^*$, $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let us be given at the step $n = 0$, the approximation $\tilde{Y}_r^{DO,0} = \tilde{U}^0 \tilde{S}^0 (\tilde{V}^0)^T$.
Output: $(\tilde{Y}_r^{DO,n})_{0\leq n\leq N}$.
While $0 \leq n \leq N-1$ do:

- **1)** compute $\tilde{Y}_1^{DO,n} := \tilde{Y}_r^{DO,n} + \left[\Delta t \overline{B}(t_n, \tilde{Y}_r^{DO,n}) + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes\Sigma}(t_n, \tilde{Y}_r^{DO,n})\right] P_{\tilde{V}^n}$; compute $\tilde{Y}_1^{DO,n} = \tilde{U}^{n+1} \tilde{S}_1^n (\tilde{V}^n)^T$ with $\tilde{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\tilde{S}_1^n \in \mathbb{R}^{r\times r}$;
- **2)** $\tilde{Y}_2^{DO,n} = \tilde{Y}_1^{DO,n} + P_{\tilde{U}^{n+1}} \left[-\Delta t \left[\overline{B}(t_n, \tilde{Y}_1^{DO,n}) + \overline{C}(t_n, \tilde{Y}_1^{DO,n})\right] - \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes\Sigma}(t_n, \tilde{Y}_1^{DO,n})\right] P_{\tilde{V}^n}$; compute $\tilde{Y}_2^{DO,n} = \tilde{U}^{n+1} \tilde{S}_2^n (\tilde{V}^n)^T$ with $\tilde{S}_2^n \in \mathbb{R}^{r\times r}$;
- **3)** $\tilde{Y}^{DO,n+1} = \tilde{Y}_2^{DO,n} + P_{\tilde{U}^{n+1}} \left[\Delta t \overline{B}(t_n, \tilde{Y}_2^{DO,n}) + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes\Sigma}(t_n, \tilde{Y}_2^{DO,n})\right]$; compute $\tilde{Y}_r^{DO,n+1} = \tilde{U}^{n+1} \tilde{S}^{n+1} (\tilde{V}^{n+1})^T$ with $\tilde{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\tilde{S}^{n+1} \in \mathbb{R}^{r\times r}$.
  $n = n + 1$.

The algorithm (10) is an implicit scheme for the drift term and explicit scheme for the multiplicative noise term, hence we do not introduce any corrective term.

We denote by $\overline{Y}_r^{DO,n}$ the approximation of $\overline{Y}^{SP,n}$ given by the rank-r truncated dynamical orthogonal splitting algorithm (10) defined below. For each iteration n given $\overline{Y}_r^{DO,n} = \overline{U}^n \overline{S}^n (\overline{V}^n)^T$ with $\overline{U}^n \in \mathcal{V}_{d,r}$, $\overline{V}^n \in \mathcal{V}_{p,r}$ and $\overline{S}^n \in \mathbb{R}^{r\times r}$, we obtain $\overline{Y}_r^{DO,n+1}$ by the algorithm (10).

Algorithm 10 Splitting DO using implicit scheme for the drift and explicit scheme for the noise

Input: Let $r \in \mathbb{N}^*$, $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let us be given at the step $n = 0$, the approximation $\overline{Y}_r^{DO,0} = \overline{U}^0 \overline{S}^0 (\overline{V}^0)^T$.
Output: $(\overline{Y}_r^{DO,n})_{0 \leq n \leq N}$.
While $0 \leq n \leq N-1$ do:

- **1)** compute $\overline{Y}_1^{DO,n} := \overline{Y}_r^{DO,n} + \left[\Delta t \overline{B}(t_n, \overline{Y}_r^{DO,n}) + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes} \overline{\Sigma}(t_n, \overline{Y}_r^{DO,n})\right] P_{\overline{V}^n}$; compute $\overline{Y}_1^{DO,n} = \overline{U}^{n+1} \overline{S}_1^n (\overline{V}^n)^T$ with $\overline{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\overline{S}_1^n \in \mathbb{R}^{r \times r}$;
- **2)** $\overline{Y}_2^{DO,n} = \overline{Y}_1^{DO,n} + P_{\overline{U}^{n+1}} \left[-\Delta t \overline{B}(t_n, \overline{Y}_1^{DO,n}) - \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes} \overline{\Sigma}(t_n, \overline{Y}_r^{DO,n})\right] P_{\overline{V}^n}$; compute $\overline{Y}_2^{DO,n} = \overline{U}^{n+1} \overline{S}_2^n (\overline{V}^n)^T$ with $\overline{S}_2^n \in \mathbb{R}^{r \times r}$;
- **3)** $\overline{Y}^{DO,n+1} = \overline{Y}_2^{DO,n} + P_{\overline{U}^{n+1}} \left[\Delta t \overline{B}(t_n, \overline{Y}_2^{DO,n}) + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes} \overline{\Sigma}(t_n, \overline{Y}_r^{DO,n})\right]$; compute $\overline{Y}_r^{DO,n+1} = \overline{U}^{n+1} \overline{S}^{n+1} (\overline{V}^{n+1})^T$ with $\overline{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\overline{S}^{n+1} \in \mathbb{R}^{r \times r}$.
  $n = n + 1$.

The algorithm (11) that we propose is an implicit scheme for the drift term and explicit scheme for the multiplicative noise term that we add only in the first equation.

We denote by $\hat{Y}_r^{DO,n}$ the approximation of $\overline{Y}^{SP,n}$ given by the rank-r truncated dynamical orthogonal splitting algorithm (11) defined below. For each iteration n given $\hat{Y}_r^{DO,n} = \hat{U}^n \hat{S}^n (\hat{V}^n)^T$ with $\hat{U}^n \in \mathcal{V}_{d,r}$, $\hat{V}^n \in \mathcal{V}_{p,r}$ and $\hat{S}^n \in \mathbb{R}^{r \times r}$, we obtain $\hat{Y}_r^{DO,n+1}$ by the algorithm (11).

Algorithm 11 Splitting DO using implicit scheme for the drift and explicit scheme for the noise term added only in the first step

Input: Let $r \in \mathbb{N}^*$, $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let us be given at the step $n = 0$, the approximation $\hat{Y}_r^{DO,0} = \hat{U}^0 \hat{S}^0 (\hat{V}^0)^T$.
Output: $(\hat{Y}_r^{DO,n})_{0 \leq n \leq N}$.
While $0 \leq n \leq N-1$ do:

- **1)** compute $\hat{Y}_1^{DO,n} := \hat{Y}_r^{DO,n} + \left[\Delta t \overline{B}(t_n, \hat{Y}_r^{DO,n}) + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes} \overline{\Sigma}(t_n, \hat{Y}_r^{DO,n})\right] P_{\hat{V}^n}$; compute $\hat{Y}_1^{DO,n} = \hat{U}^{n+1} \hat{S}_1^n (\hat{V}^n)^T$ with $\hat{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\hat{S}_1^n \in \mathbb{R}^{r \times r}$;
- **2)** $\hat{Y}_2^{DO,n} = \hat{Y}_1^{DO,n} + P_{\hat{U}^{n+1}} \left[-\Delta t \overline{B}(t_n, \hat{Y}_1^{DO,n})\right] P_{\hat{V}^n}$; compute $\hat{Y}_2^{DO,n} = \hat{U}^{n+1} \hat{S}_2^n (\hat{V}^n)^T$ with $\hat{S}_2^n \in \mathbb{R}^{r \times r}$;
- **3)** $\hat{Y}^{DO,n+1} = \hat{Y}_2^{DO,n} + P_{\hat{U}^{n+1}} \left[\Delta t \overline{B}(t_n, \hat{Y}_2^{DO,n})\right]$; compute $\hat{Y}_r^{DO,n+1} = \hat{U}^{n+1} \hat{S}^{n+1} (\hat{V}^{n+1})^T$ with $\hat{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\hat{S}^{n+1} \in \mathbb{R}^{r \times r}$.
  $n = n + 1$.

III.5.2 Numerical experiments on the multiplicative noise case

We are interested in this part in comparing the algorithms (8), (9), (10) and (11). We first study the convergence, in the full rank case ($\overline{r} = \min\{p, d\}$), as we decrease the time step Δt. We take $\Delta t = 10^{-2}$, $\Delta t = 10^{-3}$, $\Delta t = 10^{-4}$ and $\Delta t = 10^{-5}$. We sample the parameter set by $p = 100$ and we take $d = 100$ number of Brownian motion realizations. The errors κ_n^{FR}, $\tilde{\kappa}_n^{DO}(r)$, $\overline{\kappa}_n^{DO}(r)$ and $\hat{\kappa}_n^{DO}(r)$ given by the algorithms (8), (9), (10) and (11) respectively are defined by,

$$\kappa_n^{FR} := \left\| \overline{X}^n - \overline{Y}^{SP,n} \right\|, \quad \tilde{\kappa}_n^{DO}(r) := \left\| \overline{X}^n - \tilde{Y}_r^{DO,n} \right\|, \quad \overline{\kappa}_n^{DO}(r) := \left\| \overline{X}^n - \overline{Y}_r^{DO,n} \right\| \text{ and}$$

$$\hat{\kappa}_n^{DO}(r) := \left\| \overline{X}^n - \hat{Y}_r^{DO,n} \right\|.$$

In Figure III.9, we take $r = \bar{r}$, we observe the convergence to the solution of the SDE (III.21) as the time step Δt goes to zero. Which is the required result.

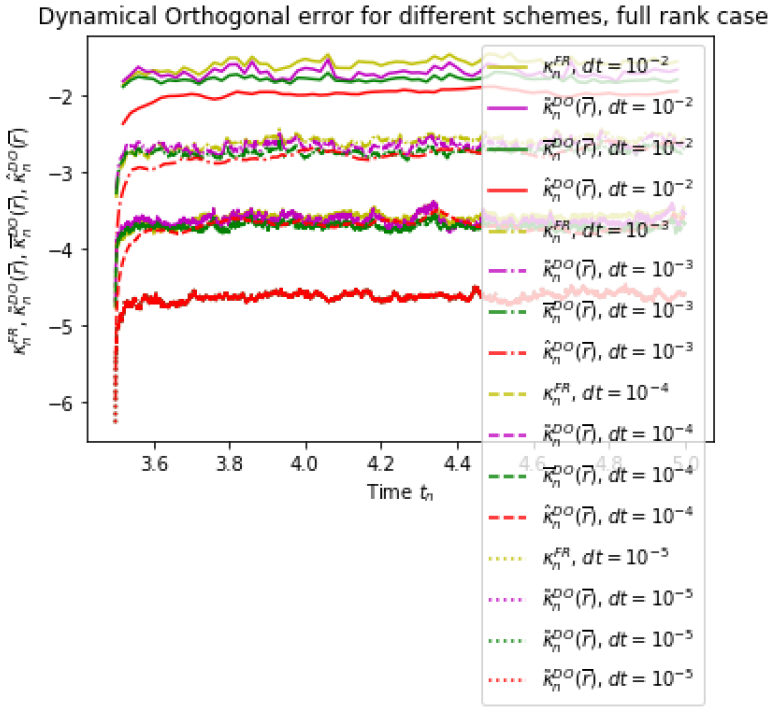


Figure III.9: Convergence results for algorithms (8), (9), (10) and (11) in the full rank case as we decrease the time step Δt.

Let us introduce the errors $\tilde{\kappa}^{DO}(r)$, $\overline{\kappa}^{DO}(r)$ and $\hat{\kappa}^{DO}(r)$ given by the algorithms (9), (10) and (11) for a rank r:

$$\tilde{\kappa}^{DO}(r) = \max_{0 \le n \le N} \tilde{\kappa}_n^{DO}(r), \quad \overline{\kappa}^{DO}(r) = \max_{0 \le n \le N} \overline{\kappa}_n^{DO}(r) \quad \text{and} \quad \hat{\kappa}^{DO}(r) = \max_{0 \le n \le N} \hat{\kappa}_n^{DO}(r).$$

In Figure III.10 we study the convergence of the algorithms (9), (10) and (11) as we increase the rank r for different time steps. We take $r \in [1; 2; 3; 10; 30; 50; 60; 70; 80; 90; 95; 100]$ and for the time step $\Delta t = 10^{-2}$, $\Delta t = 10^{-3}$, $\Delta t = 10^{-4}$ and $\Delta t = 10^{-5}$.

We remark that the algorithms (10) and (11) converge faster for each rank r than the algorithm (9) which shows a very slow convergence as we increase r. In addition, algorithms (10) and (11) are les expensive as they integrate less terms, the algorithm (11) is the cheapest one.

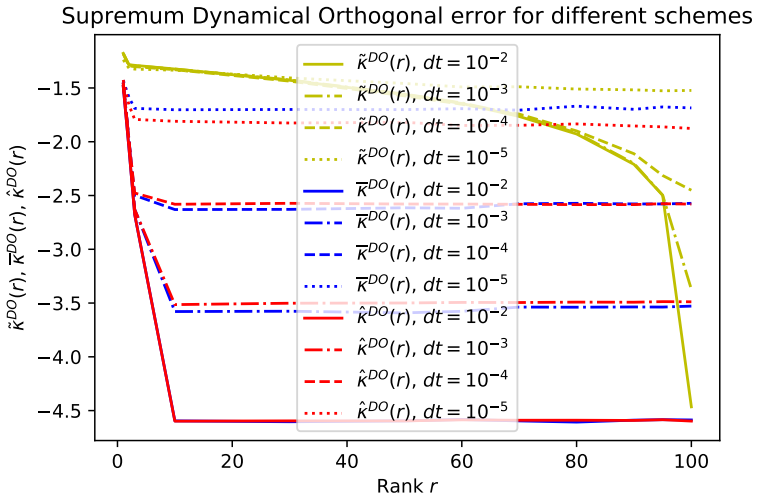


Figure III.10: Convergence of algorithms (9), (10) and (11) for different rank r at different time steps.

III.5.3 An example with a McKean nonlinearity

We study in this part numerically the low rank solution of the following SDE, the exponential Brownian X_t^σ for each $\sigma \in \mathcal{P}$, that contains a McKean nonlinearity:

$$\begin{cases} dX_t^\sigma = \frac{\sigma^2}{2} X_t^\sigma dt + \mathbb{E}[X_t^\sigma]dt + \sigma X_t^\sigma dW_t \\ X_{t=0}^\sigma = X_0^\sigma. \end{cases} \tag{III.23}$$

Notice that we can derive an explicit formula of its expectation $\mathbb{E}[X_t^\sigma]$:

$$\mathbb{E}[X_t^\sigma] = X_0^\sigma + \int_0^t \frac{\sigma^2}{2}\mathbb{E}[X_s^\sigma]ds + \int_0^t \mathbb{E}[X_s^\sigma]ds + \mathbb{E}[\int_0^t X_s^\sigma dW_s] \tag{III.24}$$

The last term is equal to zero and hence we obtain,

$$d\mathbb{E}[X_t^\sigma] = \left(\frac{\sigma^2}{2} + 1\right)\mathbb{E}[X_t^\sigma] \tag{III.25}$$

So that,

$$\mathbb{E}[X_t^\sigma] = X_0^\sigma \exp\left((\frac{\sigma^2}{2} + 1)t\right), \quad \text{for all } t \in [0, T]. \tag{III.26}$$

The parameter σ varies in a random set of cardinality p. Let $(\overline{X}^n)_{0\leq n\leq N}$ be the Euler Maruyama approximation solution of (III.23) in its matricial form.

We propose here 4 different algorithms (12), (13), (14) and (15). The algorithm (13) is similar to the algorithm (10) where we evaluate the expectantion only once in the first equation.

Then, the algoorithm (15) is similar to the algorithm (11), where we evaluate the expectantion only once and where we integrate it only in the first equation.

Let be $\overline{E}(t_n, \overline{Y}^{DO,n}) \in \mathbb{R}^{d_l,p}$ such that $\overline{E}(t_n, \overline{Y}^{DO,n}) = H \otimes \mathbb{E}_{d_l}[(\overline{Y}^{DO,n})]$, where $H = (1, \cdots, 1) \in \mathbb{R}^{d_l}$ and $\mathbb{E}_{d_l}[(\overline{Y}^{DO,n})] \in \mathbb{R}^p$ and $\left(\mathbb{E}_{d_l}[(\overline{Y}^{DO,n})]\right)_j = \mathbb{E}_{d_l}[(\overline{Y}^{DO,n})_j]$ such that,

$$\mathbb{E}_{d_l}[(\overline{Y}^{DO,n})_j] = \frac{1}{d_l}\sum_{i=1}^{d_l}(\overline{Y}^{DO,n})_{i,j} \quad 0 \leq n \leq N \text{ and } 1 \leq j \leq p. \tag{III.27}$$

Let $\overline{\Sigma}(t_n, \overline{Y}^{DO,n})) \in \mathbb{R}^{d_l,p}$ such that $(\overline{\Sigma}(t_n, \overline{Y}^{DO,n}))_{i,j} = \sigma_j \times (\overline{Y}^{DO,n})_{i,j}$.
Let $\hat{Y}^{DO,n}$, $\overline{Y}^{DO,n}$, $\tilde{Y}^{DO,n}$ and $\breve{Y}^{DO,n}$ the dynamical orthogonal solutions given by the following algorithms (12), (14), (13) and (15) respectively.

Algorithm 12 Splitting DO with McKean nonlinearity using implicit scheme for the drift and the McKean term and explicit scheme for the noise term

---

Input: Let $r \in \mathbb{N}^*$, $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let us be given at the step $n = 0$, the approximation $\hat{Y}_r^{DO,0} = \hat{U}^0\hat{S}^0(\hat{V}^0)^T$.
Output: $(\hat{Y}_r^{DO,n})_{0\leq n\leq N}$.
While $0 \leq n \leq N-1$ do:

- **1)** compute $\mathbb{E}_{d_l}[(\hat{Y}_r^{DO,n})]$ defined by (III.27) on $\hat{Y}_r^{DO,n}$,
  $\hat{Y}_1^{DO,n} := \hat{Y}_r^{DO,n} + \left[\Delta t\left[\overline{B}(t_n, \hat{Y}_r^{DO,n}) + \overline{E}(t_n, \hat{Y}_r^{DO,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n\overline{\otimes}\overline{\Sigma}(t_n, \hat{Y}_r^{DO,n})\right] P_{\hat{V}^n}$; compute $\hat{Y}_1^{DO,n} = \hat{U}^{n+1}\hat{S}_1^n(\hat{V}^n)^T$ with $\hat{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\hat{S}_1^n \in \mathbb{R}^{r\times r}$.
- **2)** compute $\mathbb{E}_{d_l}[(\hat{Y}_1^{DO,n})]$ defined by (III.27) on $\hat{Y}_1^{DO,n}$,
  $\hat{Y}_2^{DO,n} = \hat{Y}_1^{DO,n} + P_{\hat{U}^{n+1}}\left[-\Delta t\left[\overline{B}(t_n, \hat{Y}_1^{DO,n}) + \overline{E}(t_n, \hat{Y}_1^{DO,n})\right] - \sqrt{\Delta t}\overline{\overline{G}}_n\overline{\otimes}\overline{\Sigma}(t_n, \hat{Y}^{DO,n})\right] P_{\hat{V}^n}$; compute $\hat{Y}_2^{DO,n} = \hat{U}^{n+1}\hat{S}_2^n(\hat{V}^n)^T$ with $\hat{S}_2^n \in \mathbb{R}^{r\times r}$;
- **3)** compute $\mathbb{E}_{d_l}[(\hat{Y}_2^{DO,n})]$ defined by (III.27) on $\hat{Y}_2^{DO,n}$,
  $\hat{Y}^{DO,n+1} = \hat{Y}_2^{DO,n} + P_{\hat{U}^{n+1}}\left[\Delta t\left[\overline{B}(t_n, \hat{Y}_2^{DO,n}) + \overline{E}(t_n, \hat{Y}_2^{DO,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n\overline{\otimes}\overline{\Sigma}(t_n, \hat{Y}^{DO,n})\right]$; compute $\hat{Y}_r^{DO,n+1} = \hat{U}^{n+1}\hat{S}^{n+1}(\hat{V}^{n+1})^T$ with $\hat{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\hat{S}^{n+1} \in \mathbb{R}^{r\times r}$.
  $n = n + 1$.

Algorithm 13 Splitting DO with McKean nonlinearity using implicit scheme for the drift and the McKean term, used once, and explicit scheme for the noise term

Input: Let $r \in \mathbb{N}^*$, $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let us be given at the step $n = 0$, the approximation $\breve{Y}_r^{DO,0} = \breve{U}^0 \breve{S}^0 (\breve{V}^0)^T$.
Output: $(\breve{Y}_r^{DO,n})_{0 \leq n \leq N}$.
While $0 \leq n \leq N-1$ do:

- **1)** compute $\mathbb{E}_{d_l}[(\breve{Y}^{DO,n})]$ defined by (III.27) on $\breve{Y}_r^{DO,n}$, $\breve{Y}_1^{DO,n} := \breve{Y}^{DO,n} + \left[\Delta t \left[\overline{B}(t_n, \breve{Y}^{DO,n}) + \overline{E}(t_n, \breve{Y}^{DO,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \breve{Y}^{DO,n})\right] P_{\breve{V}^n}$; compute $\breve{Y}_1^{DO,n} = \breve{U}^{n+1}\breve{S}_1^n(\breve{V}^n)^T$ with $\breve{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\breve{S}_1^n \in \mathbb{R}^{r\times r}$;
- **2)** $\breve{Y}_2^{DO,n} = \breve{Y}_1^{DO,n} + P_{\breve{U}^{n+1}}\left[-\Delta t \overline{B}(t_n, \breve{Y}_1^{DO,n}) - \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \breve{Y}^{DO,n})\right] P_{\breve{V}^n}$; compute $\breve{Y}_2^{DO,n} = \breve{U}^{n+1}\breve{S}_2^n(\breve{V}^n)^T$ with $\breve{S}_2^n \in \mathbb{R}^{r\times r}$;
- **3)** $\breve{Y}^{DO,n+1} = \breve{Y}_2^{DO,n} + P_{\overline{U}^{n+1}}\left[\Delta t \overline{B}(t_n, \breve{Y}_2^{DO,n}) + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \breve{Y}^{DO,n})\right]$; compute $\breve{Y}_r^{DO,n+1} = \breve{U}^{n+1}\breve{S}^{n+1}(\breve{V}^{n+1})^T$ with $\breve{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\breve{S}^{n+1} \in \mathbb{R}^{r\times r}$.
  $n = n+1$.

Algorithm 14 Splitting DO with McKean nonlinearity using implicit scheme for the drift and the McKean term and explicit scheme for the noise term used once

Input: Let $r \in \mathbb{N}^*$, $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let us be given at the step $n = 0$, the approximation $\overline{Y}_r^{DO,0} = \overline{U}^0 \overline{S}^0 (\overline{V}^0)^T$.
Output: $(\overline{Y}_r^{DO,n})_{0 \leq n \leq N}$.
While $0 \leq n \leq N-1$ do:

- **1)** compute $\mathbb{E}_{d_l}[(\overline{Y}_r^{DO,n})]$ defined by (III.27) on $\hat{Y}_r^{DO,n}$, compute $\overline{Y}_1^{DO,n} := \overline{Y}_r^{DO,n} + \left[\Delta t \left[\overline{B}(t_n, \overline{Y}^{DO,n}) + \overline{E}(t_n, \overline{Y}^{DO,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \overline{Y}^{DO,n})\right] P_{\overline{V}^n}$; compute $\overline{Y}_1^{DO,n} = \overline{U}^{n+1}\overline{S}_1^n(\overline{V}^n)^T$ with $\overline{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\overline{S}_1^n \in \mathbb{R}^{r\times r}$;
- **2)** compute $\mathbb{E}_{d_l}[(\overline{Y}_1^{DO,n})]$ defined by (III.27) on $\overline{Y}_1^{DO,n}$, $\overline{Y}_2^{DO,n} = \overline{Y}_1^{DO,n} + P_{\overline{U}^{n+1}}\left[-\Delta t \left[\overline{B}(t_n, \overline{Y}_1^{DO,n}) + \overline{E}(t_n, \overline{Y}_1^{DO,n})\right]\right] P_{\overline{V}^n}$; compute $\overline{Y}_2^{DO,n} = \overline{U}^{n+1}\overline{S}_2^n(\overline{V}^n)^T$ with $\overline{S}_2^n \in \mathbb{R}^{r\times r}$;
- **3)** compute $\mathbb{E}_{d_l}[(\overline{Y}_2^{DO,n})]$ defined by (III.27) on $\overline{Y}_2^{DO,n}$, $\overline{Y}^{DO,n+1} = \overline{Y}_2^{DO,n} + P_{\overline{U}^{n+1}}\left[\Delta t \left[\overline{B}(t_n, \overline{Y}_2^{DO,n}) + \overline{E}(t_n, \overline{Y}_2^{DO,n})\right]\right]$; compute $\overline{Y}_r^{DO,n+1} = \overline{U}^{n+1}\overline{S}^{n+1}(\overline{V}^{n+1})^T$ with $\overline{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\overline{S}^{n+1} \in \mathbb{R}^{r\times r}$.
  $n = n+1$.

Algorithm 15 Splitting DO with McKean nonlinearity using implicit scheme for the drift and the McKean term, used once, and explicit scheme for the noise term used once

Input: Let $r \in \mathbb{N}^*$, $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let us be given at the step $n = 0$, the approximation $\tilde{Y}_r^{DO,0} = \tilde{U}^0 \tilde{S}^0 (\tilde{V}^0)^T$.
Output: $(\tilde{Y}_r^{DO,n})_{0 \leq n \leq N}$.
While $0 \leq n \leq N-1$ do:

- **1)** compute $\mathbb{E}_{d_l}[(\tilde{Y}_r^{DO,n})]$ defined by (III.27) on $\tilde{Y}_r^{DO,n}$,
  compute $\tilde{Y}_1^{DO,n} := \tilde{Y}_r^{DO,n} + \left[\Delta t \left[\overline{B}(t_n, \tilde{Y}^{DO,n}) + \overline{E}(t_n, \tilde{Y}^{DO,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes\Sigma}(t_n, \tilde{Y}^{DO,n})\right] P_{\overline{V}^n}$;
  compute $\tilde{Y}_1^{DO,n} = \tilde{U}^{n+1}\tilde{S}_1^n(\tilde{V}^n)^T$ with $\tilde{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\tilde{S}_1^n \in \mathbb{R}^{r\times r}$;
- **2)** $\tilde{Y}_2^{DO,n} = \tilde{Y}_1^{DO,n} + P_{\tilde{U}^{n+1}}\left[-\Delta t \overline{B}(t_n, \tilde{Y}_1^{DO,n})\right] P_{\tilde{V}^n}$; compute $\tilde{Y}_2^{DO,n} = \tilde{U}^{n+1}\tilde{S}_2^n(\tilde{V}^n)^T$ with $\tilde{S}_2^n \in \mathbb{R}^{r\times r}$;
- **3)** $\tilde{Y}^{DO,n+1} = \tilde{Y}_2^{DO,n} + P_{\tilde{U}^{n+1}}\left[\Delta t \overline{B}(t_n, \tilde{Y}_2^{DO,n})\right]$; compute $\tilde{Y}_r^{DO,n+1} = \tilde{U}^{n+1}\tilde{S}^{n+1}(\tilde{V}^{n+1})^T$ with $\tilde{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\tilde{S}^{n+1} \in \mathbb{R}^{r\times r}$.
  $n = n+1$.

III.5.4 Numerical experiments on the McKean nonlinear case

Let us introduce the errors:

$$\kappa_n^D(r) := \left\|\overline{X}^n - \overline{Y}_r^{DO,n}\right\|_F, \qquad \kappa_n^{DE}(r) := \left\|\overline{X}^n - \tilde{Y}_r^{DO,n}\right\|_F$$

$$\kappa_n^C(r) := \left\|\overline{X}^n - \hat{Y}_r^{DO,n}\right\|_F, \qquad \kappa_n^{CE}(r) := \left\|\overline{X}^n - \check{Y}_r^{DO,n}\right\|_F$$

In Figure III.11, we take a step size $\Delta t = 10^{-3}$, the parameter set is sampled between $\sigma = 0.5$ and $\sigma = 1$ with cardinality $p = 100$ and the number of samples is $d_l = 10^4$. The rank r is set to $r = 5$. We remark that we have almost the same order of error for the four schemes with a better approximation for the algorithms (13) and (15) with, additionally, a smaller cost than for algorithms (12) and (14) .

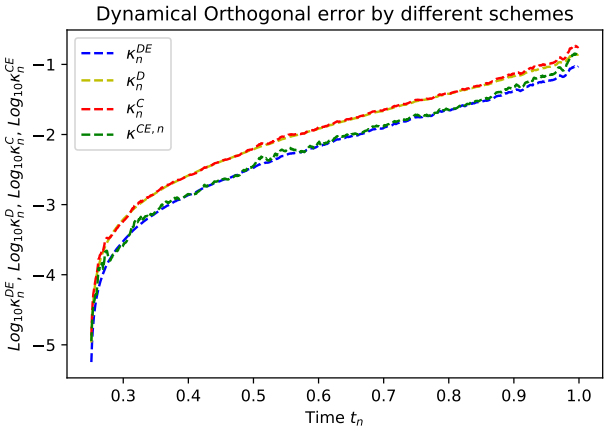


Figure III.11: Dynamical Orthogonal error using algorihtms (12), (14), (15) and (13) on the exponential Brownian case.

Let us introduce the errors $\kappa^C(r)$, $\kappa^D(r)$, $\kappa^{DE}(r)$ and $\kappa^{CE}(r)$ given by the algorithms (12), (14), (15) and (13) for a rank r:

$$\kappa^C(r) = \max_{0\leq n\leq N} \kappa_n^C(r), \quad \kappa^D(r) = \max_{0\leq n\leq N} \kappa_n^D(r), \quad \kappa^{DE}(r) = \max_{0\leq n\leq N} \kappa_n^{DE}(r) \quad \text{and} \quad \kappa^{CE}(r) = \max_{0\leq n\leq N} \kappa_n^{CE}(r).$$

In Figure III.12 we study the convergence of the algorithms (12), (14), (15) and (13) as we increase the rank r for different time steps. We take $r \in [1; 2; 3; 10; 30; 50; 60; 70; 80; 90; 95; 100]$ and for the time step $\Delta t = 10^{-2}$, $\Delta t = 10^{-3}$, $\Delta t = 10^{-4}$ and $\Delta t = 10^{-5}$.

We remark that the algorithms (12), (13), (14) and (15) converge quite in the same order for each rank r. Adding the fact that the algorithm (15) is the cheapest one as we calculate the multiplicative noise and the McKean term only once.

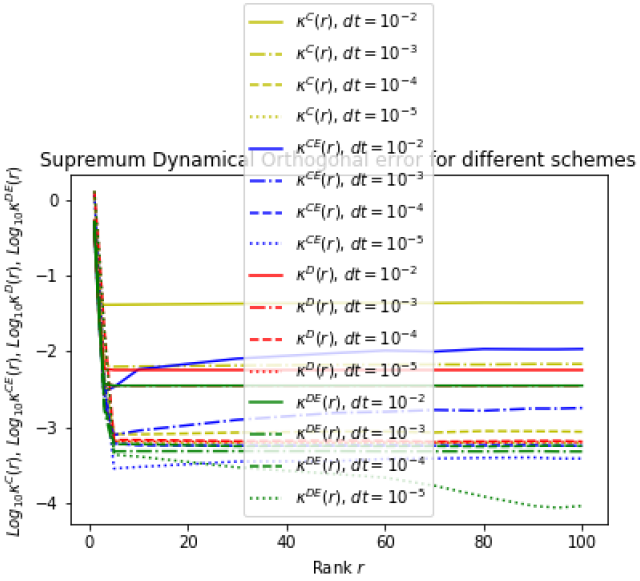


Figure III.12: Convergence of algorithms (12), (14), (15) and (13) for different rank r at different time steps.

Let $\overline{X}_{t_n}$ be the continuous solution of (III.23), at time t_n, in its matricial form and let us consider the errors,

$$\Lambda_j^{D,n}(r) = \frac{\mathbb{E}[(\overline{X}_{t_n})_j] - \mathbb{E}_{d_l}(\overline{Y}_r^{DO,n})_j]}{\mathbb{E}[(\overline{X}_{t_n})_j]}, \qquad \Lambda_j^{DE,n}(r) = \frac{\mathbb{E}[(\overline{X}_{t_n})_j] - \mathbb{E}_{d_l}[(\tilde{Y}_r^{DO,n})_j]}{\mathbb{E}[(\overline{X}_{t_n})_j]}$$

$$\Lambda_j^{C,n}(r) = \frac{\mathbb{E}[(\overline{X}_{t_n})_j] - \mathbb{E}_{d_l}[(\hat{Y}_r^{DO,n})_j]}{\mathbb{E}[(\overline{X}_{t_n})_j]}, \qquad \Lambda_j^{CE,n}(r) = \frac{\mathbb{E}[(\overline{X}_{t_n})_j] - \mathbb{E}_{d_l}[(\breve{Y}_r^{DO,n})_j]}{\mathbb{E}[(\overline{X}_{t_n})_j]}$$

In Figure III.13 are plotted four curves, representing the relative error on the expectations between the estimators $\mathbb{E}_{d_l}[(\overline{Y}_r^{DO,n})]$, $\mathbb{E}_{d_l}[(\tilde{Y}_r^{DO,n})]$, $\mathbb{E}_{d_l}[(\hat{Y}_r^{DO,n})]$ and $\mathbb{E}_{d_l}[(\breve{Y}_r^{DO,n})]$ compared to the exact expectation $\mathbb{E}[(\overline{X}_{t_n})]$. Note that these curves correspond to the worst error among all the parameters σ, which is obtained for $\sigma = 1.92$. We remark a maximal error that reaches 12 percent for all the errors. Of course we can reduce this statistical error by increasing the Monte Carlo sampling d_l.

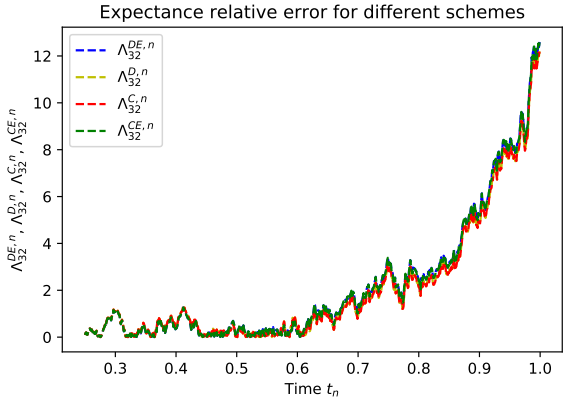


Figure III.13: Relative error expectation bewteen the estimators $\mathbb{E}_{d_l}[(\hat{Y}_r^{DO,n})_{32}]$, $\mathbb{E}_{d_l}[(\tilde{Y}_r^{DO,n})_{32}]$, $\mathbb{E}_{d_l}[(\overline{Y}_r^{DO,n})_{32}]$ and $\mathbb{E}_{d_l}[(\breve{Y}_r^{DO,n})_{32}]$ and the exact expectation at each time t_n, for the worst obtained case ($\sigma = 1.92$).

III.6 Dynamical Orthogonal approximation for control variate variance reduction on the additive noise

Several experiments have been done on randomly sampling the parameter space using the same Brownian realizations and inversely sampling different Brownian realizations using the same parameter space. From these results that we present in the annex (IV) we can remark, that the projector splitting dissociates the effect of the realizations and the parameters on the model. We have seen that if we keep the same set of parameters with changing the realizations, then we keep the same deterministic modes, and if we change the set of parameters with using the same realizations then we obtain the same stochastic modes. We have also seen that the rank dosen't effect the dynamical error as we increase the number of realizations d from a given number of

realizations d_0 ($d_0 \simeq 800$ in Figure IV.4) with a fixed set of parameters. These results motivate us to develop a method that calculates the expectation with a faster way.

We illustrate in this section how the DO approximation introduced in the preceding section can be used as a control variate in order to quickly compute expectations of quantities depending on the solution of the parametric SDE of interest. For the sake of illustration, we choose here to develop a variance reduction method for the computation of $\mathbb{E}\left[X_{t_n}^{\mu}\right]$ for $\mu \in \mathcal{P}$.

The idea of constructing the reduced estimator is based on two steps. First, in an offline phase, we compute a DO approximation of the parametric SDE for $\mu \in \mathcal{P} = \{\mu_1, \ldots, \mu_p\}$ with a projector splitting method with rank r using only a small number of realizations of the stochastic noise $d = d_s$, the aim of this first simulation is to find the best deterministic modes by increasing at each time the sampling number p of the parameter set. We thus obtain, for all $0 \leq n \leq N$, a dynamical orthogonal model $\overline{Y}_r^{DOs,n} = U_r^{DOs,n} S_r^{DOs,n} \left(V_r^{DOs,n}\right)^T$, with $U_r^{DOs,n} \in \mathcal{V}_{d_s,r}$, $S_r^{DOs,n} \in \mathbb{R}^{r\times r}$ and $V_r^{DOs,n} \in \mathcal{V}_{p,r}$. The obtained deterministic modes $\left(V_r^{DOs,n}\right)_{0\leq n\leq N}$ are then used in turn in order to compute new random realizations with $d = d_l$ of the DO reduced-order model for any random realizations of the stochastic noise, the aim of this simulation is to construct the high fidelty stochastic modes. For a given family of independent identically distributed random variables $(G_n)_{0\leq n\leq N}$, the DO reduced-order model on the deterministic modes $\tilde{Y}^{\mathrm{red}_{off},n}$ of the solution of the Euler-Maruyama scheme $\overline{X}^n$ is then computed as an element belonging to the linear space spanned by the columns of $V_r^{DOs,n}$ as follows:

$$\tilde{Y}^{\mathrm{red}_{off},n+1} = \left[\tilde{Y}^{\mathrm{red}_{off},n} + \Delta t \bar{B}(t_n, \tilde{Y}^{\mathrm{red}_{off},n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right] \tilde{V}_r^{DOs,n+1} (\tilde{V}_r^{DOs,n+1})^T.$$

This first reduced-order model on the deterministic modes $(\tilde{Y}^{\mathrm{red}_{off},n} = \tilde{U}^{DO_l,n} \tilde{S}^n (\tilde{V}^n)^T)_{0\leq n\leq N}$, with $\tilde{U}_r^{DO_l,n} \in \mathcal{V}_{d_l,r}$, $\tilde{S}_r^n \in \mathbb{R}^{r\times r}$ and $\tilde{V}_r^n \in \mathcal{V}_{p,r}$ is then used to fix the stochastic modes of the control variate that we construct in the online phase.
Indeed, in the online phase, for any given new set of parameters, we construct the reduced-order model on the stochastic modes $\tilde{Y}^{\mathrm{red}_{on},n}$ of the solution of the Euler-Maruyama scheme $\overline{X}^n$ (given by the new set of parameters) as an element belonging to the vector space spanned by the stochastic columns of $\tilde{U}_r^{DO_l,n}$ as follows:

$$\tilde{Y}^{\mathrm{red}_{on},n+1} = \tilde{U}_r^{DO_l,n+1} (\tilde{U}_r^{DO_l,n+1})^T \left[\tilde{Y}^{\mathrm{red}_{on},n} + \Delta t \bar{B}(t_n, \tilde{Y}^{\mathrm{red}_{on},n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right].$$

This approximation $\tilde{Y}^{\mathrm{red}_{on},n}$ is taken as a control variate in order to reduce the variance of a Monte-Carlo estimator for the computation of the desired expectation.
We now present the different algorithms that we use to extract the deterministic modes and the stochastic modes that we use in the algorithm (20) destinated to construct the control variate.

III.6.1 Algorithms with fixed Deterministic modes

We present two splitting reduced algorithms that aim to extract the stochastic modes. As the projector splitting scheme follows a specific order to compute the factors, we have to consider the transpose matrix $\tilde{Y}^{red,n,T} = (\tilde{Y}^{red,n})^T$ to run the reduced algorithm that we present in the following.

First fixed deterministic modes algorithm

Let be, for all $n \in \mathbb{N}^*$, $\tilde{V}^{n+1} = V_r^{DOs,n+1}$ then, the reduced splitting algorithm on the deterministic modes reads as follow,

1) compute $\tilde{Y}_1^{red,n,T} := P_{\tilde{V}^{n+1}}\left[\tilde{Y}^{red,n,T} + \left[\Delta t \overline{B}^T(t_n, \tilde{Y}^{red,n,T}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)^T\right] P_{\tilde{U}^n}\right]$; compute $\tilde{Y}_1^{red,n,T} = \tilde{V}^{n+1}\tilde{S}_1^n(\tilde{U}^n)^T$ with $\tilde{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\tilde{S}_1^n \in \mathbb{R}^{r\times r}$.

2) $\tilde{Y}_2^{red,n,T} = P_{\tilde{V}^{n+1}}\left[\tilde{Y}_1^{red,n,T} - P_{\tilde{V}^{n+1}}\left[\Delta t \overline{B}^T(t_n, \tilde{Y}_1^{red,n,T}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)^T\right] P_{\tilde{U}^n}\right]$; compute $\tilde{Y}_2^{red,n,T} = \tilde{V}^{n+1}\tilde{S}_2^n(\tilde{U}^n)^T$ with $\tilde{S}_2^n \in \mathbb{R}^{r\times r}$;

3) $\tilde{Y}^{red,n+1,T} = P_{\tilde{V}^{n+1}}\left[\tilde{Y}_2^{red,n,T} + P_{\tilde{V}^{n+1}}\left[\Delta t \overline{B}^T(t_n, \tilde{Y}_2^{red,n,T}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)^T\right]\right]$; compute $\tilde{Y}_r^{red,n+1,T} = \tilde{V}^{n+1}\tilde{S}^{n+1,T}(\tilde{U}^{n+1})^T$ with $\tilde{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\tilde{S}^{n+1} \in \mathbb{R}^{r\times r}$.

Which is resumed to one step:

1) $\tilde{Y}^{red,n+1,T} = P_{\tilde{V}^{n+1}}\tilde{Y}^{red,n,T} + P_{\tilde{V}^{n+1}}\left[\Delta t(\overline{B}^T(t_n, \tilde{Y}^{red,n,T}) - \overline{B}^T(t_n, \tilde{Y}_1^{red,n,T}))\right] P_{\tilde{U}^{n+1}} + P_{\tilde{V}^{n+1}}\left[\Delta t \overline{B}^T(t_n, \tilde{Y}_2^{red,n,T}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)^T\right]$;

In terms of factors this would lead to find at each iteration n, $\tilde{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\tilde{S}^{n+1} \in \mathbb{R}^{r\times r}$ solution of algorithm (16).

Algorithm 16 First fixed deterministic modes algorithm, additive noise

Input: Let $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let us be given at the step $n = 0$, the approximation $\tilde{Y}^{red,n}$ and $\tilde{V}^n = V^{DOs,n}$ for all $0 \leq n \leq N$.
Output: $(\tilde{Y}^{red,n})_{0\leq n\leq N}$.
While $0 \leq n \leq N-1$ do:

- 1) $(\tilde{S}^{n+1})^T(\tilde{U}^{n+1})^T = (\tilde{V}^{n+1})^T\tilde{Y}^{red,n,T} + (\tilde{V}^{n+1})^T\left[\Delta t(\overline{B}^T(t_n, \tilde{Y}^{red,n,T}) - \overline{B}^T(t_n, \tilde{Y}_1^{red,n,T}))\right] + (\tilde{V}^{n+1})^T\left[\Delta t \overline{B}^T(t_n, \tilde{Y}_2^{red,n,T}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)^T\right]$; compute

$$\tilde{Y}^{red,n+1} = \tilde{U}^{n+1}\tilde{S}^{n+1}(\tilde{V}^{n+1})^T.$$

$n = n + 1$.

Second fixed deterministic modes algorithm

The second fixed deterministic modes algorithm for stochastic modes is based on neglecting the following term in the first algorithm,

$$P_{\tilde{V}^{n+1}}\left[\Delta t(\overline{B}^T(t_n, \tilde{Y}^{red,n,T}) - \overline{B}^T(t_n, \tilde{Y}_1^{red,n,T}))\right] P_{\tilde{U}^{n+1}}$$

Hence we propose the following second algorithm (17) on the deterministic modes.

Algorithm 17 Second fixed deterministic modes algorithm, additive noise

Input: Let $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let us be given at the step $n = 0$, the approximation $\hat{Y}^{red,n}$ and $\hat{V}^n = V^{DOs,n}$ for all $0 \leq n \leq N$.
Output: $(\hat{Y}^{red,n})_{0\leq n\leq N}$.
While $0 \leq n \leq N-1$ do:

- 1) $(\hat{S}^{n+1})^T(\hat{U}^{n+1})^T = (\hat{V}^{n+1})^T\hat{Y}^{red,n,T} + (\hat{V}^{n+1})^T\left[\Delta t\overline{B}^T(t_n, \hat{Y}^{red,n,T}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)^T\right]$; compute
$$\hat{Y}^{red,n+1} = \hat{U}^{n+1}\hat{S}^{n+1}(\hat{V}^{n+1})^T.$$
$n = n + 1$.

Note that in this way, we obtain a gain of 40 percent of the computational time compared to the Projector Splitting algorithm (6) with rank r for the first algorithm (16), and 60 percent of the computational time for the second algorithm (17), (this gain is calculated on our specific model).

Next we adapt both algorithms presented above for generating new deterministic modes of a new set of parameters by using fixed stochastic modes.

III.6.2 Algorithms with fixed Stochastic modes

The idea of constructing the reduced estimator for generating deterministic modes is based on two steps. First, we compute a DO approximation of the parametric SDE for $\mu \in \mathcal{P}_{\text{train}} = \{\mu_1, \ldots, \mu_{p_t}\}$, of cardinality p_t, with a projector splitting method with rank r using only a number of realizations of the stochastic noise d. For all $0 \leq n \leq N$, we thus obtain the DO approximation $\overline{Y}_r^{DO_{p_t},n} = U_r^{DO_{p_t},n}S_r^{DO_{p_t},n}\left(V_r^{DO_{p_t},n}\right)^T$, with $U_r^{DO_{p_t},n} \in \mathcal{V}_{d,r}$, $S_r^{DO_{p_t},n} \in \mathbb{R}^{r\times r}$ and $V_r^{DO_{p_t},n} \in \mathcal{V}_{p,r}$. The obtained stochastic modes $\left(U_r^{DO_{p_t},n}\right)_{0\leq n\leq N}$ are then used in turn in order to compute the DO reduced-order model for any set of parameters $\mathcal{P}_{test}$ of cardinal p_e. For a given family of independent identically distributed random variables $(G_n)_{0\leq n\leq N}$, a reduced-order model $\tilde{Y}^{\text{red},n}$ of the solution of the Euler-Maruyama scheme $\overline{X}^n$ is then computed as an element belonging to the vector space spanned by the columns of $U_r^{DO_{p_t},n}$ as follows:

$$\tilde{Y}^{\text{red},n+1} = U_r^{DO_{p_t},n+1}(U_r^{DO_{p_t},n+1})^T\left[\tilde{Y}^{\text{red},n} + \Delta t\bar{B}(t_n, \tilde{Y}^{\text{red},n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right].$$

First fixed stochastic modes algorithm

Let be, for all $n \in \mathbb{N}^*$, $\tilde{U}^{n+1} = U_r^{DO_{p_t},n+1}$ then,

1) compute $\tilde{Y}_1^{red,n} := P_{\tilde{U}^{n+1}}\left[\tilde{Y}^{red,n} + \left[\Delta t\overline{B}(t_n, \tilde{Y}^{red,n}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)\right]P_{\tilde{V}^n}\right]$; compute $\tilde{Y}_1^{red,n} = \tilde{U}^{n+1}\tilde{S}_1^n(\tilde{V}^n)^T$ with $\tilde{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\tilde{S}_1^n \in \mathbb{R}^{r\times r}$.

2) $\tilde{Y}_2^{red,n} = P_{\tilde{U}^{n+1}}\left[\tilde{Y}_1^{red,n} - P_{\tilde{U}^{n+1}}\left[\Delta t\overline{B}(t_n, \tilde{Y}_1^{red,n}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)\right]P_{\tilde{V}^n}\right]$; compute $\tilde{Y}_2^{red,n} = \tilde{U}^{n+1}\tilde{S}_2^n(\tilde{V}^n)^T$ with $\tilde{S}_2^n \in \mathbb{R}^{r\times r}$;

3) $\tilde{Y}^{red,n+1} = P_{\tilde{U}^{n+1}}\left[\tilde{Y}_2^{red,n} + P_{\tilde{U}^{n+1}}\left[\Delta t\overline{B}(t_n,\tilde{Y}_2^{red,n}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)\right]\right]$; compute $\tilde{Y}_r^{red,n+1} = \tilde{U}^{n+1}\tilde{S}^{n+1}(\tilde{V}^{n+1})^T$ with $\tilde{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\tilde{S}^{n+1} \in \mathbb{R}^{r\times r}$.

Which is resumed to one step:

1) $\tilde{Y}^{red,n+1} = P_{\tilde{U}^{n+1}}\tilde{Y}^{red,n} + P_{\tilde{U}^{n+1}}\left[\Delta t(\overline{B}(t_n,\tilde{Y}^{red,n}) - \overline{B}(t_n,\tilde{Y}_1^{red,n}))\right]P_{\tilde{V}^{n+1}} + P_{\tilde{U}^{n+1}}\left[\Delta t\overline{B}(t_n,\tilde{Y}_2^{red,n}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)\right]$;

In terms of factors this would lead, at each iteration n to find $\tilde{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\tilde{S}^{n+1} \in \mathbb{R}^{r\times r}$ solution of algorithm (18).

Algorithm 18 First fixed stochastic modes algorithm, additive noise

Input: Let $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let us be given at the step $n = 0$, the approximation $\tilde{Y}^{red,n}$ and $\tilde{U}^{n+1} = U_r^{DO_{p_t},n+1}$ for all $0 \le n \le N$.
Output: $(\tilde{Y}^{red,n})_{0\le n\le N}$.
While $0 \le n \le N-1$ do:

- 1) $(\tilde{S}^{n+1})(\tilde{V}^{n+1})^T = (\tilde{U}^{n+1})^T\tilde{Y}^{red,n} + (\tilde{U}^{n+1})^T\left[\Delta t(\overline{B}(t_n,\tilde{Y}^{red,n}) - \overline{B}(t_n,\tilde{Y}_1^{red,n}))\right] + (\tilde{U}^{n+1})^T\left[\Delta t\overline{B}(t_n,\tilde{Y}_2^{red,n}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)\right]$; compute

$$\tilde{Y}^{red,n+1} = \tilde{U}^{n+1}\tilde{S}^{n+1}(\tilde{V}^{n+1})^T.$$

$n = n + 1$.

Second fixed stochastic modes algorithm

The second algorithm is based also on neglecting,

$$(\tilde{U}^{n+1})^T\left[\Delta t(\overline{B}(t_n,\tilde{Y}^{red,n}) - \overline{B}(t_n,\tilde{Y}_1^{red,n}))\right], \quad \text{for all} \quad 0 \le n \le N.$$

Thus the second algorithm for determinstic modes is defined in algorithm (19).

Algorithm 19 Second fixed stochastic modes algorithm, additive noise

Input: Let $T > 0$, $\Delta t > 0$ and $N \in \mathbb{N}^*$ s.t $N = \frac{T}{\Delta t}$. Let us be given at the step $n = 0$, the approximation $\hat{Y}^{red,n}$ and $\hat{U}^{n+1} = U_r^{DO_{p_t},n+1}$ for all $0 \le n \le N$.
Output: $(\hat{Y}^{red,n})_{0\le n\le N}$.
While $0 \le n \le N-1$ do:

- 1) $(\hat{S}^{n+1})(\hat{V}^{n+1})^T = (\hat{U}^{n+1})^T\hat{Y}^{red,n,T} + (\hat{U}^{n+1})^T\left[\Delta t\overline{B}(t_n,\hat{Y}^{red,n}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)\right]$; compute

$$\hat{Y}^{red,n+1} = \hat{U}^{n+1}\hat{S}^{n+1}(\hat{V}^{n+1})^T.$$

$n = n + 1$.

Now we are able to present the algorithm for constructing the control variate from the dynamical orthogonal approximation.

III.6.3 Algorithm DO as control variate

Algorithm 20 Dynamical Orthogonal approximation as Control Variate

Offline phase: Let $\mathcal{P}_t$ the training set of cardinality p_t, let d_s be a small realizations of the Brownian motion, let r be the chosen rank and ϵ a threeshold.

- **1)** Fix the sampling number p_t for the parameter set: Compute the Projector Splitting algorithm (6) on $\overline{Y}^{DO_s,n}_{Off,r} \in \mathbb{R}^{d_s \times p_t}$ as

$$\overline{Y}^{DO_s,n}_{Off,r} = U^n S^n (V^n)^T$$

  and keep the stochastic modes $(U^n)_{0\leq n\leq N}$.

- **2)** Let a new set of parameters $\mathcal{P}_t$. Compute the algorithm Projector Splitting (18) or (19) with fixed stochastic modes on $\tilde{Y}^{red_{p_t},n}_{Off,r}$ ($\hat{Y}^{red_{p_t},n}_{Off,r}$ respectively) using the previous stochastic modes U^n as

$$\tilde{Y}^{red_{p_t},n}_{Off,r} = U^n S^{red,n} (V^{red,n}_{Off})^T \quad \text{or} \quad \hat{Y}^{red_{p_t},n}_{Off,r} = U^n S^{red,n} (V^{red,n}_{Off})^T.$$

  If the dynamical orthogonal error $\epsilon^{red,1}_{n,p_t}(r) < \epsilon$ ($\epsilon^{red,2}_{n,p_t}(r) < \epsilon$, respectively defined in (III.29)) then keep p_t and the deterministic modes $(V^{red,n}_{Off})_{0\leq n\leq N}$. Else increase p_t and repeat (1) and (2).

- **3)** High fidelity resolution for stochastic modes: let us consider a large number d_l of realizations of the Brownian motion. Compute the algorithm Projector Splitting (16) or (17) with fixed deterministic modes on $\tilde{Y}^{red_{d_l},n}_{Off,r} \in \mathbb{R}^{d_l \times p_t}$ ($\hat{Y}^{red_{d_l},n}_{Off,r} \in \mathbb{R}^{d_l \times p_t}$ respectively) using the previous deterministic modes $V^{red,n}_{Off}$ as,

$$\tilde{Y}^{red_{d_l},n}_{Off,r} = U^n_{Off} S^n_{Off} (V^{red,n}_{Off})^T \quad \text{or} \quad \hat{Y}^{red_{d_l},n}_{Off,r} = U^n_{Off} S^n_{Off} (V^{red,n}_{Off})^T.$$

  Keep the stochastic modes $(U^n_{Off})_{0\leq n\leq N}$.

Online phase: let be $\mathcal{P}_e$ a set of parameters of cardinality p_e.

- **4)** Compute the algorithm projector splitting (18) or (19) with fixed stochastic modes on $\tilde{Y}^{red_{p_e},n}_{On,r} \in \mathbb{R}^{d_l \times p_e}$ ($\hat{Y}^{red_{p_e},n}_{On,r} \in \mathbb{R}^{d_l \times p_e}$ respectively) using the previous stochastic modes U^n_{Off} as,

$$\tilde{Y}^{red_{p_e},n}_{On,r} = U^n_{Off} S^n_{On} (V^n_{On})^T \quad \text{or} \quad \hat{Y}^{red_{p_e},n}_{On,r} = U^n_{Off} S^n_{On} (V^n_{On})^T.$$

In the next section we present some numerical results on each step of the Offline phase.

III.6.4 Some Results on the offline phase

a-Results for generating new parametric modes

In this part we take $p_t = 900$, and $d = 1000$, we run the simulations in the interval $[3.2, 4]$ with a time step $\Delta t = 0.001$ and we approximate the solution with a rank $r = 30$ and $\overline{r} = \min(d, p_t)$. Let be $\overline{Y}^{DO_s,n}_{Off,r}$ the solution given in step (1) and $\overline{Y}^{DO_s,n}_{Off,\overline{r}}$ the solution given by algorithm (6), let be $\tilde{Y}^{red_{p_t},n}_{Off,r}$ the solution given in step (2) by the algorithm (18) and $\overline{Y}^{red_{p_t},n}_{Off,\overline{r}}$ the solution given by algorithm (6) on the new parameter set, let be $\hat{Y}^{red_{p_t},n}_{Off,r}$ the solution given in step (2) by the algorithm (19), let be $\breve{Y}^{DO_s,n}_{Off,r}$ the solution given in step (1) with the new parameter set and $\breve{Y}^{DO_s,n}_{Off,\overline{r}}$ the solution given by the algorithm (6) with the new parameter set. Then let be the following dynamical errors in the step (1) and (2),

$$\epsilon_{n,p_t}(r) := \left\| \overline{Y}^{DO_s,n}_{Off,\overline{r}} - \overline{Y}^{DO_s,n}_{Off,r} \right\|_F \quad , \quad \overline{\epsilon}_{n,p_t}(r) := \left\| \breve{Y}^{DO_s,n}_{Off,\overline{r}} - \breve{Y}^{DO_s,n}_{Off,r} \right\|_F \tag{III.28}$$

and

$$\epsilon^{red,1}_{n,p_t}(r) := \left\| \overline{Y}^{DO_{p_t},n}_{Off,\overline{r}} - \tilde{Y}^{red_{p_t},n}_{Off,r} \right\|_F \quad , \quad \epsilon^{red,2}_{n,p_t}(r) := \left\| \overline{Y}^{DO_{p_t},n}_{Off,\overline{r}} - \hat{Y}^{red_{p_t},n}_{Off,r} \right\|_F \tag{III.29}$$

a-1-Frobenius Errors between different schemes

In Figure III.14, we plot the error $\epsilon_{n,p_t}(30)$ obtained in the offline phase in step (1), with the set of parameters $\mathcal{P}_{train}$ (in green), and we plot the error $\epsilon^{red,1}_{n,p_t}(30)$ obtained in the offline phase, step (2), on a new $\mathcal{P}_{train}$ with a caridnal p_t (in red) and the error $\epsilon^{red,2}_{n,p_t}(30)$ (in black) with d trajectories, we add the error $\overline{\epsilon}_{n,p_t}(30)$ obtained on the new set of parameters (in blue). Remark that, on the left where we have used $p_t = 200$, we have quite the same order between different methods, this shows that the sampling number p_t is enough good to obtain good deterministic modes. While in the right plot where we have used $p_t = 50$ the sampling number p_t is not enough high to obtain good deterministic modes.

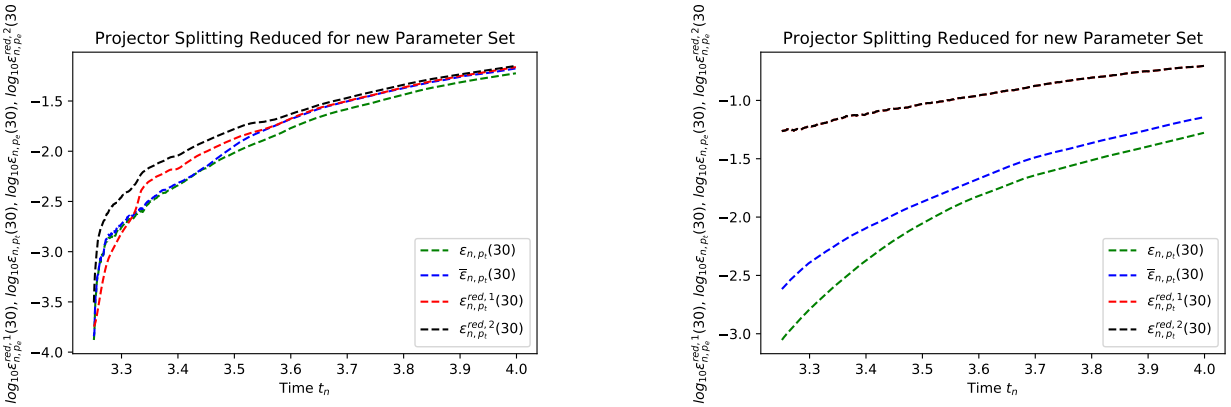


Figure III.14: Reduced Dynamical orthogonal error for a new set of parameters with fixed realizations, using $p_t = 200$ in the left and $p_t = 50$ in the right.

a-2-Effect of increasing the cardinal of the parameter training set, p_t, on the deterministic modes given by the reduced model

Here we change the cardinal of the training set $p_t = 500$, $p_t = 1000$ and $p_t = 2000$, and we look at the effect on the deterministic modes given by the reduced model (18) on a new set of parameters $\mathcal{P}_{test}$ of cardinality p_e taken at each time equal to p_t. We remark, in Figure III.15, that more we sample the training set and better is the approximation. Let be

$$\Theta_n^{Det,red,1}(\hat{r}) := \left\| V_{:\hat{r}}^n - \tilde{V}_{Off,:\hat{r}}^{red,n} \right\|_F,$$

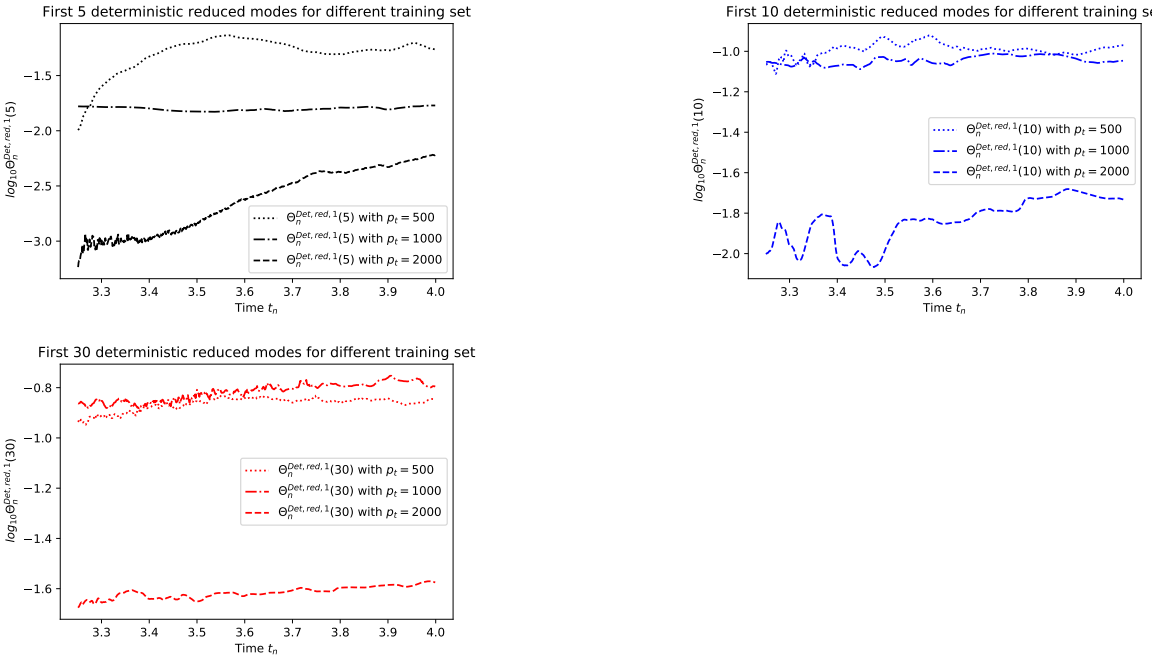


Figure III.15: First five, ten and thirty deterministic modes using different number of samples of the parametric training set.

b-Results for generating new stochastic modes

b-1-Frobenius errors between different schemes

In this part we take $p = 500$, $d_s = 800$ and $d_l = 8000$, we run the simulations in the interval $[3.5, 5]$ with a time step $\Delta t = 0.001$ and we approximate the solution with a rank $r = 50$ and recall $\overline{r} = \min(d, p)$. In Figure III.16, we plot the Frobenius error obtained with $d = d_s$ trajectories using the approximation $\overline{Y}_{Off,\overline{r}}^{DO_s,n}$ given by the algorithm (6) and the approximation $\overline{Y}_{Off,r}^{DO_s,n}$ (in green) and we plot the Frobenius errors obtained in the offline phase, step (3), with $d = d_l$, between the approximations $\overline{Y}_{Off,\overline{r}}^{DO_l,n}$ using algorithm (6) (in red), $\tilde{Y}_{Off,r}^{red_{d_l},n}$ using algorithm (16) (in black) and $\hat{Y}_{Off,r}^{red_{d_l},n}$ using algorithm (17) (in blue) with $d = d_l$ trajectories. Remark that we have quite the same order between different methods with an enhancement of the computational time that we discuss in the last part.

$$\epsilon_{n,d_l}(r) := \left\| \overline{Y}_{Off,\overline{r}}^{DO_l,n} - \overline{Y}_{Off,r}^{DO_l,n} \right\|_F \quad , \quad \epsilon_{n,d_s}(r) := \left\| \overline{Y}_{Off,\overline{r}}^{DO_s,n} - \overline{Y}_{Off,r}^{DO_s,n} \right\|_F$$

and

$$\epsilon^{red,1}_{n,d_l}(r) := \left\| \overline{Y}^{DO_l,n}_{Off,\bar{r}} - \tilde{Y}^{red_{d_l},n}_{Off,r} \right\|_F \quad , \quad \epsilon^{red,2}_{n,d_l}(r) := \left\| \overline{Y}^{DO_l,n}_{Off,\bar{r}} - \hat{Y}^{red_{d_l},n}_{Off,r} \right\|_F$$

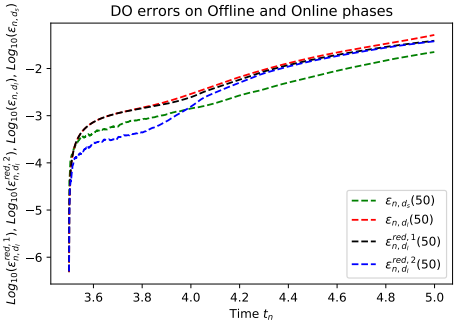
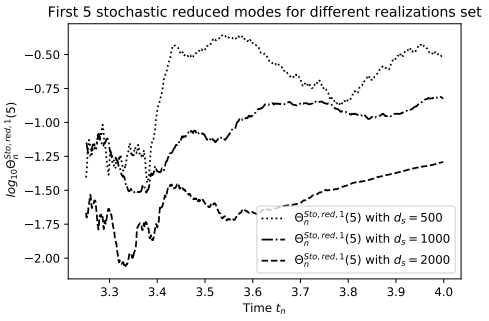


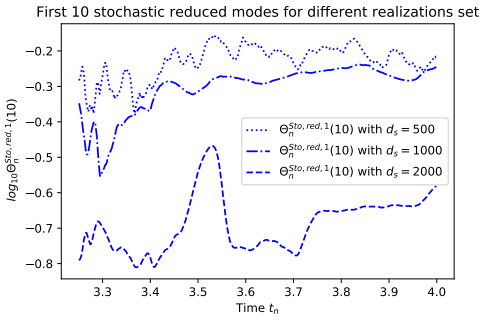
Figure III.16: Dynamical reduced order errors for both algorithms in step (3) of algorithm (20).

b-2-Effect of increasing the number of realizations d_s on the stochastic modes

Here we change the cardinality of the realizations set $d_s = 500$, $d_s = 1000$ and $d_s = 2000$, and we look at the effect on the stochastic modes given by the reduced model on a new set of realizations of cardinality $d_l = 500$, $d_l = 1000$ and $d_l = 2000$ respectively. We remark, in Figure III.17, the same effect as for deterministic modes, that more we sample the realization set and better is the approximation between the stochastic modes. Let be

$$\Theta^{Sto,red,1}_n(\hat{r}) := \left\| U^n_{:\hat{r}} - \tilde{U}^n_{Off,:\hat{r}} \right\|_F \quad , \quad \Theta^{Sto,red,2}_n(\hat{r}) := \left\| U^n_{:\hat{r}} - \hat{U}^n_{Off,:\hat{r}} \right\|_F$$





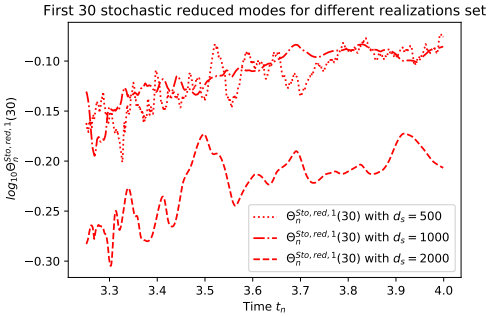


Figure III.17: First five, ten and thirty stochastic modes, by the first fixed deterministic modes algorithm (16), using different number of sample of the realization set.

These results motivate us to use the Monte Carlo estimators on the approximations given by the Control Variate algorithm (20) to approximate the expectation $\mathbb{E}[(\overline{X}^n)_j]$ for $0 \leq n \leq N$ and $1 \leq j \leq p$.

III.6.5 Some results on the online phase

1-Frobenius errors in the offline and online phases

Here we take $d_l = 10^4$, $d_s = 500$, $p_t = 200$ and $p_e = 200$ with a time step $\Delta t = 10^{-3}$. Let be, $\tilde{Y}_{On,r}^{red_{p_e},n}$ ($\hat{Y}_{On,r}^{red_{p_e},n}$ respectively) the solution given in step (4) using a new set of parameters $\mathcal{P}_{test}$ of cardinality p_e, and let $\overline{Y}_{\overline{r}}^{DO_{p_e},n}$ the solution given by the projector splitting algorithm (6) on $\mathcal{P}_{test}$.

Let us denote by,

$$\epsilon_{n,d_l}^{red_1,Off}(r) := \left\| \overline{Y}_{Off,\overline{r}}^{DO_l,n} - \tilde{Y}_{Off,r}^{red_{d_l},n} \right\|_F \quad , \quad \epsilon_{n,p_e}^{red_1,On}(r) := \left\| \overline{Y}_{\overline{r}}^{DO_{p_e},n} - \tilde{Y}_{On,r}^{red_{p_e},n} \right\|_F$$

$$\epsilon_{n,d_l}^{red_2,Off}(r) := \left\| \overline{Y}_{Off,\overline{r}}^{DO_l,n} - \hat{Y}_{Off,r}^{red_{d_l},n} \right\|_F \quad , \quad \epsilon_{n,p_e}^{red_2,On}(r) := \left\| \overline{Y}_{\overline{r}}^{DO_{p_e},n} - \hat{Y}_{On,r}^{red_{p_e},n} \right\|_F$$

In Figure III.18 are plotted 4 curves where the error $\epsilon_{n,p_t}(30)$ represents the dynamical error in step (1), the error $\bar{\epsilon}_{n,p_t}(30)$ represents the dynamical error in step (2) using the reduced algorithm (18), the error $\epsilon_{n,d_l}^{red,Off}(30)$ represents the dynamical error in step (3) using the reduced algorithm (16) and finally the error $\epsilon_{n,p_e}^{red,On}(30)$ represents the dynamical error in step (4) using the reduced algorithm (18). We remark that we have almost the same order between different steps showing that the algorithm works well.

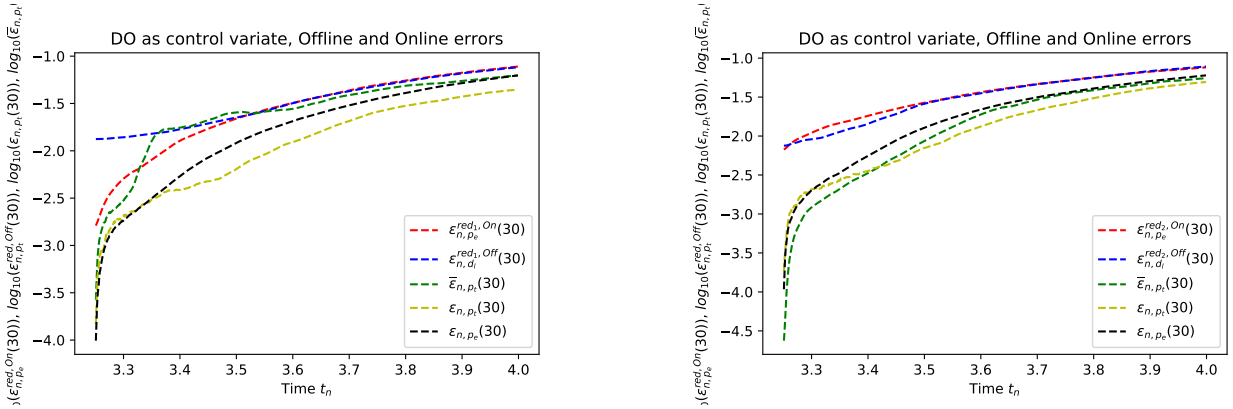


Figure III.18: Offline and Online errors for the DO as control variate algorithm (20). On the left, using the first reduced algorithm at each step and on the right using the second reduced algorithm at each step.

2-Relative error on the expectation between different schemes

Here we evaluate the relative error between estimators (III.31) and (III.32) compared to the standard estimator of Monte Carlo,

$$\mathbb{E}_{d_l}[(\overline{X}^n)_j] = \frac{1}{d_l} \sum_{i=1}^{d_l} (\overline{X}^n)_{i,j} \quad 0 \leq n \leq N \text{ and } 1 \leq j \leq p. \tag{III.30}$$

We use a number of parameter sample $p = 500$ and a number of realization $d_l = 8.10^4$. Let the approximations, $\overline{X}^n$ given by (III.16), $\overline{Y}_r^{DO_l,n}$ by algorithm (6), $\tilde{Y}_{On,r}^{red_{pe},n}$ by algorithm (20) using first reduced algorithms in each step and $\hat{Y}_{On,r}^{red_{pe},n}$ by algorithm (20) using second reduced algorithms in each step. Recall that here we use for each method different brownian realizations. Let the following Monte-Carlo estimators,

$$\mathbb{E}_{d_l}[(\tilde{Y}_{On,r}^{red_{pe},n})_j] = \frac{1}{d_l}\sum_{i=1}^{d_l}(\tilde{Y}_{On,r}^{red_{pe},n})_{i,j} \quad 0 \leq n \leq N \text{ and } 1 \leq j \leq p, \tag{III.31}$$

and,

$$\mathbb{E}_{d_l}[(\hat{Y}_{On,r}^{red_{pe},n})_j] = \frac{1}{d_l}\sum_{i=1}^{d_l}(\hat{Y}_{On,r}^{red_{pe},n})_{i,j} \quad 0 \leq n \leq N \text{ and } 1 \leq j \leq p, \tag{III.32}$$

that we use to approximate the expectation $\mathbb{E}_{d_l}[(\overline{X}^n)_j]$ for $0 \leq n \leq N$ and $1 \leq j \leq p$. Let be,

$$\Lambda_j^{DO}(r) = \frac{\mathbb{E}_{d_l}[(\overline{X}^n)_j] - \mathbb{E}_{d_l}[(\overline{Y}_r^{DO_l,n})_j]}{\mathbb{E}_{d_l}[(\overline{X}^n)_j]}, \qquad \Lambda_j^{red,1}(r) = \frac{\mathbb{E}_{d_l}[(\overline{X}^n)_j] - \mathbb{E}_{d_l}[(\tilde{Y}_{On,r}^{red_{pe},n})_j]}{\mathbb{E}_{d_l}[(\overline{X}^n)_j]},$$

and

$$\Lambda_j^{red,2}(r) = \frac{\mathbb{E}_{d_l}[(\overline{X}^n)_j] - \mathbb{E}_{d_l}[(\hat{Y}_{On,r}^{red_{pe},n})_j]}{\mathbb{E}_{d_l}[(\overline{X}^n)_j]}.$$

In Figure III.19 we plot the relative error for a given parameter $j = 310$ that coincides with $\mu = (0.7, 1.3)$. Note that we have obtained statistically the same Figure for all the parameters with a maximum of error that reaches 0.8 per cent. This result prove numerically that the estimators (III.31) and (III.32) are as good as the standard estimator (III.30).

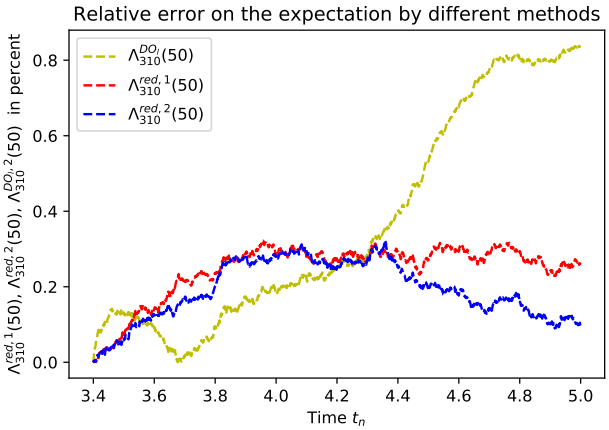


Figure III.19: Relative error (in percent) $\Lambda_{310}^{DO}(50)$, $\Lambda_{310}^{red,1}(50)$ and $\Lambda_{310}^{red,2}(50)$.

In the next part we enhance this result by showing the low computational time needed for these methods compared to the standard one.

3-Computational time between different schemes

In the Figure III.20 we plot the computational time needed to simulate different methods on the time $[3.5, 3.7]$, using a fixed number of realizations $d = 8.10^4$ and varying only the number of parameters p from $p = 500$ to $p = 25.10^3$ using the following points for $p = [0, 5.10^3; 10^3; 5.10^3; 10^4; 1, 5.10^4; 2.10^4; 2, 5.10^4]$. Remark that the logarithmic plot of the computational time shows that the methods projector splitting and both reduced methods start to be benefit for all variants from a number of parameters p around $p = 3000$ compared to the Euler Maruyama scheme. The growth is exponential and we see that we can reach an improvement of the computational time, in the online phase, by a factor of 20 between the approximation given by the Control Variate algorithm (20) using the second reduced schemes and the approximation given by the Euler Maruyama scheme.

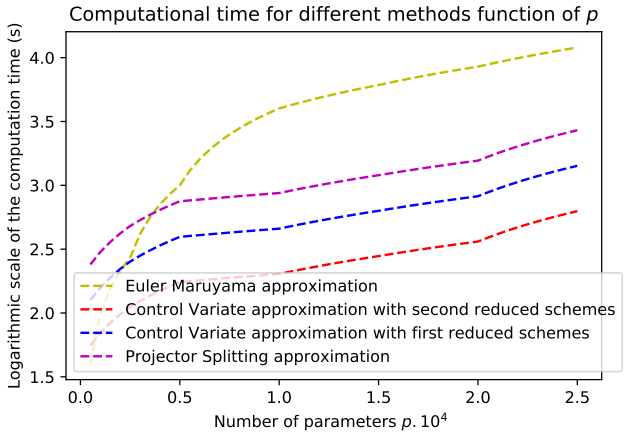


Figure III.20: Computational time for methods: Control Variate by the online phase in algorithm (20) using first algorithms, Control Variate by the online phase (20) using second algorithms, Projector Splitting by algorithm (6) and Euler Maruyama algorithm (III.16) as function of the number of parameters p and with fixed number of realizations $d = 8.10^4$.

CHAPTER IV

ANNEXES

Effect of random sampling on the projector splitting method

In this part, we study the variability of the projector splitting method with respect to the realizations of the stochastic noise and to the parameter sampling. We consider a time interval $[3.2, 4]$ with $p = 900$ and $d = 10^4$ and with a time step $\Delta t = 0.001$. The projector-splittig method is run with $r = 30$. Let the dynamical orthogonal solution written as,

$$\overline{Y}_{30}^{DO,k,n} = U_{30}^{DO,k,n} S_{30}^{DO,k,n} \left(V_{30}^{DO,k,n}\right)^T,$$

for each k test case. We want to compare the effect of changing the set of parameters on the stochastic modes and inversely the effect of changing the realizations on the deterministic modes. Let be $\Theta_n^{Sto,k,l}(\hat{r})$ and $\Theta_n^{Det,k,l}(\hat{r})$ the Frobenius errors between the first $\hat{r}$ stochastic modes (deterministic modes respectively) of the test k and the test l,

$$\Theta_n^{Sto,k,l}(\hat{r}) := \left\| U_{:\hat{r}}^{DO,k,n} - U_{:\hat{r}}^{DO,l,n} \right\|_F \quad , \quad \Theta_n^{Det,k,l}(\hat{r}) := \left\| V_{:\hat{r}}^{DO,k,n} - V_{:\hat{r}}^{DO,l,n} \right\|_F$$

Where $U_{:\hat{r}}^{DO,k,n} \in \mathbb{R}^{d,\hat{r}}$ is the matrice of the first $\hat{r}$ stochastic modes and $V_{:\hat{r}}^{DO,k,n} \in \mathbb{R}^{p,\hat{r}}$ is the matrice of the first $\hat{r}$ deterministic modes.

In Figure IV.1 we plot the Frobenius error between different stochastic modes given by three simulations where we only have changed the set of parameters and we have kept the set of realizations the same for the three. We obtain a good matching for the first modes, and we loose this matching as we compare more modes. In Figure IV.2 we plot five stochastic modes obtained by changing at each time the set of parameters. In Figure IV.3 we plot the Frobenius error between deterministic modes obtained using different realizations and with the same set of parameters. We obtain a very good matching between all modes that stays under $10^{-1.4}$ for all the deterministic modes.

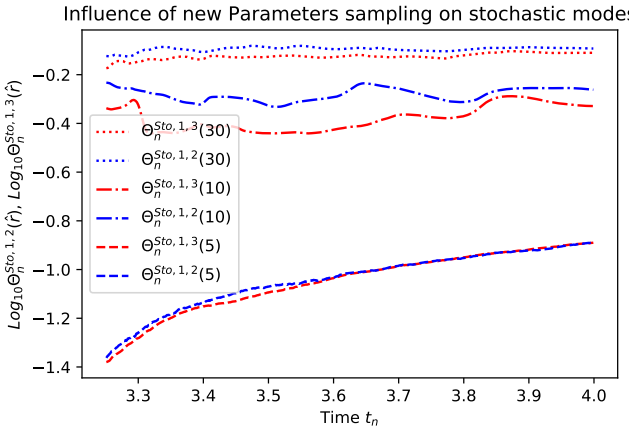


Figure IV.1: Frobenius error between stochastic modes obtained using different set of parameters.

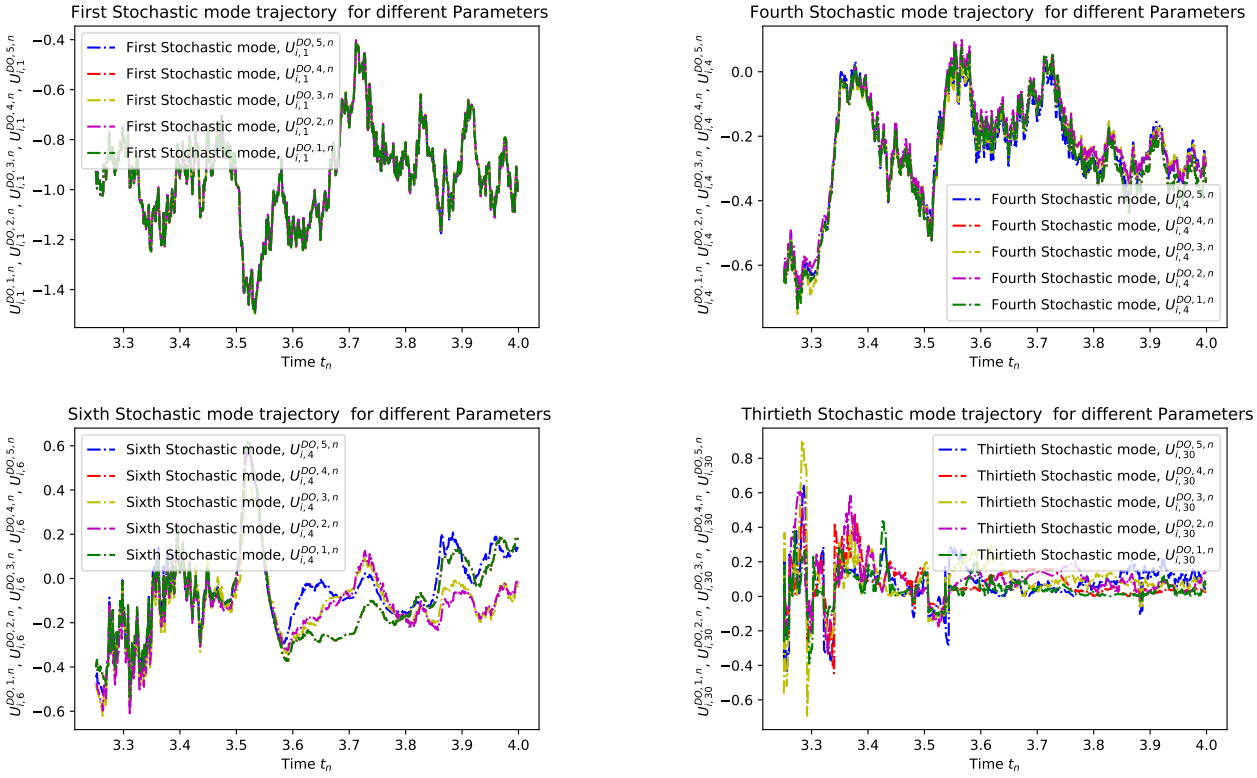


Figure IV.2: Stochastic Modes 1,4, 6 and 30 for different set of parameters

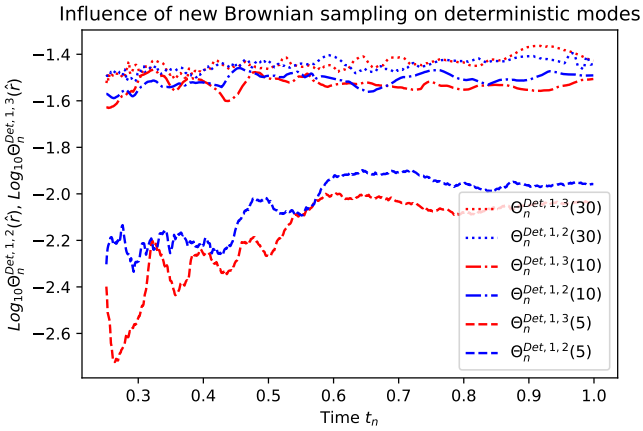


Figure IV.3: Frobenius error between deterministic modes obtained using different brownian sampling.

In Figure IV.4 are 7 curves, corresponding to 7 random realizations, where the minimal rank r_e so that $\epsilon_{\max}(r_e) \leq e$ with $e = 10^{-2}$ being a prescribed error tolerance, is plotted as a function of d the number of random realisations of the stochastic noise.

Remark that for a threshold $e = 10^{-2}$, the minimal rank r_e seems reach a value between $r = 70$ and $r = 100$ for high values of the number of random realizations d.

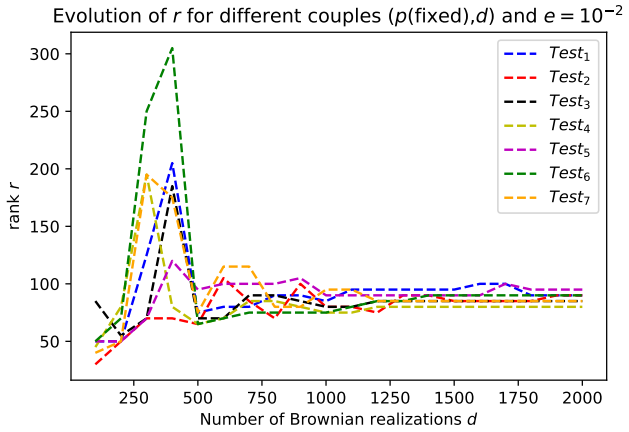


Figure IV.4: Evolution of r for different realizations with fixed parameters and with a fixed threshold $e = 10^{-2}$.

CHAPTER V

CONCLUSIONS AND PERSPECTIVES

In this thesis we have worked on two different subjects. First we have developed a theoretical analysis of a numerical method used to construct a control variate using reduced basis. The reduced basis is constructed via a Greedy algorithm where the norm is evaluated using a Monte Carlo estimator (Monte Carlo Greedy algorithm). We prove using concentration inequalities and under some conditions on the sampling number M_n at each iteration $n \in \mathbb{N}^*$, that with high probability, the Monte Carlo Greedy algorithm is a weak Greedy algorithm. Unfortunately, the theoretical obtained result could not be implemented as the lower bound on the sampling number is huge at each iteration n which implies a big increment on M_n. To escape this problem we developed a heuristic algorithm based on weakening the lower bound and replacing it by a condition inspired from the theoretical results. We apply this algorithm on three test cases. The obtained numerical results show a very good matching, in the offline phase, between several theoretical and numerical indicators as the decay of the distance between the constructed reduced basis and the manifold. In the online phase we show that for a fixed statistical error we need a sampling number $M = 10^6$ for the standard Monte Carlo estimator while we have needed only $M = 349$ for the Monte Carlo estimator on the control variate (result deduced from the first test case). A possible perspective would be to adapt the algorithm and the numerical analysis to the sampling by Markov Chains (Markov Chain Monte Carlo).
Second, we have developed different schemes to efficiently approximate the solution of a parametric stochastic differential equation in the additive and multiplicative noise case. The different schemes are inspired from the splitting method developed for parametrized differential equations. We check that these schemes are strongly consistent with order one with an Euler Maruyama discretization. We use these schemes to build a control variate to efficiently compute the expectation of (observables of) the parametric process, at each time step. The numerical results, on the additive case, show a reduction of the computational time up to 15 times compared to the computational time needed for the standard Monte Carlo estimator for the same sampling number under a relative error of about 1 per cent. Finally, various schemes are proposed to extend the previous results to a parametric stochastic differential equation including a McKean non-linear term. We again observe a significant gain of the computational time. For future work, let us mention that it would be interesting to extend the numerical analysis results which have been obtained in the literature for ODE to our setting of SDE.

BIBLIOGRAPHY

[1] Oleg Balabanov and Anthony Nouy. Randomized linear algebra for model reduction. part i: Galerkin methods and error estimation. *Advances in Computational Mathematics*, 45(5):2969–3019, 2019.

[2] Oleg Balabanov and Anthony Nouy. Randomized linear algebra for model reduction—part ii: minimal residual methods and dictionary-based approximation. *Advances in Computational Mathematics*, 47(2):1–54, 2021.

[3] Maxime Barrault, Yvon Maday, Ngoc Cuong Nguyen, and Anthony T Patera. An 'empirical interpolation'method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.

[4] Peter Binev, Albert Cohen, Wolfgang Dahmen, Ronald DeVore, Guergana Petrova, and Przemyslaw Wojtaszczyk. Convergence rates for greedy algorithms in reduced basis methods. *SIAM journal on mathematical analysis*, 43(3):1457–1472, 2011.

[5] François Bolley, Arnaud Guillin, and Cédric Villani. Quantitative concentration inequalities for empirical measures on non-compact spaces. *Probability Theory and Related Fields*, 137(3-4):541–593, 2007.

[6] Sébastien Boyaval, Tony Lelièvre, et al. A variance reduction method for parametrized stochastic differential equations using the reduced basis paradigm. *Communications in Mathematical Sciences*, 8(3):735–762, 2010.

[7] Annalisa Buffa, Yvon Maday, Anthony T Patera, Christophe Prud'homme, and Gabriel Turinici. A priori convergence of the greedy algorithm for the parametrized reduced basis method. *ESAIM: Mathematical modelling and numerical analysis*, 46(3):595–603, 2012.

[8] Gianluca Ceruti, Jonas Kusch, and Christian Lubich. A rank-adaptive robust integrator for dynamical low-rank approximation. *BIT Numerical Mathematics*, pages 1–26, 2022.

[9] Sayan Chakraborty, Souvik Chatterjee, Nilanjan Dey, Amira S Ashour, and Aboul Ella Hassanien. Comparative approach between singular value decomposition and randomized singular value decomposition-based watermarking. In *Intelligent techniques in signal processing for multimedia security*, pages 133–149. Springer, 2017.

[10] Albert Cohen, Wolfgang Dahmen, Ronald Devore, and James Nichols. Reduced basis greedy selection using random training sets. *ESAIM: Mathematical Modelling and Numerical Analysis*, 54(5):1509–1524, 2020.

[11] Ronald DeVore, Guergana Petrova, and Przemyslaw Wojtaszczyk. Greedy algorithms for reduced bases in banach spaces. *Constructive Approximation*, 37(3):455–466, 2013.

[12] Ronald A DeVore. Nonlinear approximation. *Acta numerica*, 7:51–150, 1998.

[13] Ronald A DeVore. The theoretical foundation of reduced basis methods. *Model Reduction and approximation: Theory and Algorithms*, pages 137–168, 2014.

[14] Virginie Ehrlacher, Damiano Lombardi, Olga Mula, and François-Xavier Vialard. Nonlinear model reduction on metric spaces. application to one-dimensional conservative pdes in wasserstein spaces. *ESAIM. Mathematical Modelling and Numerical Analysis*, 54, 2020.

[15] Feppon Florian and Lermusiaux Pierre. Dynamically orthogonal numerical schemes for efficient stochastic advection and lagrangian transport. *Society for Industrial and Applied Mathematics*, 60(3):595–625, 2018.

[16] Nicolas Fournier and Arnaud Guillin. On the rate of convergence in wasserstein distance of the empirical measure. *Probability Theory and Related Fields*, 162(3-4):707–738, 2015.

[17] Jan S Hesthaven, Gianluigi Rozza, Benjamin Stamm, et al. *Certified reduced basis methods for parametrized partial differential equations*, volume 590. Springer, 2016.

[18] Emil Kieri, Christian Lubich, and Hanna Walach. Discretized dynamical low-rank approximation in the presence of small singular values. *SIAM J. Numer. Anal.*, 54:1020–1038, 2016.

[19] Christian Lubich and Ivan V Oseledets. A projector-splitting integrator for dynamical low-rank approximation. *BIT Numerical Mathematics*, 54(1):171–188, 2014.

[20] Eleonora. Musharbash, Fabio. Nobile, and Tao. Zhou. Error analysis of the dynamically orthogonal approximation of time dependent random pdes. *SIAM Journal on Scientific Computing*, 37(2):A776–A810, 2015.

[21] Koch Othmar and Lubich Christian. Dynamical low rank approximation. *SIAM Journal on Matrix Analysis and Applications*, 29(2):434–454, 2007.

[22] Alfio Quarteroni, Andrea Manzoni, and Federico Negri. *Reduced basis methods for partial differential equations: an introduction*, volume 92. Springer, 2015.

[23] Bellman Richard. Dynamic programming. *Science*, 153(3731):34–37, 1966.

[24] DeVore Ronald, Ralph Howard, and Charles Micchelli. Optimal nonlinear approximation. *Manuscripta mathematica*, 63(4):469–478, 1989.

[25] Kathrin Smetana and Olivier Zahm. Randomized residual-based error estimators for the Proper Generalized Decomposition approximation of parametrized problems. *International Journal for Numerical Methods in Engineering*, 121(23):5153–5177, December 2020.

[26] Kathrin Smetana, Olivier Zahm, and Anthony T Patera. Randomized residual-based error estimators for parametrized equations. *SIAM Journal on Scientific Computing*, 41(2):A900–A926, March 2019.