Analyse de sensibilité et réduction de dimension. Application à l'océanographie

Alexandre Janon

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Mathématiques Appliquées**

Arrêté ministériel : 7 août 2006

Présentée par

Alexandre JANON

Thèse dirigée par **Clémentine PRIEUR**
et codirigée par **Maëlle NODET**

préparée au sein **du Laboratoire Jean Kuntzmann**
et de **l'École Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique**

Analyse de sensibilité et réduction de dimension
Application à l'océanographie

Thèse soutenue publiquement le **15 novembre 2012**,
devant le jury composé de :

M. Josselin GARNIER
Professeur, Université Paris 7, Président
Mme Béatrice LAURENT
Professeur, INSA Toulouse, Rapporteur
M. Yvon MADAY
Professeur, Université Paris 6, Rapporteur
M. Christophe PRUDHOMME
Professeur, Université de Strasbourg, Examinateur
M. Anthony NOUY
Professeur, École Centrale de Nantes, Examinateur
Mme Florence FORBES
Directeur de Recherche, INRIA Rhône Alpes, Examinatrice
Mme Clémentine PRIEUR
Professeur, Université Grenoble 1, Directrice de thèse
Mme Maëlle NODET
Maître de Conférences, Université Grenoble 1, Co-Directrice de thèse

Remerciements

Je souhaite commencer en remerciant mes deux directrices de thèse, Clémentine Prieur et Maëlle Nodet, d'abord pour m'avoir proposé de travailler sur ce sujet intéressant et à l'intersection de deux thèmes qui m'intéressent, et ensuite et surtout pour leur encadrement stimulant, rigoureux et sans faille durant ces trois années de collaboration.
Je remercie également les membres du jury, qui ont accepté de s'intéresser à mon travail, et qui ont tous apporté des remarques pertinentes, aussi bien lors des pré-rapports que de la soutenance.
Mes pensées vont à ma famille : ma mère (sans qui rien de tout cela n'aurait été possible !), mon oncle et ma tante, mon cousin (qui sait faire péter les bouchons de champagne sans blesser personne !) et ma cousine, première docteur de la famille !
Je remercie évidemment les membres du LJK que j'ai côtoyé : Anne, pour sa maîtrise des rouages administratifs, et bien sûr les autres doctorants (en espérant n'oublier personne !) : les cobureaux de la salle 3 (Jean-Yves, Chloé, Jean-Matthieu, Amin, Kole), ou d'ailleurs (Gaëlle, Bertrand, Vincent, Federico, Madison, Lukas, Roland, Meryam, $\theta_{t,\tau}$ – facteur essentiel de la réussite de ma semaine belge –, Jonathan, Gildas, Christine). J'ai grandement apprécié les relations amicales que nous avons eu ensemble et je vous souhaite beaucoup de réussite pour la suite.

Résumé

Les modèles mathématiques ont pour but de décrire le comportement d'un système. Bien souvent, cette description est imparfaite, notamment en raison des incertitudes sur les paramètres qui définissent le modèle. Dans le contexte de la modélisation des fluides géophysiques, ces paramètres peuvent être par exemple la géométrie du domaine, l'état initial, le forçage par le vent, ou les coefficients de frottement ou de viscosité.

L'objet de l'analyse de sensibilité est de mesurer l'impact de l'incertitude attachée à chaque paramètre d'entrée sur la solution du modèle, et, plus particulièrement, identifier les paramètres (ou groupes de paramètres) « sensibles ». Parmi les différentes méthodes d'analyse de sensibilité, nous privilégierons la méthode reposant sur le calcul des indices de sensibilité de Sobol.

Le calcul numérique de ces indices de Sobol nécessite l'obtention des solutions numériques du modèle pour un grand nombre d'instances des paramètres d'entrée. Cependant, dans de nombreux contextes, dont celui des modèles géophysiques, chaque lancement du modèle peut nécessiter un temps de calcul important, ce qui rend inenvisageable, ou tout au moins peu pratique, d'effectuer le nombre de lancements suffisant pour estimer les indices de Sobol avec la précision désirée.

Ceci amène à remplacer le modèle initial par un *métamodèle* (aussi appelé *surface de réponse* ou *modèle de substitution*). Il s'agit d'un modèle approchant le modèle numérique de départ, qui nécessite un temps de calcul par lancement nettement diminué par rapport au modèle original.

Cette thèse se centre sur l'utilisation d'un métamodèle dans le cadre du calcul des indices de Sobol, plus particulièrement sur la quantification de l'impact du remplacement du modèle par un métamodèle en terme d'erreur d'estimation des indices de Sobol. Nous nous intéressons également à une méthode de construction d'un métamodèle efficace et rigoureux pouvant être

utilisé dans le contexte géophysique.

Summary

Mathematical models seldom represent perfectly the reality of studied systems, due to, for instance, uncertainties on the parameters that define the system. In the context of geophysical fluids modelling, these parameters can be, e.g., the domain geometry, the initial state, the wind stress, the friction or viscosity coefficients.
Sensitivity analysis aims at measuring the impact of each input parameter uncertainty on the model solution and, more specifically, to identify the "sensitive" parameters (or groups of parameters). Amongst the sensitivity analysis methods, we will focus on the Sobol indices method.
The numerical computation of these indices require numerical solutions of the model for a large number of parameters' instances. However, many models (such as typical geophysical fluid models) require a large amount of computational time just to perform one run. In these cases, it is impossible (or at least not practical) to perform the number of runs required to estimate Sobol indices with the required precision.
This leads to the replacement of the initial model by a *metamodel* (also called *response surface* or *surrogate model*), which is a model that approximates the original model, while having a significantly smaller time per run, compared to the original model.
This thesis focuses on the use of metamodel to compute Sobol indices. More specifically, our main topic is the quantification of the metamodeling impact, in terms of Sobol indices estimation error. We also consider a method of metamodeling which leads to an efficient and rigorous metamodel, which can be used in the geophysical context.

Table des matières

Liste des travaux

Les chapitres 2, 3, 4 et 5 de cette thèse sont constitués des publications, ou prépublications, suivantes :

Janon1 Janon, A., Nodet M., Prieur C. Uncertainties assessment in global sensitivity indices estimation from metamodels, 2011. Accepté dans International Journal for Uncertainty Quantification.

Janon2 Janon, A., Klein T., Lagnoux A., Nodet M., Prieur C. Asymptotic normality and efficiency of two Sobol index estimators, 2012, soumis.

Janon3 Janon, A., Nodet M., Prieur C. Certified reduced basis solutions of viscous Burgers equation parametrized by initial and boundary values, 2010. Accepté dans Mathematical Modelling and Numerical Analysis.

Janon4 Janon A., Nodet M., Prieur C. Goal-oriented error estimation for reduced basis method, with application to certified sensitivity analysis, 2012, soumis.

[Janon1] constitue le chapitre 2, [Janon2] le chapitre 3, [Janon3] le chapitre 4 et [Janon4] le chapitre 5.
Deux rapports techniques, disponibles sur la page Web de l'auteur[1] ont également été rédigés et sont des synthèses bibliographiques sur des thèmes en rapport avec l'objet de cette thèse :

Janon5 Janon, A. Reduced-basis solutions of affinely-parametrized linear partial differential equations, 2010.

Janon6 Janon, A. Méthodes de quadrature multi-dimensionnelles et décomposition en polynômes de chaos. Application à la quantification d'incertitude et aux méthodes stochastiques d'analyse de sensibilité. 2010.

1. http://ljk.imag.fr/membres/Alexandre.Janon/

Enfin, les algorithmes présentés dans les chapitres 2 et 3 ont été implémentés informatiquement par l'auteur, et sont distribués au public sous forme de contributions (sous licence libre) aux packages `CompModSA` [116] et `sensitivity` [77] du logiciel R [81].
Nous donnons ci-dessous une courte description de ces packages ainsi que de notre contribution.

Package `CompModSA` : Ce package implémente de nombreuses méthodes de métamodélisation « boîte noire » (utilisant seulement un échantillon fini d'évaluations du modèle à approcher). Ces méthodes de métamodélisation sont ensuites utilisées par les estimateurs d'indices de sensibilité, inclus dans le package. La précision de l'estimation est donnée sous forme d'intervalles de confiance obtenus par *bootstrap*, en suivant la méthode décrite dans [100].

Notre contribution : Nous avons corrigé plusieurs *bugs* et incompatibilités avec les versions récentes des packages utilisés. Nous avons également rajouté une option afin de permettre l'estimation des indices de Sobol du premier ordre (auparavant, seuls les indices totaux étaient proposés). Enfin, nous avons rendu le package propre à être diffusé sur le CRAN (collection en ligne de packages R).

URL : `http://cran.r-project.org/web/packages/sensitivity/`

Package `sensitivity` : Ce package contient plusieurs méthodes d'estimation de mesures de sensibilité, dont les indices de sensibilité de Sobol. L'estimation ponctuelle est complétée par un intervalle de confiance bootstrap.

Notre contribution : Nous avons ajouté à ce package l'estimateur asymptotiquement efficace que nous considérons dans le chapitre 3 de ce manuscrit, ainsi que l'estimation des intervalles de confiance asymptotiques pour les deux estimateurs considérés dans ce chapitre. Par ailleurs, nous avons implémenté – en utilisant le langage C++ interfacé avec R – les deux méthodes d'analyse de sensibilité certifiées présentées au chapitre 2.

URL : `http://cran.r-project.org/web/packages/CompModSA/`

Chapitre 1

Introduction

1.1 Objet de la thèse

Cette thèse se place dans le contexte de la modélisation numérique du comportement des fluides géophysiques. Cette modélisation joue un rôle fondamental dans les systèmes de prévision environnementaux, que ce soit en météorologie, en océanographie, dans l'étude des fleuves et des crues, ou en climatologie.

Issus de considérations physiques, les modèles utilisés prennent la forme d'équations, ou de systèmes d'équations aux dérivées partielles (EDP) qui ne peuvent représenter la réalité qu'imparfaitement, à cause des sources d'incertitudes suivantes :

- les équations obtenues par la physique sont souvent établies en faisant des hypothèses simplificatrices (des considérations d'échelle spatiale ou d'échelle temporelle amènent à négliger l'effet de certaines forces devant d'autres), ainsi le modèle mathématique lui-même ne traduit pas l'intégralité du phénomène physique étudié ;
- les solutions du modèle ne sont bien souvent accessibles qu'au travers d'une solution calculée numériquement, qui est une approximation de la « vraie » solution des équations issues de la physique ;
- les équations du modèle – et donc les solutions du modèle – dépendent des paramètres physiques du système étudié (géométrie du domaine, état initial, état au bord du domaine, forçage par le vent, coefficients de frottement, de viscosité...) qui sont souvent mal connus.

Notre travail se centre sur cette troisième source d'incertitude. Nous souhaitons mesurer l'impact de l'incertitude attachée aux paramètres d'entrée sur la solution, et, plus particulièrement, identifier les paramètres (ou groupes de paramètres) « sensibles ». Cette identification, qui peut se faire à l'aide de plusieurs méthodes, est l'objet de l'analyse de sensibilité. Les méthodes d'analyse de sensibilité peuvent se classer en deux types :

- Dans les méthodes *locales*, l'influence d'un paramètre est mesurée par l'importance de la variation de la solution autour d'une valeur nominale de ce paramètre, c'est-à-dire par une dérivée partielle évaluée en cette valeur nominale.
- Dans les méthodes *globales* (aussi appelées méthodes *stochastiques*), l'incertitude sur les paramètres est modélisée par une loi de probabilité, connue *a priori*. Cette loi sur les entrées ainsi que le modèle déterminent entièrement la distribution de la solution du modèle, et celle-ci contient toute l'information sur l'incertitude de la sortie. Des résumés statistiques bien choisis de cette distribution permettent alors de classer les paramètres par ordre d'importance.

Parmi les différentes méthodes d'analyse de sensibilité, nous privilégierons la méthode, globale, reposant sur le calcul des indices de sensibilité de Sobol. L'indice de Sobol d'un paramètre (ou d'un groupe de paramètres) est un indicateur statistique, d'interprétation aisée, de l'importance de ce paramètre (ou de ce groupe de paramètres) sur la variabilité d'une quantité scalaire d'intérêt, fonction de la solution du modèle. Le calcul effectif des indices de sensibilité permet de hiérarchiser les paramètres d'entrée en fonction de leur influence sur la sortie. Un utilisateur du modèle peut alors identifier les paramètres les plus influents comme ceux sur lesquels l'incertitude doit être réduite – dans la mesure du possible – en priorité afin d'apporter une réduction de l'incertitude sur la sortie la plus importante. À l'inverse, les paramètres les moins influents peuvent être fixés à une valeur nominale, ce qui permet de simplifier le modèle. Par ailleurs, les experts utilisateurs du modèle ont bien souvent une idée qualitative, *a priori*, basée sur leur intuition, des paramètres les plus influents d'un système physique. La confrontation de cette préconception avec les indices calculés numériquement permet de valider, ou au contraire d'invalider, le modèle ou son implémentation informatique.

Les indices de Sobol peuvent se calculer de diverses façons, et nous ferons le

choix de nous centrer sur les méthodes dites de Monte-Carlo. Ces méthodes, d'origine stochastique, fournissent une approximation numérique des indices de Sobol entâchée d'une erreur d'*estimation*, ou *erreur Monte-Carlo*. L'utilisation d'une telle méthode d'approximation pose deux questions :

Question 1 : Dans le but de certifier cette approximation, comment évaluer l'erreur Monte-Carlo ?

Question 2 : Plusieurs formules d'approximation Monte-Carlo étant utilisables, peut-on identifier une méthode qui soit « la plus précise », en un sens à formaliser ?

Une autre problématique associée au calcul des indices de Sobol est que celui-ci nécessite l'obtention des solutions numériques du modèle pour un grand nombre d'instances des paramètres d'entrée. Cependant, dans de nombreux contextes, dont celui des modèles géophysiques, chaque lancement du modèle peut nécessiter un temps de calcul important, ce qui rend inenvisageable, ou tout au moins peu pratique, d'effectuer le nombre de lancements suffisant pour estimer les indices de Sobol avec la précision désirée.
C'est ici qu'intervient l'utilisation d'un *métamodèle* (aussi appelé *surface de réponse* ou *modèle de substitution*). Il s'agit d'un modèle, construit algorithmiquement, approchant le modèle numérique de départ, qui nécessite un temps de calcul par lancement nettement diminué par rapport au modèle original. L'utilisation d'un métamodèle sépare le calcul des solutions du modèle en deux phases :

- une phase de préparation, ou phase *offline*, durant laquelle le métamodèle est construit ;
- une phase d'utilisation, ou phase *online*, pendant laquelle le métamodèle est appelé pour chaque valeur des paramètres d'entrée sur laquelle on souhaite évaluer le modèle.

La phase de préparation pouvant être coûteuse en temps de calcul, il est bien sûr judicieux d'utiliser un métamodèle seulement si le nombre de lancements à effectuer dépasse un certain seuil. Remarquons que l'utilisation de métamodèles n'est pas cantonnée à l'analyse de sensibilité, elle trouve également sa place dans d'autres domaines où un même modèle doit être résolu pour un grand nombre de paramètres différents, tel que l'optimisation, ou dans des applications (telles que la météo ou la simulation haptique) où la réponse du modèle doit être connue en temps réel par l'utilisateur.

Parmi les méthodes de construction d'un métamodèle, les méthodes de réduction de dimension visent à identifier, durant la phase de préparation, les degrés de liberté pertinents pour décrire, de manière suffisamment précise, la dépendance de la solution du modèle en fonction des paramètres d'entrée. Le but est de remplacer le système d'équations issu de la discrétisation numérique de l'EDP originale, qui est coûteux à résoudre car faisant intervenir plusieurs milliers (voire quelques millions) d'inconnues, par un système d'équations donnant les coordonnées de la solution approchée du modèle dans le nouveau système de degrés de libertés. Dans de nombreux cas, on peut remplacer le système à plusieurs milliers d'inconnues par un système avec un nombre d'inconnues de l'ordre de la dizaine, ce qui procure des gains substantiels en temps de calcul.
L'utilisation d'un métamodèle amène trois nouvelles questions :

Question 3 : Le remplacement du modèle original par le métamodèle est une source d'erreur lors de l'estimation des indices. Comment quantifier l'impact de cette erreur sur les indices de Sobol calculés ?

Question 4 : Une façon de quantifier cette erreur est de disposer d'une méthode de métamodélisation fournissant une majoration de l'erreur entre le modèle et le métamodèle. Comment développer de telles méthodes *certifiées*, utilisables dans les modèles géophysiques qui nous intéressent ?

Question 5 : L'analyse de sensibilité s'intéresse souvent à un aspect particulier du modèle (par exemple un résumé statistique de celui-ci). On peut vouloir construire le métamodèle en cherchant à ce qu'il reproduise le plus fidèlement possible cet aspect d'intérêt. La borne d'erreur fournie par le métamodèle, évoquée à la question précédente, peut-elle également être optimisée en fonction de l'aspect retenu du modèle ?

La suite de ce manuscrit est consacrée d'une part à la présentation détaillée de plusieurs méthodes d'analyse de sensibilité globale, et d'autre part à la réponse que nous proposons aux cinq questions soulevées plus haut.
Plus spécifiquement, la section 1.2 présente quelques méthodes d'analyse de sensibilité, en se centrant sur les indices de Sobol et sur leur estimation, la section 1.3 détaille trois méthodes de métamodélisation, et les sections suivantes sont des introductions aux chapitres 2, 3, 4 et 5 de la thèse :
– la section 1.4 introduit le chapitre 2, qui traite des questions 1 et 3 ;
– la section 1.5 introduit le chapitre 3, qui traite des questions 1, 2 et 3 ;

– la section 1.6 introduit le chapitre 4, qui traite de la question 4 ;
– la section 1.7 introduit le chapitre 5, qui traite de la question 5.

Ces chapitres sont constitués de publications soumises ou acceptées, dont on trouvera les références page 13.

1.2 Introduction à l'analyse de sensibilité globale

L'objet de cette section est de présenter différentes méthodes d'analyse de sensibilité globale. La première partie introduit le cadre d'étude et les notations. La partie 1.2.2 définit plusieurs mesures de sensibilité, dont les indices de Sobol. Des méthodes de calcul numérique approché de ces indices sont décrites en partie 1.2.3. Nous terminons par la partie 1.2.4, où nous donnons des méthodes d'estimation de l'erreur d'approximation dans l'évaluation des indices de Sobol par Monte-Carlo. Les références classiques en analyse de sensibilité globale sont [88], [87], [91], [89], [28], [93], [35, chapitre 6], [60] et [15].

1.2.1 Cadre stochastique

On désigne par Y la *sortie* d'un modèle (appelée aussi *quantité d'intérêt*), fonction (à valeurs réelles) f de p variables (réelles) d'entrée notées $X_1, \ldots, X_p$:

$$Y = f(X_1, \ldots, X_p).$$

Dans le contexte des modèles de fluides géophysiques, considérons l'état du fluide constitué par exemple, en océanographie ou en hydrographie, du champ de vitesses $\vec{u}$ et la hauteur d'eau h : $\mathbf{u} = (\vec{u}, h)$) à un instant $t \in [0; T]$. Supposons que cet état soit donné par une fonction $\mathbf{u} = \mathbf{u}(x, t, X_1, \ldots, X_p)$ où x désigne la variable d'espace (x appartient à un domaine Ω, sous-ensemble compact connexe de $\mathbb{R}^2$ ou de $\mathbb{R}^3$) et $X_1, \ldots, X_p$ sont les paramètres d'entrée (coefficients de frottement, de viscosité, de viscosité numérique, ainsi que les paramètres régissant la force exercée par le vent en surface, l'état initial, l'état au bord, ou la géométrie du domaine Ω). Cette fonction $\mathbf{u}$ est donnée comme solution d'une équation aux dérivées partielles, dépendant des paramètres d'entrée, et traduisant la physique du phénomène étudié (par exemple, Navier-Stokes ou Saint-Venant). La quantité d'intérêt

en sortie du modèle est alors fonction de la variable d'état :

$$Y = f(X_1, \ldots, X_p) = f(\mathbf{u}(\cdot, \cdot, X_1, \ldots, X_p)).$$

Par exemple, Y peut être la hauteur d'eau moyenne ($|\Omega|$ désigne l'aire ou le volume de Ω) :

$$Y = \frac{1}{T\,|\Omega|} \int_0^T \int_\Omega h(x, t, X_1, \ldots, X_p) \mathrm{d}x \mathrm{d}t,$$

ou une quantité plus complexe (car non linéaire en fonction de $\mathbf{u}$), telle que le premier instant où la hauteur d'eau dépasse un seuil s fixé :

$$Y = \underset{t \in [0;T]}{\operatorname{arginf}} \{\exists x \in \Omega \text{ tel que } u(t, x, X_1, \ldots, X_p) \geq s\}.$$

Comme précisé en introduction, les paramètres d'entrée du modèle sont imparfaitement connus. Pour modéliser cette incertitude sur les entrées, on affecte une loi de probabilité (connue) au p-uplet de paramètres $(X_1, \ldots, X_p)$. Cette loi est déterminée par l'utilisateur, en fonction de sa connaissance sur l'incertitude associée à chaque paramètre. Par exemple, un paramètre continu peut suivre une loi uniforme sur un intervalle connu, ou une loi normale de moyenne et d'écart-type connus. En termes probabilistes, ceci revient à considérer que $(X_1, \ldots, X_p)$ est un vecteur aléatoire. L'incertitude sur ces entrées se « propage » alors sur la sortie par le biais de la fonction f, et Y est également vue comme une variable aléatoire, dont la loi (a priori inconnue) concentre toute l'information sur l'impact de l'incertitude des paramètres d'entrée.

Par exemple, les deux premiers moments de cette loi (s'ils existent) s'interprètent très facilement : l'espérance de Y est la moyenne pondérée sur l'ensemble des valeurs possibles du p-uplet de paramètres, tandis que la variance de Y quantifie la dispersion de la sortie autour de cette moyenne, lorsque tous les paramètres varient suivant la loi qui leur est assignée. Cette variance est donc une mesure quantitative de l'impact de l'incertitude attachée aux paramètres d'entrée sur la sortie.

Il est alors naturel de chercher à *séparer* l'impact de l'incertitude sur *chacun* des paramètres d'entrée, et de définir, pour chaque paramètre X_i d'entrée du modèle, un indice quantifiant la dispersion de la sortie qui est seulement due à la variabilité de X_i. La définition et l'interprétation de ces indices, qui fait l'objet du paragraphe suivant, constituent l'analyse de sensibilité du modèle.

1.2.2 Indices de sensibilité

Dans cette section, nous supposons que Y est une variable $L^2(P)$, pour une mesure de probabilité P. Nous supposons que Y est de variance non nulle.

Indices basés sur la corrélation et la régression linéaire.

Les principales mesures d'importance rangées dans cette catégorie sont :
– le coefficient de corrélation linéaire (Pearson) entre Y et X_i :

$$\rho_i = \frac{\mathrm{Cov}(Y, X_i)}{\sqrt{\mathrm{Var}Y \times \mathrm{Var}X_i}};$$

– le coefficient de régression standard : on considère l'approximation par moindres carrés de Y comme une fonction affine des paramètres :

$$Y = \beta_0 + \sum_{j=1}^{p} \beta_j X_j + \varepsilon$$

où

$$(\beta_0, \ldots, \beta_p) = \underset{(\alpha_0,\ldots,\alpha_p)\in\mathbb{R}^{p+1}}{\mathrm{argmin}} \mathbb{E}\left(\left(Y - \alpha_0 - \sum_{j=1}^{p} \alpha_j X_j\right)^2\right)$$

et

$$\varepsilon = Y - \beta_0 - \sum_{j=1}^{p} \beta_j X_j.$$

On définit comme indice le coefficient de régression standard de Y sur X_i :

$$SRC_i = \beta_i \sqrt{\frac{\mathrm{Var}X_i}{\mathrm{Var}Y}}.$$

Remarquons que ces deux indices coïncident lorsque les X_i sont indépendants.
Ces deux indices ne sont pas interprétables si Y est trop « éloigné » d'une fonction affine des X_j ([54], p. 11). Dans ce cas, on peut utiliser des régressions sur les rangs (*loc. cit.*), qui donne lieu à des indices pertinents si (et seulement si) la relation entre Y et ses paramètres est monotone, ou, tout au moins, « proche de la monotonicité ». Les indices de Sobol, que nous décrivons ci-après, constituent des mesures de sensibilité qui restent adaptées dans des cas où Y est une fonction qui n'est ni linéaire, ni monotone des paramètres.

Indices basés sur la variance (indices de Sobol)

Faisons comme hypothèse que les paramètres d'entrée $X_1, \ldots, X_p$ sont indépendants. Rappelons également que nous supposons $\mathbb{E}(Y^2) + \infty$.
Fixons un paramètre d'entrée X_i, pour $i = 1, \ldots, p$. Rappelons que l'espérance conditionnelle $\mathbb{E}(Y|X_i)$ est une variable aléatoire, ne dépendant que de X_i, qui est la meilleure approximation (au sens L^2) de Y parmi toutes les fonctions de X_i. Sa variance est donc un réel quantifiant la dispersion de (la meilleure approximation de) Y lorsque seul X_i varie. Plus cette variance est élevée, plus X_i est « influent » sur la sortie Y, c'est-à-dire que la variation de Y due à la variation de X_i est grande.
En ramenant cette variance d'espérance conditionnelle sur la variance totale de Y, on obtient l'*indice de Sobol* associé à X_i :

$$S_i = \frac{\mathrm{Var}(\mathbb{E}(Y|X_i))}{\mathrm{Var}Y}. \tag{1.2.1}$$

Ces indices ont été proposés et étudiés dans [97]. On peut démontrer que ces indices sont normalisés, au sens où $0 \leq S_i \leq 1$. Remarquons également que ces indices étendent les indices basés sur la corrélation, au sens où, si Y est une fonction affine des paramètres, on a $S_i = SRC_i^2$.
Il est également possible, étant donné un sous-ensemble de paramètres d'entrée $u \subseteq \{1, \ldots, p\}$, de considérer l'indice de Sobol « *closed* » défini ainsi :

$$S_u^{\mathrm{Cl}} = \frac{\mathrm{Var}(\mathbb{E}(Y|X_i, i \in u))}{\mathrm{Var}Y},$$

qui quantifie l'influence de la variation des paramètres indexés par u pris ensemble.
Signalons aussi l'indice *total* associé à i, défini par :

$$S_i^{\mathrm{T}} = 1 - S^{\mathrm{Cl}}_{\{1,\ldots,p\}\setminus\{i\}},$$

qui est une autre mesure de l'influence du paramètre X_i, en incluant l'effet de ses interactions avec les autres paramètres d'entrée.
Á titre d'exemple, on vérifiera que, pour $p = 3$, (X_1, X_2, X_3) uniformes sur $[0; 1]^3$,

$$Y = X_1 X_2 X_3$$

et $i = 1$, on a :

$$S_1 = \frac{9}{37} \text{ et } S_1^T = \frac{16}{37}.$$

Au cours de ce travail de thèse, nous nous sommes principalement intéressés aux indices S_i (aussi appelés *indices du premier ordre*). Néanmoins, certains de nos résultats peuvent s'étendre directement aux indices d'ordre supérieur S^{Cl} et S^{T}.
Par ailleurs, les indices de Sobol peuvent être définis et estimés dans le cas où les variables d'entrées sont des processus stochastiques à valeurs dans un espace de Hilbert séparable [36].

Autres indices de sensibilité

Nous présentons ici quelques autres familles d'indices de sensibilité ; ils se basent sur d'autres formalisations de la notion de paramètre influent que celle présentée plus haut.

Indices basés sur la distribution conditionnelle. Les indices de Sobol définis plus haut n'utilisent que les deux premiers moments des distributions de Y conditionnées par les paramètres d'entrée. Borgonovo et coauteurs [8, 9] ont proposé et étudié une mesure de sensibilité faisant intervenir l'intégralité de ces distributions. Notons f_Y la densité de Y, et, pour $i = 1, \dots, p$, $f_{Y|X_i}$ la densité conditionnelle de Y sachant X_i. Si ces deux densités sont « très éloignées », cela signifie que fixer X_i modifie grandement la distribution de Y, donc que le paramètre X_i est influent. Quantifions la distance entre ces densités par une distance L^1 en définissant la variable aléatoire s, fonction de X_i, par :

$$s(X_i) = \int_{\mathbb{R}} \left| f_Y(y) - f_{Y|X_i}(y) \right| \mathrm{d}y.$$

On intègre alors cette distance suivant la loi de X_i pour obtenir l'indice δ_i :

$$\delta_i = \frac{1}{2}\mathbb{E}(s(X_i)).$$

L'indice δ_i jouit de la même propriété de normalisation que les indices de Sobol : $0 \leq \delta_i \leq 1$ ([8], Propriété 1), et se généralise naturellement aux groupes de variables.

Indices basés sur les dérivées. Indépendamment du cadre stochastique (ie., la loi de probabilité donnée aux paramètres d'entrées), la dérivée partielle $\dfrac{\partial f}{\partial X_i}(X_1, \dots, X_p)$ est un indicateur (signé) de la sensibilité de f à sa

$i^{\text{ème}}$ variable, localement autour de $(X_1, \ldots, X_p)$. Sobol et Kucherenko ([99]) intègrent cet indicateur afin d'obtenir une mesure globale de sensibilité :

$$\nu_i = \mathbb{E}\left(\left(\frac{\partial f}{\partial X_i}(X_1, \ldots, X_p)\right)^2\right).$$

Il est facile de vérifier que, dans le cas où f est affine, et si les entrées sont indépendantes, on a $\nu_i = S_i$. Dans le cas général, on peut obtenir une inégalité reliant ν_i et S_i, au prix d'une hypothèse sur la loi des paramètres d'entrée (qui exclut par exemple les lois uniformes sur des intervalles compacts) (*op. cit.* et [63]).

1.2.3 Calcul des indices de Sobol

Nous nous centrons maintenant sur l'aspect pratique du calcul des indices de Sobol du premier ordre définis en (1.2.1), toujours sous hypothèse d'indépendance des paramètres d'entrée.

Nécessité de l'approximation numérique

Le calcul exact des indices de sensibilité, en particulier celui des indices de Sobol, n'est pas toujours possible. En effet, les lois des paramètres d'entrée, ainsi que l'expression de la sortie du modèle f peuvent être extrêmement complexes, et les intégrales exprimant les variances et les variances d'espérances conditionnelles qui interviennent dans la définition des indices peuvent s'avérer incalculables analytiquement. Il faut donc avoir recours à des méthodes d'approximation numérique, qui sont au demeurant suffisantes pour les applications pratiques décrites au paragraphe précédent.
Présentons maintenant quelques méthodes d'approximation numérique des indices de Sobol, regroupées en fonction du type de plan d'expériences qu'elles utilisent. Nous commençons par présenter les estimateurs basés sur un plan Monte-Carlo *pick-freeze*, qui sont les estimateurs sur lesquels nous nous sommes centrés durant cette thèse.

Utilisation d'un plan Monte-Carlo *pick-freeze*

Nous pouvons réécrire la définition de S_i de la manière suivante :

$$S_i = \frac{\mathrm{Cov}(Y, Y')}{\mathrm{Var}Y} \tag{1.2.2}$$

pour :

$$Y' = f(X'_1, X'_2, \ldots, X'_{i-1}, X_i, X'_{i+1}, \ldots, X'_p),$$

et où, pour $j \neq i$, X'_j désigne une variable aléatoire indépendante et de même loi que X_j.

Cette propriété est démontrée au chapitre 3 (Lemme 1).

Le nom du plan provient de la façon de générer les paramètres d'entrée utilisés pour calculer Y' à partir de ceux utilisés pour calculer Y : la variable i, qui est la variable choisie (*picked*), est gelée (*frozen*) et toutes les autres sont remplacées par des copies indépendantes et de même loi.

Fixons maintenant $N \in \mathbb{N}^*$ et considérons deux échantillons indépendants identiquement distribués (iid) :

$$\left(X_1^k, \ldots, X_p^k\right)_{k=1,\ldots,N} \text{ et } \left(X_1^{k\prime}, \ldots, X_p^{k\prime}\right)_{k=1,\ldots,N}$$

de la loi de $(X_1, \ldots, X_p)$.

Notons maintenant, pour $k = 1, \ldots, N$:

$$Y_k = f(X_1^k, \ldots, X_p^k) \text{ et } Y'_k = f(X_1^{k\prime}, \ldots, {X_{i-1}^k}', X_i^k, {X_{i+1}^k}', \ldots, X_p^{k\prime}).$$

En remplaçant dans (1.2.2) les covariances et variances par leurs estimateurs empiriques, on obtient un premier estimateur naturel de S_i :

$$\widehat{S}_i = \frac{\frac{1}{N}\sum_{k=1}^N Y_k Y'_k - \left(\frac{1}{N}\sum_{k=1}^N Y_k\right)\left(\frac{1}{N}\sum_{k=1}^N Y'_k\right)}{\frac{1}{N}\sum_{k=1}^N Y_k^2 - \left(\frac{1}{N}\sum_{k=1}^N Y_k\right)^2}.$$

En remarquant que $\mathbb{E}(Y) = \mathbb{E}(Y')$, on peut aussi remplacer l'estimation empirique de $\mathbb{E}(Y')$ par celle de $\mathbb{E}(Y)$ et ainsi obtenir l'estimateur originellement proposé par Sobol dans [98] :

$$\widehat{S}_i^{\text{Sob}} = \frac{\frac{1}{N}\sum_{k=1}^N Y_k Y'_k - \left(\frac{1}{N}\sum_{k=1}^N Y_k\right)^2}{\frac{1}{N}\sum_{k=1}^N Y_k^2 - \left(\frac{1}{N}\sum_{k=1}^N Y_k\right)^2}.$$

Enfin, comme Y et Y' ont même loi, on peut estimer $\mathbb{E}(Y)$ et $\mathbb{E}(Y^2)$ par :

$$\widehat{E} = \frac{1}{N}\sum_{k=1}^N \frac{Y_k + Y'_k}{2} \text{ et } \widehat{E}_2 = \frac{1}{N}\sum_{k=1}^N \frac{Y_k^2 + Y_k'^2}{2},$$

ce qui donne lieu à un troisième estimateur, proposé dans [68] :

$$\widehat{T}_i = \frac{\frac{1}{N}\sum_{k=1}^N Y_k Y'_k - \widehat{E}^2}{\widehat{E}_2 - \widehat{E}^2}.$$

Ces trois estimateurs peuvent donner une valeur approchée de l'indice de Sobol S_i à partir de $2N$ évaluations de la fonction donnant la sortie du modèle f, et l'on peut estimer les p indices du premier ordre $S_1, \ldots, S_p$ à l'aide de $N(p+1)$ évaluations de f.
Remarquons que tous ces estimateurs peuvent être utilisés en utilisant un échantillonnage déterministe de la loi de X (suites à discrépance faible) ou un échantillonnage aléatoire mais non iid (par exemple, un échantillonnage par hypercubes latins étudié dans [104]). Cependant, l'étude théorique des propriétés de $\widehat{S}_i$ et de $\widehat{T}_i$ que nous ferons au Chapitre 3 repose sur le fait que le plan d'expérience soit aléatoire et iid.
L'écriture (1.2.2) permet également d'exprimer S_i à l'aide d'intégrales dans l'espace des paramètres d'entrée, qui peuvent être estimées à l'aide d'une méthode usuelle de quadrature multidimensionnelle, telle la méthode de Smolyak (grille creuse, *sparse grid*), voir [37] pour des généralités sur cette méthode et [14] pour l'application au calcul des indices de Sobol.
Enfin, notons que la propriété (1.2.2) se généralise aux indices *closed* :

$$S_u^{\mathrm{Cl}} = \frac{\mathrm{Cov}(Y, Y^u)}{\mathrm{Var} Y}$$

où $Y^u = f(X^u)$, où la $j^{\text{ème}}$ composante de $X^u \in \mathbb{R}^p$ est donnée par :

$$X_j^u = \begin{cases} X'_j \text{ si } j \notin u \\ X_j \text{ si } j \in u. \end{cases}$$

Cette égalité permet d'utiliser les estimateurs présentés plus haut pour estimer S_u^{Cl}, il suffit de remplacer l'échantillon $\{Y'_k\}$ par un échantillon iid. de la loi de Y^u. Par complément à 1, les indices totaux S_i^T peuvent également être estimés.
Dans ce travail, nous nous sommes centrés sur l'étude des estimateurs basés sur un plan Monte-Carlo *pick-freeze*. Donnons cependant quelques autres estimateurs.

Utilisation d'un plan Monte-Carlo *pick-freeze* répliqué

En reprenant les notations du paragraphe précédent, notons :

$$Y'' = f(X'_1, \ldots, X'_p) \text{ et } Y''_k = f(X_1^{k\prime}, \ldots, X_p^{k\prime}).$$

On rajoute donc au plan d'expérience *pick-freeze* utilisé plus haut les « réplications » indépendantes issues de l'échantillon primé $\left\{\left(X_1^{k\prime}, \ldots, X_p^{k\prime}\right)\right\}_{k=1,\ldots,N}$.

Comme Y et Y'' sont indépendants, le numérateur de S_i peut se réécrire :

$$\mathrm{Cov}(Y, Y') = \mathbb{E}(YY') - \mathbb{E}(Y)^2 = \mathbb{E}(Y(Y' - Y'')),$$

comme noté dans [99], §4.
Cela donne lieu à un nouvel estimateur empirique de S_i :

$$\widehat{S}_i^{\mathrm{R}} = \frac{\frac{1}{N}\sum_{k=1}^{N} Y_k(Y'_k - Y''_k)}{\frac{1}{N}\sum_{k=1}^{N} Y_k^2 - \left(\frac{1}{N}\sum_{k=1}^{N} Y_k\right)^2}.$$

Comparé à l'estimateur de Sobol original $\widehat{S}^{\mathrm{Sob}}$, cet estimateur nécessite un plan d'expérience de plus grande taille ($3N$ au lieu de $2N$) mais il s'est avéré plus précis sur certains cas tests (voir *loc. cit.*), lorsque l'indice à estimer est de « faible » valeur.

Utilisation d'un plan basé sur l'analyse harmonique (FAST et variantes)

Décrivons maintenant la méthode FAST (*Fourier Amplitude Sensitivity Test*), telle que proposée dans [22, 24, 23] et présentée dans une plus grande généralité dans [106].
Commençons par supposer que $X = (X_1, \ldots, X_p)$ suit une loi uniforme sur $[0;1]^p$. Lorsque les paramètres d'entrée sont indépendants, il est toujours possible (au moins en théorie) de se ramener à ce cas, par composition avec la fonction de répartition vectorielle $F : \mathbb{R}^p \to [0;1]^p$ donnée par :

$$F(x_1, \ldots, x_p) = (P(X_1 \leq x_1), \ldots, P(X_p \leq x_p)).$$

Pour tout p-uplet $k \in \mathbb{Z}^p$, notons $c_k(f)$ le $k^{\text{ème}}$ coefficient de Fourier de f, donné par :

$$c_k(f) = \int_{[0;1]^p} f(X)\exp(-2i\pi X \cdot k)\mathrm{d}X.$$

Pour tout $i = 1, \ldots, p$, le $i^{\text{ème}}$ indice de Sobol s'exprime en fonction des coefficients de Fourier :

$$S_i = \frac{\sum_{l\in\mathbb{Z}^*} |c_{l\delta_i}(f)|^2}{\sum_{k\in\mathbb{Z}^d, k\neq 0} |c_k(f)|^2}, \tag{1.2.3}$$

où δ_i est le vecteur de $\mathbb{Z}^p$ dont la $j^{\text{ème}}$ composante vaut :

$$(\delta_i)_j = \begin{cases} 1 \text{ si } i = j, \\ 0 \text{ sinon.} \end{cases}$$

Choisissons un vecteur de *fréquences* $\omega = (\omega_1, \ldots, \omega_p) \in \mathbb{N}^p$, et notons $x(t)$ le vecteur de $\mathbb{R}^p$ dont la $j^{\text{ème}}$ composante est donnée par :

$$(x(t))_j = \frac{1}{\pi} \arcsin(\sin(\omega_j t)) + \frac{1}{2}.$$

Notons, pour $l \in \mathbb{Z}$:

$$c_l^{\text{Erg}}(f) = \int_0^1 f \circ x(t) \exp(-2i\pi l t) \mathrm{d}t.$$

Les auteurs de FAST proposent l'approximation suivante des coefficients de Fourier de f :

$$c_k(f) \approx c_{k\cdot\omega}^{\text{Erg}}(f),$$

où $k \cdot \omega$ est le produit scalaire Euclidien de k avec ω.

Cette approximation, qui permet de remplacer une intégrale p-dimensionnelle par une intégrale unidimensionnelle, est basée sur le théorème ergodique de Weyl [114], et n'est pertinente que si le vecteur ω est bien choisi, par exemple à l'aide du critère d'absence d'interférences donné dans [22].

L'étape suivante est d'approcher l'intégrale apparaissant dans $c_l^{\text{Erg}}(f)$ par la méthode des rectangles à $N > 0$ points, ce qui donne lieu à l'approximation de $c_l^{\text{Erg}}(f)$:

$$c_l^{\text{FAST}}(f) = \frac{1}{n} \sum_{j=0}^{N-1} f \circ x\left(\frac{j}{N}\right) \exp\left(-2i\pi l \frac{j}{N}\right).$$

Enfin, les sommes apparaissant dans (1.2.3) sont tronquées, ce qui donne l'approximation S_i^{FAST} suivante :

$$S_i^{\text{FAST}} = \frac{\sum_{k=1}^{n-1} \left| c_{k\omega_i}^{\text{FAST}}(f) \right|^2}{\sum_{l=1}^{n-1} \left| c_l^{\text{FAST}}(f) \right|^2}.$$

La méthode FAST a été ensuite étendue en méthode EFAST, afin de permettre l'estimation des indices totaux ([90]), puis hybridée avec la méthode RBD (*random balanced design*), ce qui a donné la méthode RBD-FAST [103]. Ces méthodes ont récemment fait l'objet de diverses généralisations avec analyse d'erreur dans [106]. Une autre analyse d'erreur de la méthode FAST est proposée dans [117].

Utilisation d'une décomposition en polynômes de chaos

Notre référence dans cette section est [38].
Nous supposons cette fois-ci que le vecteur d'entrées $X = (X_1, \dots, X_p)$ suit une loi normale multivariée $\mathcal{N}_p(0, I_p)$, où I_p désigne la matrice identité d'ordre p.

Polynômes de chaos. Nous introduisons la famille de polynômes orthogonaux de Hermite $(\psi_n)_{n\in\mathbb{N}}$, tels que $\deg \psi_n = n$, et

$$\int_{\mathbb{R}} \psi_n(x)\psi_m(x)w(x)\mathrm{d}x = \begin{cases} 1 & \text{si } n = m \\ 0 & \text{sinon} \end{cases}$$

pour tous n et m entiers, où la fonction de poids $w(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)$ est la densité d'une variable aléatoire normale centrée réduite.
Etant donné un multi-indice $i = (i_1, \dots, i_p) \in \mathbb{N}^p$, on note :

$$\psi_i(X) = \psi_{i_1}(x_1) \cdots \psi_{i_d}(x_p) \tag{1.2.4}$$

le polynôme de Hermite à p variables d'ordre i.
Comme $Y \in L^2(\Omega)$ est X-mesurable, le théorème de Cameron-Martin [16] affirme que Y peut s'écrire sous la forme d'une décomposition en polynômes de chaos (de Hermite) :

$$Y = \sum_{i\in\mathbb{N}^p} y_i \psi_i(X) \tag{1.2.5}$$

pour une collection infinie de réels $(y_i)_{i\in\mathbb{N}^p}$ appelés coefficients de Y dans sa décomposition en polynômes de chaos, la sommation sur le multi-indice i étant convergente dans L^2.

Lien avec les indices de Sobol [21, 102]. L'intérêt de la décomposition en polynômes de chaos est que les indices de Sobol peuvent s'exprimer en fonction des coefficients de Y dans cette décomposition, à l'aide d'une formule très similaire à celle utilisée pour la méthode FAST :

$$S_i = \frac{\sum_{j=1}^{\infty} y_{I(j,i)}^2}{\sum_{j\in\mathbb{N}^p, j\neq 0} y_j^2}, \tag{1.2.6}$$

où :

$$I(j,i) = (\underbrace{0, \dots, 0}_{i-1}, j, \underbrace{0, \dots, 0}_{p-i})$$

est le multi-indice de $\mathbb{N}^p$ contenant j à la $i$ème place, et 0 aux autres places. Le lien entre cette formule et la méthode FAST est précisé dans [106], l'idée sous-jacente étant l'utilisation de la tensorisation d'une base orthonormée de $L^2(P_{X_i})$.
Pour être évaluées numériquement, les sommes de (1.2.6) doivent être tronquées à un ordre $P \in \mathbb{N}^*$:

$$S_i \approx \frac{\sum_{j=1}^{P} y_{I(j,i)}^2}{\sum_{j\in\mathbb{N}^p, 0<|j|\leq P} y_j^2}, \tag{1.2.7}$$

où, pour un multi-indice $j = (j_1, \ldots, j_p) \in \mathbb{N}^p$,

$$|j| = j_1 + \ldots + j_p.$$

Tout revient donc à approcher les $\dfrac{(p+P)!}{p!P!} - 1$ coefficients y_j (pour $0 < |j| < P$) par $\tilde{y}_j$, ce qui donne lieu à l'approximation de S_i par S_i^{PC} définie par :

$$S_i^{\mathrm{PC}} = \frac{\sum_{j=1}^{P} \tilde{y}_{I(j,i)}^2}{\sum_{j\in\mathbb{N}^p, 0<|j|\leq P} \tilde{y}_j^2}.$$

Les paragraphes suivants sont consacrés à trois méthodes de calcul des $\tilde{y}_j$.

Calcul par projection. Les propriétés d'orthogonalité des polynômes de Hermite permettent d'écrire :

$$\forall j \in \mathbb{N}^p, \quad y_j = \int_{\mathbb{R}^p} Y \psi_j w_d,$$

et cette intégrale multi-dimensionnelle peut ensuite être approchée par diverses méthodes de quadrature, tant déterministes que stochastiques (Monte-Carlo).

Calcul par régression. Dans l'approche par régression, présentée dans [102], on se donne un plan d'expérience $\Xi = \left\{X^{(1)}, \ldots, X^{(N)}\right\} \subset \mathbb{R}^d$, on calcule les sorties du modèle sur ce plan : $Y(X^{(1)}), \ldots, Y(X^{(N)})$, puis on estime les coefficients de la décomposition en polynômes de chaos tronquée par moindres carrés :

$$(\tilde{y}_i)_{i\in\mathbb{N}^d, |i|\leq P} = \underset{(z_i)_{i\in\mathbb{N}^d, |i|\leq P}}{\operatorname{argmin}} \sum_{n=1}^{N} \left(Y\left(X^{(n)}\right) - \sum_{|i|\leq P} z_i \psi_i\left(X^{(n)}\right) \right)^2 \tag{1.2.8}$$

Pour que ce problème soit bien posé, la taille du plan d'expérience (N) doit être supérieure ou égale au nombre de coefficients à estimer. Lorsque ces deux quantités sont égales, la régression est une interpolation et la méthode porte alors le nom de *collocation*.
Pour le choix du plan d'expérience, [102] contient des stratégies basées sur les racines de polynômes de Hermite.
Le problème (1.2.8) peut également être rendu bien posé lorsque le plan d'expérience contient moins de points que de coefficients à estimer par une méthode de régularisation L^1. Ceci est intéressant car le nombre de coefficients à estimer croît rapidement avec le nombre d'entrées p et l'ordre de troncature P. On peut ainsi être amené à résoudre (1.2.8) avec la contrainte supplémentaire :

$$\sum_{i\in\mathbb{N}^d,|i|\leq P} |z_i| \leq s \tag{1.2.9}$$

où s est un paramètre de régularisation à choisir. Plus s est petit, et plus les solutions sont « creuses » (c'est à dire que beaucoup des $\tilde{y}_i$ estimés sont nuls). Ce problème de moindres carrés contraints peut être résolu à l'aide d'un algorithme *Least Angle Regression* [7].

Calcul par une méthode intrusive. Par rapport aux autres méthodes présentées, qui traitaient le modèle comme une boîte noire à laquelle on peut faire appel pour obtenir les sorties évaluées sur différents jeux de paramètres d'entrée, les méthodes intrusives type Galerkin font des hypothèses fortes sur le « type » de sortie considérée.
Nous esquissons seulement le principe de la méthode, et renvoyons à [38] et [29] pour plus de détails.
Dans cette approche, on suppose que Y est une fonctionnelle ϕ d'une fonction $u = u(\cdot, X)$ satisfaisant une équation aux dérivées partielles possédant de "bonnes propriétés" (par exemple, l'ellipticité et la linéarité). L'idée consiste à injecter la décomposition de u en polynômes de chaos :

$$u(\cdot, X) = \sum_{j\in\mathbb{N}^p} u_j(\cdot)\psi_j(X)$$

dans l'équation aux dérivées partielles afin d'écrire un système (infini) d'EDP portant sur les fonctions u_j. Ce système est alors tronqué afin de ne conserver qu'un nombre fini de $u_j(\cdot)$ à calculer ; puis le système d'EDP obtenu est discrétisé afin d'obtenir un système fini d'équations portant sur des inconnues réelles.

La résolution de ce système permet dans certains cas (le cas le plus simple étant le cas où ϕ est linéaire) d'estimer les coefficients de la décomposition en polynômes de chaos de Y.
Signalons que le système fini d'équations à résoudre se révèle être souvent de grande taille, et que des méthodes spécifiques de réduction de dimension, telle que la décomposition spectrale généralisée (GSD) [74] peuvent être utilisées afin d'accélérer sa résolution numérique.
Terminons cette section en citant une autre méthode d'estimation des indices de Sobol, qui est présentée et étudiée dans [25] ; l'étude de ces estimateurs faisant appel à des outils développés dans [64] pour l'estimation d'intégrales de fonctionnelles d'une densité.

1.2.4 Évaluation de l'erreur d'estimation par Monte-Carlo

L'utilisation d'une approximation numérique pose naturellement la question de la précision de cette approximation. Lorsqu'un plan d'expérience aléatoire est utilisé (plan Monte-Carlo), il est traditionnel en statistique de chercher un intervalle de confiance pour la quantité à estimer.
Un tel intervalle de confiance peut être obtenu de deux façons : par bootstrap ou en utilisant un résultat de normalité asymptotique.

Intervalles de confiance par bootstrap

Rappelons le principe du *bootstrap* expliqué dans [33].
Soit un estimateur $\widehat{U}$ d'une quantité U inconnue, fonction de la distribution d'une population $\mathcal{P}$. Cet estimateur est fonction d'un échantillon de taille N noté $(Z_1, \ldots, Z_N)$:

$$\widehat{U} = \widehat{U}(Z_1, \ldots, Z_N).$$

Dans notre contexte, on a $\widehat{U} \in \{\widehat{S}_i, \widehat{T}_i, \widehat{S}_i^{\mathrm{Sob}}, \widehat{S}_i^{\mathrm{R}}\}$, et $Z_i = (Y_i, Y_i', Y_i'')$ lorsque $\widehat{U} = \widehat{S}_i^{\mathrm{R}}$ (cas d'un plan pick-freeze répliqué), ou $Z_i = (Y_i, Y_i')$ sinon (cas d'un plan pick-freeze). L'utilisation du bootstrap dans le cadre du calcul des indices de Sobol a été considérée, sous un angle pratique, dans [1].
Une estimation ponctuelle de U est obtenue en tirant aléatoirement un échantillon $(z_1, \ldots, z_N)$ par N tirages indépendants dans $\mathcal{P}$, et en calculant $u = \widehat{U}(z_1, \ldots, z_N)$.
Le principe du bootstrap est de fixer un nombre R de réplications, et de tirer, pour $r = 1, \ldots, R$, de manière équiprobable *et avec remise* dans $(z_1, \ldots, z_N)$

un N-échantillon $(z_1^{i*}, z_2^{i*}, \ldots, z_N^{i*})$. On obtient alors la $r^{\text{ème}}$ *réplication* de $\widehat{U}$, notée u^{r*} :

$$u^{r*} = \widehat{U}(z_1^{i*}, z_2^{i*}, \ldots, z_N^{i*}).$$

Il existe alors plusieurs variantes du bootstrap, qui diffèrent dans la façon de passer de l'ensemble :

$$\mathcal{R} = \{u^{1*}, \ldots, u^{R*}\}$$

de réplications de $\widehat{U}$ à un intervalle de confiance de risque $\alpha \in]0;1[$ pour U. Les deux premières variantes que nous décrivons sont données dans [33].

Bootstrap classique (*vanilla*). On prend comme intervalle de confiance approché au niveau $1-\alpha$ l'intervalle dont les extrémités sont les quantiles $\alpha/2$ et $1-\alpha/2$ de $\mathcal{R}$.

Bootstrap avec approximation gaussienne. On approxime les quantiles $\alpha/2$ et $1-\alpha/2$ de $\mathcal{R}$ respectivement par $m - q_\alpha\sigma$ et $m + q_\alpha\sigma$, où m et σ désignent resp. la moyenne et l'écart-type de $\mathcal{R}$, et q_α est le quantile $1-\alpha/2$ d'une loi normale centrée réduite.

Le bootstrap classique ou avec approximation gaussienne peuvent mal fonctionner dans le cas où l'estimateur $\widehat{U}$ est biaisé. Pour pallier ce défaut, il a été proposé [31, 32] la méthode suivante de bootstrap, appelée *bias-corrected bootstrap* (BC bootstrap).

Bootstrap avec correction de biais. Notons Φ la fonction de répartition de la loi normale centrée réduite :

$$\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} \exp\left(-\frac{t^2}{2}\right) \mathrm{d}t,$$

et Φ^{-1} son inverse.
En utilisant $\mathcal{R}$ et l'estimée ponctuelle $u = \widehat{U}(z_1, \ldots, z_N)$, on estime une « constante de correction de biais » z_0 :

$$\widehat{z_0} = \Phi^{-1}\left(\frac{\#\{u^* \in \mathcal{R} \text{ tels que } u^* \leq \widehat{u}\}}{R}\right).$$

Puis, pour $\beta \in]0;1[$, on définit l'« estimateur de quantile corrigé » par :

$$\widehat{q}(\beta) = \Phi(2\widehat{z_0} + z_\beta)$$

où $z_\beta = \Phi^{-1}(\beta)$.
L'intervalle de confiance fourni par cette méthode est alors l'intervalle dont les extrémités sont les quantiles $\widehat{q}(\alpha/2)$ et $\widehat{q}(1-\alpha/2)$ de $\mathcal{R}$.
Les intervalles de confiance construits par cette méthode sont robustes au biais et à la non-normalité de $\widehat{U}$. Plus précisément, cette construction est justifiée dans [31] dans le cas où il existe une fonction croissante g, $z_0 \in \mathbb{R}$ et $\sigma > 0$ tels que :

$$g(\widehat{U}) \overset{\mathcal{L}}{=} \mathcal{N}(U - z_0\sigma, \sigma^2) \text{ et } g(\widehat{U}^*) \overset{\mathcal{L}}{=} \mathcal{N}(u - z_0\sigma, \sigma^2),$$

où $\mathcal{N}(m, \sigma^2)$ désigne la loi normale de moyenne m et de variance σ^2, et $\widehat{U}^*$ est l'estimateur $\widehat{U}$ bootstrappé, à échantillon $\{z_1, \ldots, z_n\}$ et estimée ponctuelle u fixée, c'est-à-dire que :

$$\widehat{U}^* = \widehat{U}(Z_1^*, \ldots, Z_N^*),$$

pour $(Z_1^*, \ldots, Z_N^*)$ un échantillon iid. de la loi équirépartie sur $(z_1, \ldots, z_N)$ (en d'autres termes, les réplications $u^{1*}, u^{2*}, \ldots, u^{R*}$ sont R réalisations iid. de $\widehat{U}^*$).
Dans la pratique, la fonction g nous semble difficile à expliciter ; par contre, l'observation du diagramme quantile-quantile (*QQ plot*) de l'échantillon de réplications $\mathcal{R}$ permet de valider empiriquement l'existence d'une telle fonction g.

L'avantage d'une méthode de bootstrap provient de sa nature algorithmique et de sa mise en oeuvre possible pour tout estimateur, même ayant une expression complexe en fonction de l'échantillon. Cependant, même si son utilisation en pratique donne de bons résultats, la justification théorique de son utilisation est délicate. De plus, le bootstrap ne donne pas d'expression analytique de la largeur des intervalles de confiance obtenus (indicateur de la précision de l'estimation ponctuelle), qui soit exploitable dans le but de comparer en général (indépendemment d'exemples numériques) la précision de plusieurs estimateurs de la même quantité. Ces deux points justifient l'utilisation d'intervalles de confiance obtenus par normalité asymptotique, que nous décrivons maintenant.

Intervalles de confiance par normalité asymptotique

On dit qu'une suite d'estimateurs $(\widehat{U}_N)_N$ convergeant presque sûrement vers une valeur U, possède la propriété de normalité asymptotique s'il existe des

réels strictement positifs γ et σ^2, appelés respectivement *vitesse de convergence* et *variance asymptotique*, tels qu'on ait la convergence suivante en loi :

$$N^{\gamma}(\widehat{U}_N - U) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}(0, \sigma^2)$$

où $\mathcal{N}(0, \sigma^2)$ désigne la loi normale centrée de variance σ^2.
Dans notre contexte d'estimation des indices de Sobol, on a $U = S_i$ et la suite d'estimateurs est la suite, indexée par la taille de l'échantillon, donnée par l'expression de $\widehat{S}_i$, $\widehat{T}_i$, $\widehat{S}_i^{\mathrm{Sob}}$ ou $\widehat{S}_i^{\mathrm{R}}$.
Étant donné un niveau de risque $\alpha \in]0;1[$, un intervalle de confiance *asymptotique* au niveau $1-\alpha$ est donné par :

$$I_{N,\alpha} = \left]\widehat{U}_N - q_\alpha \frac{\sigma}{N^{\gamma}}; \widehat{U}_N + q_\alpha \frac{\sigma}{N^{\gamma}}\right[,$$

où q_α est, comme précédemment, le quantile $1-\alpha/2$ d'une loi normale centrée réduite.
Cet intervalle est justifié asymptotiquement par le fait que, par définition de la convergence en loi :

$$P(I_{N,\alpha} \ni U) \underset{N\to\infty}{\longrightarrow} 1-\alpha. \tag{1.2.10}$$

En pratique, σ n'est pas calculable analytiquement, mais peut s'estimer, à partir du même plan d'expérience et de manière consistante, par une suite $\widehat{\sigma}_N^2$, ce qui donne l'intervalle de confiance asymptotique suivant :

$$\widehat{I}_{N,\alpha} = \left]\widehat{U}_N - q_\alpha \frac{\widehat{\sigma}_N}{N^{\gamma}}; \widehat{U}_N + q_\alpha \frac{\widehat{\sigma}_N}{N^{\gamma}}\right[,$$

qui satisfait également la propriété (1.2.10).
Les propriétés de normalité asymptotique pour $\widehat{S}_i$ et $\widehat{T}_i$ sont établies dans le Chapitre 3, Proposition 6.

1.3 Introduction à la métamodélisation

Dans cette section, nous décrivons en détails les principes de trois méthodes de métamodélisation. Rappelons que le but de ces méthodes est de fournir des algorithmes de construction d'une fonction $\widetilde{f}$ approchant le modèle numérique à étudier f tout en étant plus rapide à évaluer numériquement que celui-ci. Nous faisons également, en lien avec l'objet de cette thèse, une présentation de l'analyse de l'erreur entre f et $\widetilde{f}$.
Nous présentons les méthodes suivantes :

– le krigeage (1.3.1) ;
– l'interpolation à noyau (1.3.2) ;
– la méthode base réduite (1.3.3).

Signalons qu'il existe d'autres méthodes de métamodélisation que celles présentées ici. On en trouvera un aperçu dans le chapitre 5 de [35].

1.3.1 Métamodélisation par krigeage

Le *krigeage*, aussi appelé *métamodélisation par processus gaussiens* est une méthode de construction d'un métamodèle apparue dans [67]. Nous allons maintenant présenter cette méthode, en renvoyant à [93], [75] §3.1, [60] pour les détails.

L'hypothèse centrale du krigeage est de supposer que f est une réalisation d'un processus gaussien défini sur un espace probabilisé (Ω, P_Ω) :

$$\exists \omega^* \in \Omega, \quad f(\cdot) = Z(\cdot, \omega^*), \tag{1.3.1}$$

où Z est un processus gaussien, i.e. pour tous $k \in \mathbb{N}$ et pour tous

$$\left(x^1, \ldots, x^k\right) \in (\mathbb{R}^p)^k,$$

le vecteur :

$$(Z(x^1), \ldots, Z(x^k))$$

suit, sous P_Ω, une loi gaussienne en dimension k.

On choisit souvent comme espérance de Z une fonction affine :

$$\forall x = (x_1, \ldots, x_p), \quad \mathbb{E}_\Omega(Z(x)) = m(x) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p,$$

et comme fonction de covariance de Z un produit de puissances d'exponentielles :

$$\forall x = (x_1, \ldots, x_p), \forall x' = (x'_1, \ldots, x'_p)$$
$$\mathrm{Cov}_\Omega(Z(x), Z(x')) = R(x, x') = \sigma^2 \prod_{j=1}^{p} \exp\left(-\theta_j \left|x_j - x'_j\right|^{\alpha_j}\right),$$

dépendants de $\Theta = (\beta_0, \ldots, \beta_p, \theta_1, \ldots, \theta_p, \sigma) \in \mathbb{R}^{p+1} \times (\mathbb{R}_+^*)^{p+1}$ et de $\alpha = (\alpha_1, \ldots, \alpha_p)$.

On suppose qu'on connaît un n-échantillon de sorties du (vrai) modèle, constituant notre *échantillon d'apprentissage* :

$$\mathcal{D} = \{(x^1, f(x^1)), \ldots, (x^n, f(x^n))\},$$

et on prend comme métamodèle :

$$\widetilde{f}(x) = \mathbb{E}_\Omega \left(Z(x) | Z(x^1) = f(x^1), \dots, Z(x^n) = f(x^n) \right). \tag{1.3.2}$$

Compte tenu de l'hypothèse de processus gaussien, cette espérance conditionnelle admet une expression analytique :

$$\widetilde{f}(x) = m(x) + k_{\mathcal{D}}(x)^t \Sigma_{\mathcal{D}}^{-1} y_{\mathcal{D}}, \tag{1.3.3}$$

où :

$$k_{\mathcal{D}}(x) = \left(R(x, x^1), \dots, R(x, x^n) \right)^t \in \mathbb{R}^n,$$

$$y_{\mathcal{D}}(x) = \left(f(x^1) - m(x^1), \dots, f(x^n) - m(x^n) \right)^t \in \mathbb{R}^n,$$

et $\Sigma_{\mathcal{D}}$ est une matrice symétrique $n \times n$ dont le coefficient (k, l) est donné par :

$$(\Sigma_{\mathcal{D}})_{k,l} = R(x^k, x^l).$$

En pratique, on ne connaît ni le vecteur α d'exposants, ni le vecteur Θ. Généralement, le choix de α est fait *a priori* (par exemple $\alpha = (1, \dots, 1)$ ou $\alpha = (2, \dots, 2)$), puis on estime Θ à partir de $\mathcal{D}$, par exemple à l'aide du principe de maximum de vraisemblance :

$$\widehat{\Theta} = \underset{\Theta \in \mathbb{R}^{p+1} \times \mathbb{R}_+^{p+1}}{\operatorname{argmax}} \; p_\Theta(f(x^1), \dots, f(x^n)),$$

où p_Θ désigne la densité, sous P_Ω, du vecteur gaussien $(Z(x^1), \dots, Z(x^n))$. La métamodélisation par krigeage admet une interprétation élégante dans un cadre bayésien : l'hypothèse (1.3.1) s'interprète comme une loi *a priori* sur la fonction f (inconnue) qui impose à f d'être lisse (les propriétés de dérivabilité d'un champ gaussien dépendant de sa fonction de covariance, cf. Theorème 2.2.1 de [56]). Les observations sont les éléments de l'échantillon d'apprentissage, et la loi *a posteriori*, loi du processus Z conditionnée par les observations, peut être identifiée explicitement [113] comme la loi d'un processus gaussien ayant comme espérance $\widetilde{f}$ et fonction de covariance :

$$\begin{aligned} \operatorname{Cov}_\Omega \left(Z(x), Z(x') | Z(x^1) = f(x^1), \dots, Z(x^n) = f(x^n) \right) \\ = R(x, x') - k_{\mathcal{D}}(x)^t \Sigma_{\mathcal{D}}^{-1} k_{\mathcal{D}}(x'). \end{aligned}$$

Dans ce contexte, le métamodèle $\widetilde{f}$ est l'estimateur bayésien du maximum *a posteriori*, et Θ forme le vecteur des hyperparamètres. L'utilisation des

hyperparamètres par maximum de vraisemblance qualifie donc la méthode décrite plus haut de méthode bayésienne empirique (*empirical Bayes*).
Remarquons enfin que l'on peut prendre comme métamodèle l'espérance de Z conditionnée non seulement par l'échantillon d'apprentissage, mais aussi par des informations connues *a priori* sur le modèle, par exemple la positivité, la monotonie, la convexité, les extrema [26]. Un tel conditionnement augmenté permet souvent d'améliorer les performances du krigeage, à taille d'échantillon d'apprentissage égale.

1.3.2 Métamodélisation par interpolation à noyau

La motivation et l'étude des métamodèles par interpolation à noyau se fait dans le contexte des espaces de Hilbert à noyau reproduisant (appelés RKHS, pour *reproducing kernel Hilbert space*). Commençons par présenter ces espaces.

Espaces de Hilbert à noyau reproduisant

Soit $(\mathcal{H}, \langle\cdot,\cdot\rangle_{\mathcal{H}}, \|\cdot\|_{\mathcal{H}})$ un espace de Hilbert contenu dans l'ensemble des fonctions définies sur $\mathcal{X} \subset \mathbb{R}^p$ à valeurs réelles. On dit que $\mathcal{H}$ est un RKHS si, quel que soit $x \in \mathcal{X}$, la forme linéaire d'évaluation en x :

$$\delta_x : \mathcal{H} \to \mathbb{R}, \quad f \mapsto f(x)$$

est continue pour $\|\cdot\|_{\mathcal{H}}$.
L'application du théorème de représentation de Riesz donne l'existence d'un noyau reproduisant $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ tel que :

$$\forall f \in \mathcal{H},\ \forall x \in \mathcal{X},\ f(x) = \langle f, K(x,\cdot)\rangle_{\mathcal{H}},$$

qui satisfait de plus la propriété de symétrie :

$$\forall (x,y) \in \mathcal{X} \times \mathcal{X},\ K(x,y) = K(y,x)$$

et de définie positivité :

$$\forall k \in \mathbb{N}, (\alpha_1,\dots,\alpha_k) \in \mathbb{R}^k, \forall x^1,\dots,x^k \in \mathcal{X},\ \sum_{i=1}^{k}\sum_{j=1}^{k} \alpha_i\alpha_j K(x^i,x^j) \geq 0,$$

avec égalité si et seulement si $\alpha_i = 0\, \forall i = 1,\dots,k$.

La réciproque de ce résultat constitue le théorème de Moore-Aronszajn [2] : pour tout noyau symétrique défini positif K, il existe un unique RKHS $\mathcal{H}$ dont K est le noyau reproduisant. Cet espace est construit comme la complétion de :

$$\mathcal{H} = \bigcup_{n\in\mathbb{N}} \Big\{ f : \mathcal{X} \to \mathbb{R} \; ; \; \exists x^1, \dots, x^n \in \mathcal{X}, \alpha_1, \dots, \alpha_n \in \mathbb{R} \\ \text{tel que } f = \sum_{i=1}^{n} \alpha_i K(x^i, \cdot),$$

pour la norme induite par le produit scalaire :

$$\left\langle \sum_{i=1}^{n} \alpha_i K(x_i, \cdot), \sum_{i=1}^{n} \alpha'_i K(x'_i, \cdot) \right\rangle_{\mathcal{H}} = \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha'_j K(x_i, x'_j).$$

Interpolation à noyau

Soit f appartenant à un RKHS $\mathcal{H}$, dont le noyau reproduisant est noté K et $x^1, \dots, x^n \in \mathcal{X}$. On définit $\widetilde{f}$ comme le projeté orthogonal de f sur le sous-espace :

$$\mathcal{H}_{x^1,\dots,x^n} = \text{Vect}\left\{ K(x^1, \cdot), \dots, K(x^n, \cdot) \right\}.$$

On a alors ([5], Proposition 2.7) que $\widetilde{f}$ est l'interpolateur de f de norme minimale, c'est-à-dire que $\widetilde{f}$ est solution de :

$$\underset{g\in\mathcal{H}_{x^1,\dots,x^n}}{\text{argmin}} \; \|g\|_{\mathcal{H}}$$

sous les contraintes :

$$g(x^i) = f(x^i) \;\; \forall i = 1, \dots, n.$$

De plus, on a l'expression suivante pour $\widetilde{f}$:

$$\forall x \in \mathbb{R}^p, \;\; \widetilde{f}(x) = y^t \Sigma^{-1} u(x)$$

où :

$$y = \Big(f(x^1), \dots, f(x^n) \Big)^t,$$

$$u(x) = \Big(K(x^1, x), \dots, K(x^n, x) \Big)^t,$$

et Σ est la matrice symétrique d'ordre n dont le coefficient (i, j) vaut :

$$\Sigma_{i,j} = K(x^i, x^j).$$

La comparaison avec (1.3.3) fait apparaître que, au niveau de la définition du métamodèle, l'interpolation par un noyau K est équivalente à un krigeage dont la fonction de covariance est K. L'étude approfondie du lien entre ces deux méthodes est faite dans [95]. Cependant, les interprétations du métamodèle $\widetilde{f}$ (comme espérance conditionnelle ou comme projeté orthogonal) diffèrent, tout comme les hypothèses de validité du métamodèle : réalisation d'un processus gaussien (1.3.1) dans un cas, et appartenance à un RKHS dans l'autre). À ce titre, signalons que cette dernière hypothèse est aisée à vérifier dans certains cas, grâce au Théorème 1 de [95], qui donne des conditions suffisantes sur K pour que le RKHS associé à K soit un espace de Sobolev. Dans de tels cas, l'hypothèse d'appartenance au RKHS peut se vérifier en testant simplement l'intégrabilité des carrés des dérivées partielles de f.

Dans la pratique, l'espace $\mathcal{H}$ sous-jacent à l'interpolateur est donné à partir du noyau K, choisi en fonction de la régularité de la fonction à estimer ; ainsi le noyau peut être par exemple de type exponentiel :

$$K(x, x') = \prod_{j=1}^{p} \exp\left(-\theta_j \left|x_j - x'_j\right|^{\alpha_j}\right),$$

pour des paramètres $\theta_1, \ldots, \theta_p, \alpha_1, \ldots, \alpha_p$ fixés.

Certains choix de noyaux permettent également de traduire des propriétés du modèle, et ainsi d'améliorer les performances de l'interpolation : par exemple, utilisation d'une propriété de moyenne nulle [30], ou du faible nombre d'interactions entre les variables d'entrée [70].

Toujours sur un plan pratique, le calcul de $\widetilde{f}(x)$ pour un $x \in \mathcal{X}$ nécessite seulement d'avoir à disposition un échantillon d'apprentissage :

$$\{(x^1, f(x^1)), \ldots, (x^n, f(x^n))\},$$

qui peut être construit à l'aide de n évaluations de f.

Analyse d'erreur

Soit $x \in \mathcal{X}$, on a :

$$\begin{aligned} \left|f(x) - \widetilde{f}(x)\right| &= \left|\langle f, K(x,\cdot)\rangle_{\mathcal{H}} - \sum_{i=1}^{n} \left\langle f, K(x^i,\cdot)\left(\Sigma^{-1}u(x)\right)_i\right\rangle_{\mathcal{H}}\right| \\ &\leq \|f\|_{\mathcal{H}} \left\|K(x,\cdot) - \sum_{i=1}^{n} K(x^i,\cdot)\left(\Sigma^{-1}u(x)\right)_i\right\|_{\mathcal{H}} \end{aligned}$$

par l'inégalité de Cauchy-Schwarz.
De plus,

$$\begin{aligned}
\left\| K(x,\cdot) - \sum_{i=1}^{n} K(x^i,\cdot)\left(\Sigma^{-1}u(x)\right)_i \right\|_{\mathcal{H}}^2 &= \left\| K(x,\cdot) - u(\cdot)^t\Sigma^{-1}u(x) \right\|_{\mathcal{H}}^2 \\
&= \langle K(x,\cdot), K(x,\cdot)\rangle_{\mathcal{H}} \\
&\quad -2\left\langle K(x,\cdot), u(\cdot)^t\Sigma^{-1}u(x)\right\rangle_{\mathcal{H}} \\
&\quad +\left\langle u(\cdot)^t\Sigma^{-1}u(x), u(\cdot)^t\Sigma^{-1}u(x)\right\rangle_{\mathcal{H}} \\
&= K(x,x) - 2u(x)^t\Sigma^{-1}u(x) \\
&\quad + \sum_{1\le i,j\le n} K(x^i,x^j)\left(\Sigma^{-1}u(x)\right)_i\left(\Sigma^{-1}u(x)\right)_j \\
&= K(x,x) - 2u(x)^t\Sigma^{-1}u(x) \\
&\quad +(\Sigma^{-1}u(x))^t\Sigma(\Sigma^{-1}u(x)) \\
&= K(x,x) - 2u(x)^t\Sigma^{-1}u(x) \\
&\quad +u(x)^t\Sigma^{-1}u(x) \\
&= K(x,x) - u(x)^t\Sigma^{-1}u(x).
\end{aligned}$$

On a donc la majoration d'erreur suivante :

$$\forall x \in \mathcal{X},\ \left|f(x) - \widetilde{f}(x)\right| \le \|f\|_{\mathcal{H}}\sqrt{K(x,x) - u(x)^t\Sigma^{-1}u(x)}$$

Faisons plusieurs remarques :

– La borne d'erreur est proportionnelle à la norme de f dans $\mathcal{H}$.
Signalons que, pour les noyaux invariants par translation, c'est-à-dire les noyaux K tels qu'il existe une fonction $k : \mathbb{R}^p \to \mathbb{R}$ telle que :

$$\forall x, x' \in \mathcal{X},\ \ K(x,x') = k(x-x'),$$

la norme $\|\cdot\|_{\mathcal{H}}$ peut être reliée aux coefficients de Fourier de f (plus ces coefficients décroissent lentement, plus $\|f\|_{\mathcal{H}}$ est grande [5, Théorème 2.6]), cette norme est donc d'autant plus élevée que f est « oscillante ». Toutes choses égales par ailleurs, l'interpolation (en tous cas la borne d'erreur) est donc moins bonne pour une fonction très oscillante, ce qui est conforme à l'intuition.

– L'autre terme de la borne :

$$\pi_{x^1,\ldots,x^n}(x) = \sqrt{K(x,x) - u(x)^t\Sigma^{-1}u(x)}$$

s'appelle fonction puissance et ne dépend pas de la fonction à évaluer. Il est calculable numériquement. Ceci fait qu'il suffit de connaître $\|f\|_{\mathcal{H}}$ pour pouvoir calculer cette borne.

Lorsque cette norme n'est pas connue, nous avons proposé (chapitre 2, section 2.5) de l'évaluer à partir d'un second échantillon d'apprentissage de taille n_τ :

$$\left\{\left(x_\tau^1, f(x_\tau^1)\right), \ldots, (x_\tau^{n_\tau}, f(x_\tau^{n_\tau}))\right\}$$

en utilisant :

$$\widetilde{\|f\|}_{\mathcal{H}} = \max_{i=1,\ldots,n_\tau} \frac{f(x_\tau^i) - \widetilde{f}(x_\tau^i)}{\pi_{x^1,\ldots,x^n}(x_\tau^i)}.$$

On obtient ainsi un indicateur ponctuel explicitement calculable de l'erreur métamodèle.

– Par ailleurs, [65] donne, sous des conditions de régularité de f, de K, de $\mathcal{X}$ et du plan d'expérience utilisé pour $\{x^1, \ldots, x^n\}$, une majoration de la vitesse de décroissance de $\pi_{x^1,\ldots,x^n}(x)$ en fonction de n. Cette majoration est de la forme :

$$\pi_{x^1,\ldots,x^n}(x) \leq Ce^{-\mathcal{K}/h}$$

où C et $\mathcal{K}$ sont des constantes, et :

$$h = \sup_{z\in\mathcal{X}} \left[\min_{i=1,\ldots,n} \left\|x^i - z\right\|\right]$$

pour une norme $\|\cdot\|$ sur $\mathcal{X}$.

Le nombre h est relié au nombre de points $n^*(\varepsilon)$ d'un ε-recouvrement de $\mathcal{X}$:

$$n^*(\varepsilon) = \min\{m \in \mathbb{N}^* \mid \exists (z_1, \ldots, z_k) \in \mathcal{X} \text{ t.q. } \forall z \in \mathcal{X}, \exists i \in \{1, \ldots, k\} \text{ t.q. } \|z - z_i\| \leq \varepsilon\}.$$

En d'autres termes, $n^*(\varepsilon)$, connu sous le nombre de recouvrement de $\mathcal{X}$, est le nombre minimal d'éléments d'un ensemble E tel que tout point de $\mathcal{X}$ soit à distance inférieure à ε d'un point de E.

On peut démontrer que lorsque $\mathcal{X}$ est un compact de $\mathbb{R}^p$, il existe deux constantes A et B telles que :

$$A\varepsilon^{-p} \leq n^*(\varepsilon) \leq B\varepsilon^{-p}.$$

Ainsi, en supposant que le plan d'expérience $\{x^1, \ldots, x^n\}$ soit choisi optimalement, nous avons, pour une constante B' :

$$h \leq B' n^{-1/p}.$$

Cette majoration *a priori* de l'erreur sera utilisée en section 3.4.4 afin d'illustrer les résultats du chapitre 3.

1.3.3 Métamodélisation par base réduite

Contexte et hypothèses

Contrairement aux deux autres méthodes de métamodélisation présentées plus haut, la méthode base réduite, que nous décrivons dans cette section, est une méthode intrusive, ce qui signifie qu'elle fait des hypothèses fortes sur la structure de la sortie et du modèle à réduire. Les hypothèses nécessaires à cette méthode, telle que nous allons la présenter, sont les suivantes :

1. On suppose que la sortie f est une fonction d'une variable d'état dépendante des paramètres d'entrée $u = u(X)$; en d'autres termes on a :
$$f(X) = f(u(X)).$$

2. On suppose que :
$$\forall X, \;\; u(X) \in H,$$
où H est un espace de Hilbert de dimension finie ; typiquement, H est la discrétisation d'un espace fonctionnel (par exemple, H est un espace éléments finis $\mathbf{P}^1$).

3. On suppose que la variable d'état $u = u(X)$ est solution d'un problème variationnel linéaire sur H :
$$\forall X, \;\; a(u(X), v; X) = \psi(v), \;\; \forall v \in H \tag{1.3.4}$$
où $a(\cdot, \cdot; X)$ et ψ sont, respectivement, une forme bilinéaire symétrique et une forme linéaire sur H.
Ce problème variationnel est, de manière typique, la discrétisation de la forme faible d'une équation aux dérivées partielles.

4. On suppose que la forme bilinéaire $a(\cdot, \cdot, X)$ est coercive :
$$\forall X, \;\; \exists \alpha(X) \text{ tel que } a(u, v, X) \geq \alpha(X) \, \|u\| \, \|v\| \, \forall u, v \in H. \tag{1.3.5}$$

5. On suppose que la dépendance en X de a peut se paramétriser sous forme dite *affine*, c'est à dire qu'il existe $Q \in \mathbb{N}$, des fonctions $\Theta_1, \ldots, \Theta_Q$,

à valeurs réelles, et des formes bilinéaires $a_1, \ldots, a_Q$ indépendantes de X, telles que :

$$\forall X, \quad a(\cdot, \cdot; X) = \sum_{q=1}^{Q} \Theta_q(X) a_q(\cdot, \cdot).$$

Certaines de ces hypothèses peuvent être relaxées. Par exemple, la méthode base réduite a été étendue à des problèmes variationnels non linéaires et/ou comportant une dimension temporelle (tels que les équations linéaires paraboliques [45, 108], l'équation de Boussinesq [61], ou le système de Navier-Stokes incompressible [110], ainsi qu'au chapitre 4 pour l'équation de Burgers). La méthode d'interpolation empirique (aussi appelée méthode *magic points*), décrite dans [44], permet de se ramener au cas où l'hypothèse 4 (existence d'une décomposition affine de a) est vérifiée, moyennant certaines approximations.
La résolution du problème variationnel (1.3.4) se fait généralement en considérant une base $\Phi = \{\phi_1, \ldots, \phi_{\mathcal{N}}\}$ de H, en introduisant les composantes de $u(X)$ dans cette base :

$$u(X) = \sum_{i=1}^{\mathcal{N}} u_i(X) \phi_i,$$

et en cherchant les coefficients comme solution du système linéaire à $\mathcal{N}$ équations et autant d'inconnues :

$$\begin{cases} \sum_{i=1}^{\mathcal{N}} u_i(X) a(\phi_i, \phi_1; X) = \psi(\phi_1) \\ \vdots \\ \sum_{i=1}^{\mathcal{N}} u_i(X) a(\phi_i, \phi_{\mathcal{N}}; X) = \psi(\phi_{\mathcal{N}}) \end{cases}$$

Généralement, la base Φ est choisie en fonction de la régularité du problème variationnel considéré, et non en fonction de la variété paramétrique $\{u(X)\}_X$ sur laquelle évolue la variable d'état lorsque le paramètre d'entrée varie. Ceci entraîne que, généralement, le nombre d'inconnues à trouver ($\mathcal{N}$) est grand, ce qui cause une résolution numérique coûteuse.

Principe de la méthode base réduite

L'idée-clé de la méthode base réduite est d'abord de trouver une famille libre dans H notée $\{\zeta_1, \ldots, \zeta_n\}$, et engendrant un sous-espace de H noté $\widetilde{H}$, puis

de chercher une approximation $\widetilde{u}(X) \in \widetilde{H}$ sous la forme :

$$\widetilde{u}(X) = \sum_{i=1}^{n} \widetilde{u}_i(X)\zeta_i,$$

satisfaisant une version affaiblie de (1.3.4) :

$$a(\widetilde{u}(X), v; X) = \psi(v), \quad \forall v \in \widetilde{H}.$$

Ce dernier problème variationnel peut s'écrire comme un système à n équations et autant d'inconnues :

$$\begin{cases} \displaystyle\sum_{i=1}^{n} \widetilde{u}_i(X) \sum_{q=1}^{Q} a_q(\zeta_i, \zeta_1)\Theta_q(X) = \psi(\zeta_1) \\ \vdots \\ \displaystyle\sum_{i=1}^{n} \widetilde{u}_i(X) \sum_{q=1}^{Q} a_q(\zeta_i, \zeta_n)\Theta_q(X) = \psi(\zeta_n) \end{cases} \tag{1.3.6}$$

La méthode base réduite fonctionne lorsque l'on peut choisir $n \ll \mathcal{N}$ et conserver une approximation $u(X) \approx \widetilde{u}(X)$ satisfaisante. Le gain en temps de calcul est réalisé en remplaçant la résolution du système complet à $\mathcal{N}$ inconnues par un système à n inconnues, ce qui fait que la méthode base réduite se place parmi les méthodes de réduction de dimension.
Remarquons enfin que la sortie réduite $\widetilde{f}$ peut être calculée à partir de $\widetilde{u}$ par la formule :

$$\widetilde{f}(X) := f(\widetilde{u}(X)) = \sum_{i=1}^{n} \widetilde{u}_i(X) f(\zeta_i). \tag{1.3.7}$$

La décomposition offline/online est donc la suivante :

- durant la phase offline (faite une seule fois), la base réduite est choisie (nous verrons en section 1.3.3 comment faire cela), et les éléments indépendants du paramètre X apparaissant dans le système (1.3.6) et dans la formule (1.3.7) sont précalculés et stockés ;
- durant la phase online (qui a lieu pour chaque valeur du paramètre d'entrée X sur lequel évaluer $\widetilde{f}$), le système (1.3.6) est assemblé, à partir des éléments stockés durant la phase offline, ainsi que de l'évaluation des fonctions Θ_q sur le paramètre voulu. Ce système est ensuite résolu afin de trouver les coefficients $\{\widetilde{u}_i(X)\}$, puis, enfin, $\widetilde{f}(X)$ est calculée au moyen de (1.3.7).

Remarquons que la complexité algorithmique de la phase online, bien que dépendante de n et de Q, est totalement indépendante de la dimension du problème d'origine $\mathcal{N}$.
Signalons enfin qu'il existe une autre façon de définir la sortie réduite $\widetilde{f}$; cette autre façon sera motivée et étudiée en partie 1.7.2.

Borne d'erreur

Discutons maintenant de l'obtention de la borne d'erreur entre la sortie du modèle et la sortie du métamodèle, c'est-à-dire du réel $\varepsilon(X)$ tel que :

$$\left|f(X) - \widetilde{f}(X)\right| \leq \varepsilon(X) \quad \forall X.$$

Cette borne d'erreur doit être calculable explicitement à l'aide d'une procédure offline/online efficace, c'est à dire que la complexité de la procédure online ne doit pas dépendre de la dimension du problème à réduire, tout comme les calculs de $\widetilde{u}$ et de $\widetilde{f}$ décrits à la section précédente.
Commençons d'abord par établir une borne d'erreur entre les variables d'état non réduites et réduites, ie. cherchons $\varepsilon_u(X)$ telle que :

$$\|u(X) - \widetilde{u}(X)\| \leq \varepsilon_u(X) \quad \forall X,$$

où $\|\cdot\|$ désigne la norme dont H est muni.
On démontre que :

$$\|u(X) - \widetilde{u}(X)\| \leq \frac{\|r(X)\|_*}{\alpha(X)},$$

où $r(X)$ est la forme linéaire sur H donnant le résidu :

$$r(X) = a(\widetilde{u}(\mu), \cdot; X) - f(\cdot),$$

$\|\cdot\|_*$ désigne la norme duale (norme triple) sur H :

$$\|r\|_* = \sup_{v \in H, \|v\|=1} |r(v)|,$$

et $\alpha(X)$ est la constante de coercivité de $a(\cdot, \cdot; X)$, qui satisfait (1.3.5).
Il est possible de développer ([Janon5, §3.2 et §3.3], ainsi que [72],[51] et [19]) une procédure offline/online efficace pour calculer :
– la valeur exacte de la norme duale du résidu $\|r(X)\|_*$;
– une borne inférieure de la constante de coercivité, ie. $\alpha_{inf}(X)$ tel que :

$$\alpha(X) \geq \alpha_{inf}(X) \quad \forall X.$$

Les détails de ces calculs ne sont pas nécessaires à la compréhension des chapitres suivants de ce manuscrit, car les méthodes de calcul que nous proposons sont différentes et seront précisées à l'intérieur des chapitres.
On obtient ainsi une borne d'erreur calculable sur u :

$$\varepsilon_u(X) = \frac{\|r(X)\|_*}{\alpha_{inf}(X)}.$$

De cette borne sur u peut facilement se déduire une borne sur la sortie du métamodèle :

$$\left|f(X) - \widetilde{f}(X)\right| \leq \|f\|_* \, \varepsilon_u(X),$$

où $\|f\|_*$, indépendante de X, peut être calculée et mémorisée durant la phase offline.
Nous développerons au chapitre 5 une alternative plus efficace à cette dernière borne d'erreur.

Choix de la base réduite

Nous présentons maintenant deux algorithmes de choix d'une base réduite $\mathcal{B} = \{\zeta_1, \ldots, \zeta_n\}$.

Algorithme glouton (*greedy*). Cet algorithme est présenté dans [72] et étudié d'un point de vue théorique dans [12]. L'idée est de construire itérativement $\mathcal{B}$ en rajoutant, à chaque étape, la solution du modèle non réduit, pour la valeur du paramètre maximisant la borne d'erreur ε_u calculée sur un échantillon de jeux de paramètres $\Xi \subset \mathbb{R}^p$ tiré aléatoirement.
L'algorithme est le suivant :

1. Initialiser $\mathcal{B} = \{u(X^1)\}$, pour X^1 paramètre tiré aléatoirement.
2. Tirer aléatoirement $n_\Xi > 0$ paramètres $X^1, \ldots, X^{n_\Xi}$, formant ainsi l'ensemble Ξ.
3. Répéter, pour $i = 2, \ldots, n$:
   (a) Calculer :
   $$X^{*i} = \underset{X \in \Xi}{\operatorname{argmax}} \, \varepsilon_u(X),$$
   où $\varepsilon_u(X)$ est la borne de l'erreur entre u et $\widetilde{u}$ obtenue par la méthode base réduite projetant sur la base $\mathcal{B}$ courante.
   (b) Poser : $\mathcal{B} \leftarrow \mathcal{B} \cup \{X^{i*}\}$ et réorthonormaliser $\mathcal{B}$.

Signalons, en lien avec la question 5 de la section 1.1, que la borne d'erreur ε_u utilisée dans cet algorithme peut être remplacée par une borne d'erreur optimisée sur la sortie, telle la borne utilisant le dual $\varepsilon^{\text{Dual}}$, décrite en section 1.7 ou la borne décrite au chapitre 5.

Algorithme basé sur la POD. Cet algorithme, inspiré par la procédure de POD (*Proper Orthogonal Decomposition* [18], [96]) considère la recherche d'une base orthonormée $\mathcal{B}^*$ minimisant l'erreur quadratique, en moyenne sur X, de projection orthogonale de $u(X)$:

$$\mathcal{B}^* = \underset{\mathcal{B} \text{ famille orthonormée}}{\operatorname{argmin}} \mathbb{E}\left(\|u(X) - \Pi_{\mathcal{B}} u(X)\|^2\right),$$

où $\Pi_{\mathcal{B}}$ désigne la projection orthogonale sur $\mathcal{B}$.
L'espérance ci-dessus ne pouvant pas, en général, être calculée analytiquement, elle est remplacée par une de ses estimations Monte-Carlo :

$$\widehat{\mathcal{B}} = \underset{\mathcal{B} \text{ famille orthonormée}}{\operatorname{argmin}} \sum_{X \in \Xi} \left(\|u(X) - \Pi_{\mathcal{B}} u(X)\|^2\right),$$

où Ξ est un échantillon iid. de la loi de X.
Le calcul de $\widehat{\mathcal{B}}$ peut alors se faire ([112], [Janon5] §4.2) algorithmiquement en calculant d'abord $\#\Xi$ solutions du modèle numérique, puis en trouvant les n vecteurs propres associés aux plus grandes valeurs propres d'une matrice d'ordre $\#\Xi$.

Choix guidé par la sortie (*goal-oriented*). Également en lien avec la question 5 énoncée en section 1.1, on peut vouloir choisir une base $\mathcal{B}$ telle que :

$$\mathcal{B} = \underset{\{\zeta_1,\ldots,\zeta_n\}}{\operatorname{argmin}} \sum_{X \in \Xi} \|f(u(X)) - f(\tilde{u}(X))\|^2 \tag{1.3.8}$$

sous les contraintes suivantes :

$$\begin{cases} \tilde{u}(X) = \sum_{i=1}^{n} \widetilde{u}_i(X)\zeta_i \quad \forall X \in \Xi \\ \sum_{i=1}^{n} \left[\sum_{q=1}^{Q} \Theta_q(X) a_q(\zeta_i, \zeta_p)\right] \tilde{u}_i(X) = \psi(\zeta_p) \quad \forall \mu \in \Xi, \forall p = 1, \ldots, N \\ < \zeta_i, \zeta_p > = \begin{cases} 1 & \text{si } i = p \\ 0 & \text{sinon,} \end{cases} \quad \forall n, m = 1, \ldots, N \end{cases}$$

où, comme précédemment, Ξ est un échantillon de la loi de X.

Cette approche est proposée dans [13], dans le contexte des systèmes dynamiques linéaires, où les auteurs décrivent des affaiblissements de ce problème d'optimisation qui permettent son implémentation numérique afin de choisir la base $\mathcal{B}$ en lien avec la fonctionnelle de sortie f. Cependant, le problème d'optimisation obtenu est *a priori* difficile (non convexe, avec des minima non globaux), ce qui fait que cette méthode nécessite une charge de calcul supérieure à celle de la POD classique.

1.4 Introduction au Chapitre 2 Prise en compte de l'erreur métamodèle

Dans cette section, nous introduisons le chapitre 2, qui est constitué de l'article accepté [Janon1] ; c'est à l'occasion de l'écriture de cet article qu'ont été diffusées les contributions aux packages R `sensitivity` et `CompModSA` présentées dans la liste des travaux en page 13. Le but de ce chapitre est d'apporter une réponse aux questions 1 et 3 soulevées dans l'introduction en section 1.1. Nous rappelons que ces deux questions visent à quantifier l'erreur d'estimation par Monte-Carlo (question 1) et l'impact du remplacement d'un modèle, lorsque celui-ci est coûteux à évaluer, par un métamodèle approchant (question 3).
Nous commençons par développer notre problématique, puis nous présentons deux approches déjà proposées dans la littérature. Nous terminons par un résumé de l'approche que nous proposons et des résultats obtenus.

1.4.1 Problématique

Dans ce chapitre, nous revenons à l'estimation des indices de Sobol par méthode de Monte-Carlo, en utilisant l'estimateur $\widehat{S}_i$:

$$\widehat{S}_i = \frac{\frac{1}{N}\sum_{k=1}^{N} Y_k Y'_k - \left(\frac{1}{N}\sum_{k=1}^{N} Y_k\right)\left(\frac{1}{N}\sum_{k=1}^{N} Y'_k\right)}{\frac{1}{N}\sum_{k=1}^{N} Y_k^2 - \left(\frac{1}{N}\sum_{k=1}^{N} Y_k\right)^2}. \tag{1.4.1}$$

où $\{Y_k\}_{k=1,\dots,N}$ et $\{Y'_k\}_{k=1,\dots,N}$ sont deux échantillons iid. des variables Y et Y' définies respectivement par :

$$Y = f(X_1, \dots, X_p) \text{ et } Y' = f(X'_1, \dots, X'_{i-1}, X_i, X'_{i+1}, \dots, X'_p), \tag{1.4.2}$$

où f est le modèle à étudier, et $X = (X_1, \dots, X_p)$ et $X' = (X'_1, \dots, X'_p)$ sont deux copies indépendantes de la loi des paramètres d'entrée.

Comme précisé en introduction de ce chapitre, le nombre d'évaluations de la fonction f à faire pour calculer une réalisation de $\widehat{S}_i$ peut être grand et on peut être amené, lorsque le temps de calcul nécessaire à ces évaluations est trop important, à remplacer f par un métamodèle, que nous noterons $\widetilde{f}$, qui approche f tout en étant plus rapide à calculer. On définit alors $\widetilde{Y}$ et $\widetilde{Y}'$ par :

$$\widetilde{Y} = \widetilde{f}(X_1, \dots, X_p) \text{ et } \widetilde{Y}' = \widetilde{f}(X'_1, \dots, X'_{i-1}, X_i, X'_{i+1}, \dots, X'_p).$$

Si l'on utilise $\widehat{S}_i$ en remplaçant directement Y et Y' par, respectivement, $\widetilde{Y}$ et $\widetilde{Y}'$, on estime l'indice de sensibilité du métamodèle $\widetilde{f}$ et non celui du vrai modèle. Notre objectif est de proposer des approches et de la littérature pour répondre à la Question 3 formulée dans la Section 1.1, à savoir rendre compte de l'erreur induite sur l'estimation des indices de sensbilité lorsque le vrai modèle f est remplacé par le métamodèle $\widetilde{f}$. L'évaluation de l'erreur de métamodèle doit être combinée à celle de l'erreur d'estimation Monte-Carlo, faisant ainsi le lien avec la Question 1 de cette même section.
Avant de résumer l'approche que nous proposons dans le Chapitre 2, nous décrivons deux approches existant dans la littérature.

1.4.2 Approche existante 1 : utilisation du krigeage

L'approche que nous allons présenter maintenant est apparue dans [66]. Cette approche nécessite que le métamodèle utilisé soit un métamodèle de krigeage, décrit en section 1.3.1. Les auteurs proposent alors, au §2.3, d'*op. cit.*, de considérer les deux indices suivants :

$$\widetilde{S}_i^1 = \frac{\mathrm{Var}_{X_i}\mathbb{E}_X\left(\widetilde{f}(X)|X_i\right)}{\mathrm{Var}_X\left(\widetilde{f}(X)\right)},$$

noté S_i dans l'article original, et est égal à l'indice de Sobol du premier ordre si le vrai modèle était $\widetilde{f}$.
Le deuxième indice considéré est le suivant :

$$\widetilde{S}_i^2(\omega) = \frac{\mathrm{Var}_{X_i}\mathbb{E}_X(Z(X,\omega)|X_i)}{\mathbb{E}_{\omega'\in\Omega}(\mathrm{Var}_X Z(X,\omega'))},$$

noté $\widetilde{S}_i$, et dont la loi, conditionnellement à $Z(x^1) = f(x^1), \dots, Z(x^n) = f(x^n)$, permet de définir un indice ponctuel de sensibilité et une indication de sa précision (par, respectivement, l'espérance et la variance de cette loi

conditionnelle). Grâce à $\widetilde{S}_i^2$, des intervalles de confiance peuvent être également produits, à l'aide des quantiles empiriques de cette distribution, qui peut être simulée numériquement par une méthode décrite dans l'article cité. Les auteurs ont comparé, sur des exemples où les vrais indices de sensibilité du modèle sont connus, l'estimation de S_i par $\widetilde{S}_i^1$ d'une part, et par l'espérance, conditionnellement à l'échantillon d'apprentissage, de $\widetilde{S}_i^2$ d'autre part, et montrent que ce deuxième choix commet une erreur quadratique moyenne d'estimation plus faible. Par ailleurs, il est observé que les intervalles de confiance pour S_i ont un niveau de risque se rapprochant du niveau voulu lorsque la taille (n) de l'échantillon d'apprentissage augmente (pour de petites valeurs de n, donc un métamodèle peu fidèle, le niveau de risque est bien souvent supérieur à ce qui est demandé), sauf dans certains cas, où le niveau réel de risque reste à 100%, pour un risque cible de 10%, ce qui indique que la méthode manque de robustesse dans ces cas là.
En guise de discussion, nous faisons les deux remarques suivantes sur cette approche :

– elle se limite aux métamodèles de krigeage, alors que d'autres méthodes de métamodélisation sont disponibles et pourraient, suivant le cas traité, s'avérer plus efficaces ;
– l'erreur commise en estimant les hyperparamètres du modèle n'est pas prise en compte.

1.4.3 Approche existante 2 : bootstrap de métamodèle

L'autre approche que nous considérons maintenant est présentée dans [100]. Elle consiste à « bootstrapper » le métamodèle afin de mesurer sa variabilité et son impact sur l'estimation des indices de Sobol.
Plus préciser, nous considérons un métamodèle $\widetilde{f} = \widetilde{f}_{\mathcal{D}}$, fonction d'un échantillon d'apprentissage $\mathcal{D}$:

$$\mathcal{D} = \{(x^1, f(x^1)), \dots, (x^n, f(x^n))\}.$$

Pour $R \in \mathbb{N}$, on génère un échantillon de R réplications de $\widehat{S}_i$ en répétant, pour $r = 1, \dots, R$, les étapes suivantes :

1. on tire un n-échantillon $\{x_*^1, \dots, x_*^n\}$ depuis la loi du paramètre d'entrée X ;
2. on calcule une version « bruitée » de l'échantillon d'apprentissage :

$$\mathcal{D}_* = \{(x_*^1, f_*(x^1)), \dots, (x_*^n, f_*(x^n))\}.$$

où, pour $k = 1, \ldots, n$:

$$f_*(x_*^k) = \widetilde{f}_{\mathcal{D}}(x_*^k) + e_k^r,$$

où e_k^r est échantillonné suivant la loi équirépartie sur l'ensemble des résidus du métamodèle :

$$\left\{ \widetilde{f}_{\mathcal{D}}(x^l) - f(x^l),\ l = 1, \ldots, n \right\};$$

3. on calcule une réplication $\widehat{S}_i^{r*}$ de $\widehat{S}_i$ en utilisant (1.4.1) (ou toute autre formule estimant S_i par Monte-Carlo) et (1.4.2) où f est remplacée par $f_{\mathcal{D}_*}$.

L'ensemble

$$\mathcal{R} = \{\widehat{S}_i^{1*}, \ldots, \widehat{S}_i^{R*}\}$$

de réplications est alors utilisé pour produire un intervalle de confiance, en suivant par exemple une des méthodes décrite en Section 1.2.4. Cette méthode est bien entendu intéressante si le nombre n de points dans l'échantillon d'apprentissage peut être choisi nettement inférieur à la taille (N) de l'échantillon Monte-Carlo utilisé pour calculer $\widehat{S}_i$.

Par rapport à l'approche décrite précédemment, cette méthode laisse plus de liberté au niveau de la construction du métamodèle (les auteurs la testent d'ailleurs pour plusieurs types de métamodèles), mais présente selon nous deux inconvénients :

– la prise en charge de l'erreur de métamodèle nécessite une hypothèse sur la loi de l'erreur ;
– le temps de calcul pour construire chaque métamodèle répliqué $f_{\mathcal{D}_*}$ à partir de $\mathcal{D}_*$ peut être important, et peut, étant répété R fois, rendre prohibitif le temps nécessaire au calcul de l'intervalle de confiance.

Par ailleurs, dans les tests numériques que nous avons effectués et que nous présentons au Chapitre 2, Table 2.3, nous avons observé que les intervalles de confiance produits avec cette méthode peuvent avoir un niveau de risque supérieur à celui demandé (98% au lieu de 5%, pour l'exemple le plus extrême).

Remarquons enfin que l'approche décrite ci-dessus n'est évidemment pas pertinente pour les métamodèles interpolants, qui sont ceux satisfaisant la propriété :

$$\forall k = 1, \ldots, n, \quad \widetilde{f}_{\mathcal{D}}(x^k) = f(x^k),$$

puisqu'alors tous les résidus sont nuls. Ainsi, pour le krigeage, qui produit des métamodèles interpolants, les auteurs proposent au §3.5 d'utiliser une méthode différente : il s'agit, pour obtenir R réplications de $\widehat{S}_i$, de répéter les étapes suivantes pour $r = 1, \ldots, R$:

1. Générer aléatoirement deux échantillons $\{X^k\}_{k=1,\ldots,N}$ et $\{X^{k\prime}\}_{k=1,\ldots,N}$ suivant la loi de X.
2. Pour $k = 1, \ldots, N$, tirer Y_k suivant la loi *a posteriori* de $f(X^k)$, à savoir (en reprenant les notations de la section 1.3.1 :
$$\mathcal{N}\left(m(X^k), R(X^k, X^k) - k_{\mathcal{D}}(X^k)^t \Sigma_{\mathcal{D}}^{-1} k_{\mathcal{D}}(X^k)\right),$$
et Y'_k suivant la loi *a posteriori* de $f(X_1^{k\prime}, \ldots, X_{i-1}^{k\ \prime}, X_i^k, X_{i+1}^{k\ \prime}, \ldots, X_p^{k\prime})$:
$$\mathcal{N}\left(m(X_{pf}^k), R(X_{pf}^k, X_{pf}^k) - k_{\mathcal{D}}(X_{pf}^k)^t \Sigma_{\mathcal{D}}^{-1} k_{\mathcal{D}}(X_{pf}^k)\right),$$
où
$$X_{pf}^k = \left(X_1^{k\prime}, \ldots, X_{i-1}^{k\ \prime}, X_i^k, X_{i+1}^{k\ \prime}, \ldots, X_p^{k\prime}\right).$$
3. Calculer la $r^{\text{ème}}$ réplication de $\widehat{S}_i$ en utilisant (1.4.1).

On termine en fournissant un intervalle de crédibilité bayésien de risque α constitué par les quantiles $\alpha/2$ et $1 - \alpha/2$ de l'ensemble des R réplications de $\widehat{S}_i$ venant d'être calculées.

1.4.4 Notre approche

Hypothèse d'existence d'une borne ponctuelle. Nous décidons de quantifier l'erreur de métamodèle au moyen d'une borne d'erreur ponctuelle ε, satisfaisant :
$$\forall x, \quad \left|f(x) - \widetilde{f}(x)\right| \leq \varepsilon(x).$$
Cette borne ponctuelle doit être calculable explicitement (numériquement), pour un paramètre d'entrée x donné, en un temps comparable à celui nécessaire au calcul d'une évaluation du métamodèle $\widetilde{f}$ (en tous cas, il doit être sensiblement inférieur au temps de calcul de f), faute de quoi la méthode n'aura aucun intérêt pratique.
La méthode base réduite, décrite en section 1.3.3, est une méthode de métamodélisation fournissant une telle borne d'erreur. L'interpolation par noyau reproduisant (RKHS), est une autre méthode de métamodélisation, décrite à la section 1.3.2, disposant également d'une borne d'erreur ; cependant cette borne d'erreur n'est pas explicitement calculable en totalité, et l'on doit se contenter d'une borne *approchée* (un indicateur d'erreur).

Principe de la méthode (Section 2.3.1). On commence par tirer deux échantillons iid. $\{X^k\}_{k=1,\dots,N}$ et $\{X^{k\prime}\}_{k=1,\dots,N}$, puis on calcule un N-échantillon de sorties métamodèles et de bornes d'erreur suivant un plan *pick-freeze* :

$$\widetilde{Y}_k = \widetilde{f}(X^k), \quad \widetilde{Y}'_k = \widetilde{f}(X_1^{k\prime}, \dots, X_{i-1}^{k\prime}, X_i^k, X_{i+1}^{k\prime}, \dots, X_p^{k\prime}),$$

$$\varepsilon_k = \varepsilon(X^k), \quad \varepsilon'_k = \varepsilon(X_1^{k\prime}, \dots, X_{i-1}^{k\prime}, X_i^k, X_{i+1}^{k\prime}, \dots, X_p^{k\prime}).$$

Aucune évaluation de f n'étant disponible, ces quatre N-échantillons ne permettent pas d'évaluer l'estimateur $\widehat{S}_i$ sur le vrai modèle. L'idée centrale est d'encadrer $\widehat{S}_i$ par deux estimateurs $\widehat{S}_i^m$ et $\widehat{S}_i^M$ qui sont fonctions des échantillons de sorties du métamodèle et des bornes d'erreur :

$$\begin{aligned}\widehat{S}_i^m \left(\left(\widetilde{Y}_k\right)_{k=1,\dots,N}, \left(\widetilde{Y}'_k\right)_{k=1,\dots,N}, (\varepsilon_k)_{k=1,\dots,N}, (\varepsilon'_k)_{k=1,\dots,N} \right) \\ \leq \widehat{S}_i\left((Y_k), (Y'_k)\right) \leq \\ \widehat{S}_i^M \left(\left(\widetilde{Y}_k\right)_{k=1,\dots,N}, \left(\widetilde{Y}'_k\right)_{k=1,\dots,N}, (\varepsilon_k)_{k=1,\dots,N}, (\varepsilon'_k)_{k=1,\dots,N} \right).\end{aligned}$$

Supposons que nous puissions calculer numériquement ces deux estimateurs. On peut alors calculer R réplications bootstrap de chacun de ces estimateurs :

$$\mathcal{R}_m = \left\{ \widehat{S}_i^m \left(\left(\widetilde{Y}_k\right)_{k\in L_r}, \left(\widetilde{Y}'_k\right)_{k\in L_r}, (\varepsilon_k)_{k\in L_r}, (\varepsilon'_k)_{k\in L_r} \right), \ \ r = 1, \dots, R \right\}$$

et :

$$\mathcal{R}_M = \left\{ \widehat{S}_i^M \left(\left(\widetilde{Y}_k\right)_{k\in L_r}, \left(\widetilde{Y}'_k\right)_{k\in L_r}, (\varepsilon_k)_{k\in L_r}, (\varepsilon'_k)_{k\in L_r} \right), \ \ r = 1, \dots, R \right\}$$

où $L_1, \dots, L_R$ sont R listes de longueur N, indépendantes et tirées aléatoirement avec remise dans $\{1, \dots, N\}$.

On peut alors fixer un niveau de risque et utiliser une technique de bootstrap (cf. 1.2.4) pour produire un intervalle de confiance $[I_1^m; I_2^m]$ à partir de l'ensemble $\mathcal{R}_m$, et un intervalle de confiance $[I_1^M; I_2^M]$. On combine alors ces deux intervalles en un intervalle de confiance $\widetilde{I} = [I_1^m; I_2^M]$.

On peut facilement démontrer que, si on utilise la technique du boostrap classique, on a :

$$P\left(S_i \ni \widetilde{I}\right) \geq P\left(S_i \ni I\right), \tag{1.4.3}$$

où I désigne l'intervalle bootstrap construit (hypothétiquement) à partir de l'estimateur $\widehat{S}_i$.

Autrement dit, si la technique de bootstrap utilisée fonctionne correctement (ie., donne un intervalle ayant le niveau de risque fixé) pour estimer S_i à partir du vrai modèle, alors notre technique donnera un intervalle *conservatif*.
Remarquons que l'utilisation du bootstrap avec correction de biais ne permet de démontrer (1.4.3) dans tous les cas. Cependant, nos essais numériques montrent qu'en pratique, les résultats donnés par le bootstrap classique et celui avec correction de biais ne diffèrent pas significativement. Ceci est à rapprocher de la propriété de normalité asymptotique (introduite section 1.5 et démontrée au chapitre 3) qui montre que le biais est asymptotiquement négligeable devant l'écart-type de l'estimateur.

Nous terminons le résumé de notre approche en donnant deux choix possibles pour les estimateurs encadrants $\widehat{S}_i^m$ et $\widehat{S}_i^M$, ainsi qu'une brève description des résultats obtenus avec chacun d'entre eux.

Encadrement analytique (*loc. cit.*). On remarque que $\widehat{S}_i$ est l'argmin. d'un trinôme (noté R dans l'article), qui peut être encadré par deux trinômes (notées R_{inf} et R_{sup}) dont les coefficients sont des fonctions des sorties métamodèles et des bornes d'erreur. On peut alors en déduire un encadrement de $\widehat{S}_i$ qui est calculable explicitement par un algorithme rapide.
Nous avons testé l'estimation d'intervalles de confiance en utilisant cet encadrement à l'aide du métamodèle certifié décrit dans le Chapitre 4. Les vraies valeurs analytiques des indices n'étant pas connues pour ce modèle, nous avons vérifié la justesse de notre méthode en calculant des « vérités terrain » à l'aide d'un estimateur Monte-Carlo construit sur un large échantillon et utilisant le vrai modèle numérique. Nous avons constaté que les intervalles de confiance avaient bien le niveau requis, même lorsque le métamodèle est peu fidèle. De plus, nous montrons sur cet exemple que l'utilisation d'un métamodèle base réduite pour estimer permet des gains significatifs en temps de calcul par rapport à l'utilisation du vrai modèle, tout en ayant une certification sur la perte de précision. Nous avons, enfin, testé l'utilisation de la méthode existante basée sur le bootstrap métamodèle décrite plus haut, à l'aide de l'implémentation de référence faite par Storlie dans le package `CompModSA` [116] et en utilisant un échantillon d'apprentissage de la même taille que celui nécessaire à la construction du métamodèle base réduite utilisé ; notre méthode apporte un résultat plus précis (des intervalles de

confiance moins larges, à niveau fixé) tout en nécessitant un temps de calcul nettement plus court. Ce test valide donc le choix du métamodèle base réduite et de notre méthode de calcul d'intervalles de confiance.
Nous avons ensuite voulu effectuer la même comparaison sur un métamodèle obtenu par interpolation à noyau (voir section 1.3.2). Notre méthode s'est alors avérée décevante car, les bornes d'erreur métamodèle ayant une plus grande magnitude dans ce cas, l'encadrement de $\widehat{S}_i$ s'est avéré trivial (c'est-à-dire plus large que $[0;1]$). Ceci nous a amenés à développer, pour de tels cas, une autre méthode de calcul de bornes pour $\widehat{S}_i$.

Encadrement avec lissage (Section 2.3.2). Commençons par indiquer pourquoi l'encadrement présenté plus haut est souvent trop pessimiste. Définissons le pavé de $\mathbb{R}^{2N}$ suivant :

$$\mathcal{Z} = \prod_{k=1}^{N} \left[\widetilde{Y}_k - \varepsilon_k; \widetilde{Y}_k + \varepsilon_k\right] \times \prod_{k=1}^{N} \left[\widetilde{Y}'_k - \varepsilon'_k; \widetilde{Y}'_k + \varepsilon'_k\right],$$

et la fonction (les sommes portent sur k entre 1 et N) :

$$\psi_i(\mathbf{z}) = \frac{\frac{1}{N}\sum z_k z'_k - (\frac{1}{N}\sum z_k)(\frac{1}{N}\sum z'_k)}{\frac{1}{N}\sum (z_k)^2 - (\frac{1}{N}\sum z_k)^2},$$

pour $\mathbf{z} = (z_1, \ldots, z_N, z'_1, \ldots, z'_N)$.
Comme :

$$\widehat{S}_i = \psi_i(Y_1, \ldots, Y_N, Y'_1, \ldots, Y'_N),$$

il est clair que l'encadrement optimal de $\widehat{S}_i$ est réalisé par le choix :

$$\widehat{S}_i^m = \min_{\mathbf{z}\in\mathcal{Z}} \psi_i(\mathbf{z}) \quad \text{et} \quad \widehat{S}_i^M = \max_{\mathbf{z}\in\mathcal{Z}} \psi_i(\mathbf{z}).$$

En calculant, dans l'exemple RKHS évoqué plus haut, la solution $\mathbf{z}^*$ de ces problèmes d'optimisation, nous nous sommes aperçu que le « pire cas » correspondait à celui d'une fonction f très peu régulière (la fonction f se retrouvant à partir de ses valeurs échantillonnées $\mathbf{z}^*$). En général, les sorties considérées bénéficient d'un minimum de régularité en fonction des paramètres de sortie. Ce pire cas est donc peu vraisemblable et doit être évité. Ceci suggère d'utiliser un « encadrement » pénalisant les $\mathbf{z}$ correspondant à des fonctions peu lisses :

$$\mathbf{z}^m = \operatorname*{argmin}_{\mathbf{z}\in\mathcal{Z}} \psi_i(\mathbf{z}) + \lambda\Pi(\mathbf{z}) \quad \text{et} \quad \mathbf{z}^M = \operatorname*{argmax}_{\mathbf{z}\in\mathcal{Z}} \psi_i(\mathbf{z}) - \lambda\Pi(\mathbf{z}),$$

où Π est une fonction de pénalisation et $\lambda > 0$ un coefficient de pénalisation. On utilise ensuite :

$$\widehat{S}_i^m = \psi_i(\mathbf{z}^m) \quad \text{et} \quad \widehat{S}_i^M = \psi_i(\mathbf{z}^M).$$

En pratique, on utilise un algorithme de minimisation numérique (du type quasi Newton, L-BFGS-B) pour calculer $\mathbf{z}^m$ et $\mathbf{z}^M$. Le problème d'optimisation résolu est de dimension importante $(2N)$; malgré cela, cet algorithme donne des résultats corrects en des temps raisonnables. Nous utilisons une pénalité particulière, définie en (2.3.17), qui permet d'avoir un gradient de la fonction objectif rapide à évaluer.
Par rapport à l'encadrement analytique, cet encadrement est moins rigoureux et fait intervenir un paramètre de pénalisation à choisir correctement. Cependant, il nous a permis de traiter de manière correcte l'exemple de l'interpolation RKHS, tout en produisant des intervalles de confiance non triviaux ayant le niveau de risque assez proche de celui requis, ce qui n'a pas été le cas lorsque nous avons testé le package `CompModSA` avec krigeage.

1.5 Introduction au Chapitre 3 Propriétés asymptotiques d'estimateurs de l'indice de Sobol

Cette section introduit le chapitre 3, constitué de l'article soumis [Janon2]. Ce chapitre traite des propriétés asymptotiques de deux estimateurs Monte-Carlo construits sur un plan pick-freeze, et vise à répondre aux questions 1 (quantification de l'erreur d'estimation), 2 (recherche d'un estimateur Monte-Carlo optimal) et 3 (impact du remplacement du modèle par un métamodèle dans l'estimation des indices de Sobol) de la section 1.1. Ces propriétés sont d'abord étudiées dans un contexte où le vrai modèle à étudier est disponible, puis étendues dans le cas où celui-ci est remplacé par un métamodèle. Après avoir exposé la problématique de l'article, nous donnons, en sections 1.5.2 et 1.5.3, les outils que nous avons utilisés. Enfin, nous résumons, en section 1.5.4, les résultats que nous avons obtenus dans ce travail.

1.5.1 Problématique

Nous nous plaçons à nouveau dans le cadre de l'estimation des indices de Sobol par une méthode de Monte-Carlo pick-freeze, plus précisément en utilisant les estimateurs suivants, que nous avons déjà introduits auparavant :

$$\widehat{S}_{N,i} = \frac{\frac{1}{N}\sum_{k=1}^{N} Y_k Y'_k - \left(\frac{1}{N}\sum_{k=1}^{N} Y_k\right)\left(\frac{1}{N}\sum_{k=1}^{N} Y'_k\right)}{\frac{1}{N}\sum_{k=1}^{N} Y_k^2 - \left(\frac{1}{N}\sum_{k=1}^{N} Y_k\right)^2},$$

$$\widehat{T}_{N,i} = \frac{\frac{1}{N}\sum_{k=1}^{N} Y_k Y'_k - \widehat{E}^2}{\widehat{E}_2 - \widehat{E}^2},$$

où

$$\widehat{E} = \frac{1}{N}\sum_{k=1}^{N} \frac{Y_k + Y'_k}{2} \text{ et } \widehat{E}_2 = \frac{1}{N}\sum_{k=1}^{N} \frac{Y_k^2 + Y_k'^2}{2}.$$

Notre premier objectif est d'établir, pour ces estimateurs, les résultats de normalité asymptotique que nous avons évoqué en Section 1.2.4. Ainsi, nous voulons démontrer que :

$$\sqrt{N}(\widehat{S}_{N,i} - S_i) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}(0, \sigma_S^2)$$

et :

$$\sqrt{N}(\widehat{T}_{N,i} - S_i) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}(0, \sigma_T^2),$$

pour σ_S^2 et σ_T^2 deux réels positifs.

Ces résultats peuvent être utilisés afin de fournir des intervalles de confiance justifiés asymptotiquement pour S_i, ce qui constitue un aspect de notre réponse à la question 1 (estimation de l'erreur Monte-Carlo) de notre introduction. L'outil principal utilisé pour établir ce résultat sera la δ-méthode que nous rappelons en section 1.5.2.

Nous souhaitons également comparer, en termes de variance asymptotique, ces deux estimateurs ; un estimateur ayant une variance asymptotique inférieure est à préférer, puisque celui-ci donnera des intervalles de confiance asymptotiques moins larges. Nous nous posons également la question de la variance asymptotique *minimale* que peut posséder un estimateur « raisonnable » construit sur un plan d'échantillonnage Monte-Carlo pick-freeze, et, si possible, identifier un estimateur ayant cette variance minimale. La formalisation de cette notion d'estimateur « raisonnable » et l'explicitation d'un estimateur de variance minimale constitue la réponse à la question 2 (identification d'un estimateur Monte-Carlo optimal) de l'introduction. L'outil

utilisé est ici la théorie de l'efficacité asymptotique, que nous présentons section 1.5.3.

Enfin, notre troisième objectif est d'étudier les propriétés asymptotiques (normalité et efficacité) des estimateurs $\widetilde{S}_{N,i}$ et $\widetilde{T}_{N,i}$ donnés par :

$$\widetilde{S}_{N,i} = \frac{\frac{1}{N}\sum_{k=1}^{N} \widetilde{Y}_k \widetilde{Y}'_k - \left(\frac{1}{N}\sum_{k=1}^{N} \widetilde{Y}_k\right)\left(\frac{1}{N}\sum_{k=1}^{N} \widetilde{Y}'_k\right)}{\frac{1}{N}\sum_{k=1}^{N} \widetilde{Y}_k^2 - \left(\frac{1}{N}\sum_{k=1}^{N} \widetilde{Y}_k\right)^2},$$

$$\widetilde{T}_{N,i} = \frac{\frac{1}{N}\sum_{k=1}^{N} \widetilde{Y}_k \widetilde{Y}'_k - \widetilde{E}^2}{\widetilde{E}_2 - \widetilde{E}^2},$$

où

$$\widetilde{E} = \frac{1}{N}\sum_{k=1}^{N} \frac{\widetilde{Y}_k + \widetilde{Y}'_k}{2} \text{ et } \widetilde{E}_2 = \frac{1}{N}\sum_{k=1}^{N} \frac{\widetilde{Y}_k^2 + \widetilde{Y}_k'^2}{2}.$$

et où $\{\widetilde{Y}_k\}_{k=1,\dots,N}$ et $\{\widetilde{Y}'_k\}_{k=1,\dots,N}$ sont des échantillons iid. de, respectivement :

$$\widetilde{Y} = \widetilde{f}(X_1, \dots, X_p) \;\; \widetilde{Y}' = \widetilde{f}(X'_1, \dots, X'_{i-1}, X_i, X'_{i+1}, \dots, X'_p).$$

Les estimateurs $\widetilde{S}_{N,i}$ et $\widetilde{T}_{N,i}$ sont donc simplement les estimateurs $\widehat{S}_{N,i}$ et $\widehat{T}_{N,i}$ utilisant les évaluations du métamodèle $\widetilde{f}$ comme données.

Il est aisé de vérifier que la normalité asymptotique de $\widetilde{S}_{N,i}$ et $\widetilde{T}_{N,i}$ relativement à S_i ne peut pas avoir lieu si $\widetilde{f} \neq f$. En fait, dans ce cas, on a même que ces deux estimateurs ne sont pas consistants pour estimer S_i (Proposition 9 du chapitre 3).

La solution que nous proposons est de faire dépendre le métamodèle utilisé de la taille d'échantillon N :

$$\widetilde{f} = \widetilde{f}_N. \tag{1.5.1}$$

Intuitivement, tout résultat de consistance ou de normalité asymptotique de $\widetilde{S}_{N,i}$ et $\widetilde{T}_{N,i}$ repose sur la convergence, en un certain sens restant à préciser, du métamodèle $\widetilde{f}_N$ vers le vrai modèle f. Notre but est de préciser en quel sens, et à quelle vitesse, doit avoir lieu cette convergence.

Ceci constitue une réponse à la question 3 de l'introduction, qui portait sur l'estimation de l'impact de l'erreur métamodèle sur l'estimation des indices. Cette réponse est complémentaire de celle apportée au chapitre 2, et en diffère sur les points suivants :

– nous n'utilisons pas d'évaluations de la borne d'erreur ponctuelle ε ;

– l'erreur métamodèle sur l'estimation des indices de Sobol n'est pas quantifiée numériquement, mais des conditions sont données sur la précision du métamodèle pour que l'erreur métamodèle puisse être considérée comme négligeable, asymptotiquement.

Le résultat que nous utiliserons dans cette étude est le théorème central limite triangulaire (aussi appelé théorème de Lindeberg-Feller), que nous rappelons en section 1.5.2.

1.5.2 δ-méthode et TCL triangulaire

δ-méthode

Le théorème suivant, énoncé dans [109, 3.1], porte le nom de théorème de la δ-méthode.

Theorem 1. *Soit $(U_N)_N$ une suite d'estimant de manière consistante un vecteur $U \in \mathbb{R}^k$, où $k \in \mathbb{N}^*$.*
On suppose que $(U_N)_N$ est asymptotiquement normale, c'est-à-dire que :

$$\sqrt{N}(U_N - U) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}(0, \Sigma) \tag{1.5.2}$$

où Σ est une matrice symétrique positive d'ordre k.
Soit $\Phi : \mathbb{R}^k \to \mathbb{R}$ une fonction de classe $\mathcal{C}^1$ sur un ouvert de $\mathbb{R}^k$ contenant $\{U\} \cup \{U_N\}_{N \geq N_0}$, pour $N_0 \in \mathbb{N}^*$.
Alors $(\Phi(U_N))_{N \geq N_0}$ estime $\Phi(U)$ de manière consistante et est asymptotiquement normale :

$$\sqrt{N}\left(\Phi(U_N) - \Phi(U)\right) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}\left(0, \nabla\Phi(U)^t \Sigma \nabla\Phi(U)\right),$$

où $\nabla\Phi(U)$ désigne le vecteur gradient de Φ, évalué en U.

Ce résultat permet de déduire la normalité asymptotique de $(\Phi(U_N))_N$ à partir de la normalité asymptotique de (U_N), et de calculer la variance limite connaissant celle de (U_N) et le vecteur gradient de Φ. Nous en faisons usage pour démontrer les propositions 6 et 11.
Dans nos cas d'utilisation de ce théorème, la suite (U_N) est formée de moyennes empiriques, et la normalité asymptotique de (U_N) est obtenue par application du théorème central limite vectoriel ([109, 2.18]), ou (pour la proposition 11) du théorème central limite triangulaire, que nous rappelons ci-après.

Théorème central limite triangulaire

Énonçons le théorème ainsi :

Theorem 2. *Soit, pour tout $n \in \mathbb{N}^*$, une collection $Y_{n,1}, \dots, Y_{n,n}$ de vecteurs aléatoires de $\mathbb{R}^k$ iid. et de variance finie, telle que :*

$$\forall \varepsilon > 0,\ \sum_{i=1}^{n} \mathbb{E}\left(\|Y_{n,i}\|^2 \mathbf{1}_{\|Y_{n,i}\| \geq \varepsilon} \right) \to 0 \text{ quand } n \to +\infty, \tag{1.5.3}$$

où $\|\cdot\|$ désigne la norme euclidienne sur $\mathbb{R}^k$.
Supposons également qu'il existe une matrice carrée Σ d'ordre k telle que :

$$\sum_{i=1}^{n} \mathbb{E}\left((Y_{ni} - \mathbb{E}(Y_{ni}))(Y_{ni} - \mathbb{E}(Y_{ni}))^t \right) \to \Sigma \text{ quand } n \to +\infty.$$

Alors :

$$\sum_{i=1}^{n} (Y_{ni} - \mathbb{E}(Y_{ni})) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}(0, \Sigma).$$

où $\mathcal{N}(0, \Sigma)$ désigne la loi normale en dimension k centrée, de matrice de covariance Σ.

Ce théorème, portant le nom de théorème central limite triangulaire (ou théorème de Lindeberg-Feller) est donné et démontré dans [109, 2.27]. Il généralise le théorème central limite « classique » dans le cas où l'on considère la suite, indexée par $N \in \mathbb{N}^*$, des moyennes de N variables iid. mais dont la loi dépend de N. Remarquons que cette généralisation requiert l'hypothèse supplémentaire d'uniforme intégrabilité (1.5.3).
L'utilisation de ce théorème est nécessaire pour démontrer la proposition 11 à cause de la dépendance en N du métamodèle, qui a été motivée en section 1.5.1.

1.5.3 Théorie de l'efficacité asymptotique

Notre référence dans cette section est l'ouvrage [109]. Le but de ce chapitre est de présenter la notion d'efficacité asymptotique utilisée au chapitre 3, et de donner les définitions et propriétés nécessaires à la compréhension des parties 3.2.2 et 3.3.3.

Introduction et notations

On considère le modèle statistique non paramétrique $\mathcal{P}$ formé des fonctions de répartition de vecteurs aléatoires de $\mathbb{R}^p$.
On observe des réalisations iid. $X_1, \ldots, X_N$ d'une loi $P \in \mathcal{P}$ (inconnue) et on cherche à estimer $\Psi(P)$ – où Ψ est une fonctionnelle statistique : $\Psi : \mathcal{P} \to \mathbb{R}^k$ – en fonction de ces réalisations. Pour cela on dispose d'une suite $(U_N)_{N\in\mathbb{N}^*}$ d'estimateurs de $\Psi(P)$. Chaque U_N est fonction des observations : $U_N = U_N(X_1, \ldots, X_N)$.
On s'intéresse au cas où (U_N) est asymptotiquement normale, c'est-à-dire qu'elle est telle que :

$$\sqrt{N}(U_N - \Psi(P)) \to \mathcal{N}(0, \Sigma) \text{ en loi quand } N \to +\infty$$

où Σ est la matrice (symétrique, positive) de variance-covariance asymptotique de (U_N).
Lorsque $k = 1$ (ie. la quantité à estimer est un scalaire), la normalité asymptotique s'exprime ainsi :

$$\sqrt{N}(U_N - \Psi(P)) \to \mathcal{N}(0, \sigma^2) \text{ en loi quand } N \to +\infty$$

où σ^2 est la variance asymptotique de (U_N). La normalité asymptotique permet d'utiliser (U_N) pour construire des intervalles de confiance asymptotiques pour $\Psi(P)$, dont la largeur est proportionnelle à σ. Une suite d'estimateurs asymptotiquement normale sera donc (asymptotiquement) d'autant plus "précise" que sa variance asymptotique sera petite. Si cette variance asymptotique est la plus faible possible, parmi toutes les variances asymptotiques des suites « régulières » d'estimateurs de $\Psi(P)$, on dira que la suite (U_N) est *efficace*.
Pour $k \neq 1$, cette notion d'efficacité se généralise en utilisant l'ordre sur les matrices de covariance donné par : $\Sigma \leq \Sigma'$ si $\Sigma' - \Sigma$ est une matrice (symétrique) positive.
Un certain nombre de questions sont à régler :

- Comment est définie la classe des suites « régulières » ?
- Comment, étant données Ψ et P, calculer la variance minimale d'une suite régulière d'estimateurs de $\Psi(P)$?
- Comment démontrer qu'une suite (U_N) donnée est asymptotiquement efficace ?

Les deux premières questions sont traitées dans la section 1.5.3. La section 1.5.3 donne des critères et des propriétés permettant de montrer l'efficacité asymptotique.

Efficacité asymptotique

Définition 1. *On dit que* $\Psi : \mathcal{P} \to \mathbb{R}^k$ *est différentiable en* P *suivant* $\mathcal{P}$ *s'il existe* $\Psi'_P : L^2(P) \to \mathbb{R}^k$ *telle que :*

$$\forall g \text{ mesurable bornée telle que } \int g \mathrm{d}P = 0, \quad \frac{\Psi(P_{t,g}) - \Psi(P)}{t} \to \Psi'_P(g)$$

quand $t \to 0$*, où* $P_{t,g} = P + tgP \in \mathcal{P}$*.*
Dans ce cas, il existe une unique fonction $\widetilde{\psi}_P \in L^2(P)$ *à valeurs dans* $\mathbb{R}^k$ *telle que :*

$$\forall g \text{ mesurable bornée telle que } \int g \mathrm{d}P = 0, \quad \Psi'_P(g) = \int \widetilde{\psi}_P g \mathrm{d}P$$

$\widetilde{\psi}_P$ *est appelée fonction d'influence efficace de* Ψ *en* P*, relativement à* $\mathcal{P}$*.*

Remarques :

1. La condition $\int g \mathrm{d}P = 0$ assure que $\mathrm{d}P_t$ est une mesure de probabilité (a une masse totale égale à 1).
2. Dans notre référence [*op.cit.*, 25.3] on considère des modèles statistiques $\mathcal{P}$ plus généraux. Une telle généralité sera d'ailleurs nécessaire au chapitre 3, car le modèle $\mathcal{P}$ utilisé sera un sous-ensemble strict du modèle non-paramétrique.

   Dans ce cas, la définition ci-dessus doit être modifiée en prenant g dans un "espace tangent" $\mathcal{P}'_P$. Dans le modèle non-paramétrique, dans lequel nous nous plaçons dans cette introduction, il est démontré [*op.cit.*, 25.16] que cet espace tangent est :

   $$\mathcal{P}'_P = \{g \in L^2(P) \text{ tq. } \int g \mathrm{d}P = 0\}.$$

   L'« espace tangent » que nous avons choisi :

   $$\widetilde{\mathcal{P}'_P} = \{g \text{ mesurable bornée tq. } \int g \mathrm{d}P = 0\}$$

   est un sous-ensemble strict de $\mathcal{P}'_P$; cependant la définition obtenue est équivalente puisque $\widetilde{\mathcal{P}'_P}$ est dense dans $\mathcal{P}'_P$ ([*op.cit.*, 25.24]).

Définition 2. *Une suite d'estimateurs* (U_N) *est dite* régulière *pour* $\Psi(P)$ *s'il existe une loi* L *telle que* $\forall g \in L^2(P)$ *telle que* $\int g\mathrm{d}P = 0$ *:*

$$\sqrt{N}(U_N - \Psi(P_{1/\sqrt{N},g})) \to L \text{ en loi, sous la loi } P_{1/\sqrt{N},g}, \text{ quand } N \to +\infty$$

où $dP_{t,g} = dP + tgdP$*. La loi* L *est appelée* loi limite *de* (U_N)*.*

Le point important dans la définition ci-dessus est que L ne dépend pas de la « direction de perturbation » g ; plus précisément, le comportement asymptotique de (U_N) à l'ordre $1/\sqrt{N}$ n'est pas modifié lorsque la loi des observations subit une perturbation en $g/\sqrt{N}$.
Le théorème [*op.cit.*, théorème de convolution 25.20] ci-dessous donne l'expression de la matrice de variance-covariance asymptotique minimale pour une suite d'estimateurs régulière :

Theorem 3. *Supposons que* Ψ *est différentiable en* P*, et que* (U_N) *est une suite d'estimateurs régulière pour* $\Psi(P)$*, ayant pour loi limite* L*, de matrice de variance-covariance* Σ*. Alors*

$$\Sigma - \int \widetilde{\psi}_P \widetilde{\psi}_P^T \mathrm{d}P$$

est une matrice positive.

Ce résultat justifie la définition suivante :

Définition 3. *On dit qu'une suite d'estimateurs* (U_N) *est* asymptotiquement efficace *pour* $\Psi(P)$ *si elle est régulière en* $\Psi(P)$*, avec pour loi limite* $\mathcal{N}(0, \int \widetilde{\psi}_P \widetilde{\psi}_P^T \mathrm{d}P)$*.*

Remarques :

1. En prenant $g = 0$, on obtient qu'une suite asymptotiquement efficace est asymptotiquement normale, avec $\int \widetilde{\psi}_P \widetilde{\psi}_P^T \mathrm{d}P$ comme matrice de variance-covariance asymptotique.
2. Le théorème LAM [25.21] donne une autre « interprétation » de l'optimalité de la variance $\int \widetilde{\psi}_P \widetilde{\psi}_P^T \mathrm{d}P$, qui se passe de l'hypothèse de régularité de (U_N).

Propriétés

Toutes les propriétés ci-dessous seront utilisées au chapitre 3 afin de démontrer l'efficacité asymptotique de l'estimateur de l'indice de Sobol considéré.

Critère d'efficacité. Ce critère est souvent utilisé en pratique pour montrer l'efficacité asymptotique ([*op.cit.*, Lemme 25.23]) :

Proposition 1. *Supposons Ψ différentiable en P, ayant comme fonction d'influence efficace $\widetilde{\psi}_P$. Une suite d'estimateurs (U_N) est asmptotiquement efficace pour $\Psi(P)$ si et seulement si :*

$$\sqrt{N}(U_N - \Psi(P)) - \frac{1}{\sqrt{N}} \sum_{i=1}^{N} \widetilde{\psi}_P(X_i) \to 0$$

en probabilité, quand $N \to +\infty$.

Efficacité de l'estimation empirique. Le résultat suivant [*op. cit.*, 25.24] nous donne un exemple important d'estimateur asymptotiquement efficace :

Proposition 2. *Soit $f \in L^2(P)$ à valeurs réelles. Les estimateurs empiriques :*

$$U_N = \frac{1}{N} \sum_{i=1}^{N} f(X_i)$$

forment une suite asymptotiquement efficace pour $\psi(P) = \int f \mathrm{d}P$.

Efficacité jointe et efficacité marginale. La proposition ci-dessous est une simplification de [*op. cit.*, 25.50]. Elle réduit l'efficacité de l'estimation d'un vecteur (efficacité jointe) à l'efficacité de l'estimation de ses composantes (efficacité marginale).

Proposition 3. *Soit $\Psi : \mathcal{P} \to \mathbb{R}^k$ différentiable en P, et (U_N) une suite d'estimateurs de $\Psi(P)$. Si, pour tout $l = 1, \ldots, k$, la suite des $l^{èmes}$ composantes $(U_{N,l})_N$ est asymptotiquement efficace pour $\Psi(P)_l$, alors (U_N) est asymptotiquement efficace pour $\Psi(P)$.*

Démonstration. Désignons par $\widetilde{\psi}_P$ la fonction d'influence efficace de $\Psi(P)$, et par $\widetilde{\psi}_{Pl}$ (pour $l = 1, \ldots, k$) celle de $\Psi(P)_l$. Il est clair que :

$$\widetilde{\psi}_P = \left(\widetilde{\psi}_{P1}, \ldots, \widetilde{\psi}_{Pk}\right)^T$$

l'efficacité asymptotique de (U_N) provient alors du critère de la Proposition 1 et du fait que la convergence en probabilité des marginales entraîne la convergence jointe [*op. cit.*, 2.7, (vi)]. □

δ-méthode et efficacité. Le résultat suivant [*op. cit.*, Partie 25.7, pp. 386–387] est à l'efficacité asymptotique ce que la δ-méthode est à la normalité asymptotique :

Proposition 4. *Soit $\Psi : \mathcal{P} \to \mathbb{R}^k$ différentiable en P, $\Phi : \mathbb{R}^k \to \mathbb{R}^n$ différentiable en $\Psi(P)$, (U_N) asymptotiquement efficace pour $\Psi(P)$. Alors $(\Phi(U_N))$ est asymptotiquement efficace pour $\Phi(\Psi(P))$.*

1.5.4 Résultats obtenus

Estimateurs $\widehat{S}_{N,i}$ et $\widehat{T}_{N,i}$. Nous obtenons (proposition 6 du chapitre 3) les résultats de normalité asymptotique annoncés en introduction, avec :

$$\sigma_S^2 = \frac{\mathrm{Var}\left((Y - \mathbb{E}(Y))\left[(Y' - \mathbb{E}(Y)) - S_i(Y - \mathbb{E}(Y))\right]\right)}{(\mathrm{Var}Y)^2},$$

$$\sigma_T^2 = \frac{\mathrm{Var}\left((Y - \mathbb{E}(Y))(Y' - \mathbb{E}(Y)) - S_i/2\left((Y - \mathbb{E}(Y))^2 + (Y' - \mathbb{E}(Y))^2\right)\right)}{(\mathrm{Var}Y)^2}.$$

Nous montrons, en figures 3.1 et 3.2, sur un exemple où les vrais indices sont connus analytiquement, que les intervalles de confiance asymptotiques obtenus sont consistants (c'est-à-dire qu'ils possèdent le niveau de risque désiré).
Au sujet de la comparaison des deux estimateurs considérés, nous démontrons que $\sigma_T^2 \leq \sigma_S^2$ avec égalité si et seulement si $S_i \in \{0; 1\}$ (Proposition 7). Cette propriété est confirmée numériquement (Figure 3.3).
Cette comparaison est étendue en identifiant $\widehat{T}_{N,i}$ comme estimateur asymptotiquement efficace (Proposition 8). Cela signifie que cet estimateur possède la variance asymptotique minimale parmi les estimateurs réguliers sur plan Monte-Carlo pick-freeze.

Estimateurs $\widetilde{S}_{N,i}$ et $\widetilde{T}_{N,i}$. Nous démontrons que, sous des hypothèses techniques de domination (très souvent vérifiées dans les cas pratiques), les estimateurs $\widetilde{S}_{N,i}$ et $\widetilde{T}_{N,i}$ sont asymptotiquement normaux (avec variances asymptotiques respectives σ_S^2 et σ_T^2, identiques à celles de $\widehat{S}_{N,i}$ et $\widehat{T}_{N,i}$) si et seulement si $\mathrm{Var}(f - \widetilde{f}_N) = o(1/N)$. Sur de nombreux exemples (voir les résultats numériques des sections 3.4.2, 3.4.3, 3.4.4 et 3.4.5), il semble que cette condition soit également nécessaire.
Sous d'autres hypothèses techniques, ainsi que sous l'hypothèse $\mathbb{E}(f - \widetilde{f}_N) = o(1/\sqrt{N})$, nous démontrons également que $\widetilde{T}_{i,N}$ est asymptotiquement efficace pour S_i.

1.6 Introduction au Chapitre 4 Solutions réduites pour l'équation de Burgers

Cette section présente le chapitre 4. L'objectif de ce chapitre est de présenter un métamodèle certifié pour l'équation de Burgers, contribuant ainsi à notre réponse à la question 4 (développement de métamodèles certifiés en géophysique). Ce chapitre est une publication acceptée [Janon3]. Nous commençons par décrire la problématique du chapitre, puis nous présentons les approches existantes dans la littérature avant de finir sur une présentation de notre approche et des résultats que nous avons obtenus.

1.6.1 Problématique

Le but du chapitre 4 est d'appliquer la méthode de la base réduite sur l'équation de Burgers avec viscosité : on cherche $u = u(t,x)$ définie sur $[0;T] \times [0;1]$ telle que

$$\frac{\partial u}{\partial t} + \frac{1}{2}\frac{\partial}{\partial x}(u^2) - \nu \frac{\partial^2 u}{\partial x^2} = f,$$

où $\nu > 0$ est la viscosité, f le terme de forçage, et où on a la condition initiale :

$$u(t=0,x) = u_0(x) \quad \forall x \in [0;1]$$

et la condition aux limites (ou condition de bord) de Dirichlet :

$$\begin{cases} u(t,x=0) = b_0(t) \\ u(t,x=1) = b_1(t) \end{cases} \qquad \forall t \in [0;T].$$

Si cette équation, en elle-même, ne modélise pas le comportement d'un fluide géophysique, elle présente, par rapport à la méthode base réduite « simple » présentée dans la section 1.3.3, deux spécificités :

- la non-stationarité (c'est-à-dire la mise en jeu de la variable temporelle t),
- et la non-linéarité, de type quadratique (présence de la dérivée du carré de l'inconnue),

qui se retrouvent dans le modèle utilisé en océanographie des équations de Saint-Venant (Shallow-Water) que nous souhaitons étudier par la suite. La seule différence entre les deux est le nombre de fonctions inconnues (trois dans le cas de Saint-Venant, une seule pour Burgers), qui se traduit seulement par une augmentation de la dimension des problèmes considérés, ce qui

n'est pas de nature à créer des difficultés supplémentaires, au moins dans le cadre de l'analyse théorique.

Ce chapitre s'inscrit donc dans le contexte de la question 4 de la section 1.1 (développement de métamodèles certifiés pour les modèles géophysiques).

Nous souhaitons pouvoir considérer comme paramètres d'entrée la viscosité ν, le terme de forçage f, la condition initiale u_0 et les conditions aux limites b_0 et b_1.

Pour ces trois derniers paramètres, de nature fonctionnelle, il y a nécessité, pour se ramener à un espace d'entrées de dimension finie, d'utiliser une paramétrisation, telle qu'une troncature en série de Fourier, où les paramètres d'entrée sont les coefficients de Fourier, et dont les fréquences respectives sont fixées. L'étude de la sensibilité par rapport à ces paramètres fait sens physiquement : on étudie le comportement de la solution face à des modifications des données fonctionnelles à des échelles spatiales ou temporelles différentes.

Mathématiquement on va donc considérer des paramétrisations de la forme :

$$b_0(t) = b_{0m} + \sum_{l=1}^{n(b_0)} A_l^{b_0} \Phi_l^{b_0}(t) \qquad b_1(t) = b_{1m} + \sum_{l=1}^{n(b_1)} A_l^{b_1} \Phi_l^{b_1}(t) \tag{1.6.1}$$

$$f(t,x) = f_m + \sum_{l=1}^{n_T(f)} \sum_{p=1}^{n_S(f)} A_{lp}^f \Phi_l^{fT}(t) \Phi_p^{fS}(x) \quad u_0(x) = u_{0m} + \sum_{l=1}^{n(u_0)} A_l^{u_0} \Phi_l^{u_0}(x) \tag{1.6.2}$$

où le vecteur des paramètres

$$\mu = (\nu, b_{0m}, A_1^{b_0}, \ldots, A_{n(b_0)}^{b_0}, b_{1m}, A_1^{b_1}, \ldots, A_{n(b_1)}^{b_0}, f_m, A_{11}^f, A_{12}^f, \ldots, A_{1,n_S(f)}^f, A_{2,1}^f, \ldots, A_{2,n_S(f)}^f, \ldots, A_{n_T(f),n_S(f)}^f, u_{0m}, A_1^{u_0}, \ldots, A_{n(u_0)}^{u_0})$$

satisfait une condition technique de compatibilité (4.1.4) destinée à assurer la régularité de la solution u.

Les fonctions $\Phi^{b_0}, \Phi^{b_1}, \Phi^{fS}, \Phi^{fT}$ et Φ^{u_0} sont des fonctions, de l'espace ou du temps, quelconques suffisamment régulières (des sinus, ou des cosinus dans le cas de la paramétrisation par des séries de Fourier tronquées).

Signalons à ce stade que, conformément à l'usage dans les publications traitant de la méthode base réduite, le vecteur des paramètres d'entrée est noté $\mu = (\mu_1, \ldots, \mu_p)$ dans les chapitres 4 et 5, et non $X = (X_1, \ldots, X_p)$, comme dans le reste du manuscrit.

L'un des intérêts de la méthode base réduite est la disponibilité d'une borne d'erreur rigoureuse calculable explicitement via une procédure offline/online efficace. Nous souhaitons également disposer d'une borne d'erreur dans le contexte présenté ci-dessus.

1.6.2 Approches existantes

Nous présentons maintenant les deux approches existant dans la littérature.

Paramétrisation de la viscosité

L'approche décrite ici est celle de [71]. Dans cette approche, la condition initiale et la condition aux limites sont fixées (à zéro), et le terme de forçage est fixé (à 1). Le seul paramètre est donc la viscosité ν. Nous résumons rapidement les caractéristiques de cette approche ; en effet, ces dernières sont reprises dans notre approche, les détails peuvent donc être trouvés dans la section suivante ou dans le chapitre 4.

Gestion de la dépendance en temps et de la non-linéarité. L'équation de Burgers est d'abord discrétisée en temps suivant un schéma implicite, c'est-à-dire que l'on se donne un pas de temps $\Delta t > 0$ et que l'on cherche $u_1, \ldots, u_{T/\Delta t}$ satisfaisant la relation de récurrence :

$$u_{t+1} - \Delta t F(t, u_{t+1}) = u_t, \quad \forall t,$$

où $F(t, \cdot)$ est un opérateur différentiel en espace.
Chacune de ces relations est une EDP (non-linéaire) faisant intervenir seulement les variables d'espace. L'étape suivante est de discrétiser en espace, par une méthode d'éléments finis $\mathcal{P}^1$, ce qui transforme l'EDP non-linéaire en système d'équations algébriques non-linéaires.
La résolution de ce système non-linéaire est alors effectuée à l'aide de la méthode de Newton, qui ramène le problème non-linéaire à une succession de problèmes linéaires. Chacun de ces problèmes linéaires peut être alors traité par la méthode base réduite, ce qui donne une procédure offline/online de calcul de $\widetilde{u}$.

Choix de base. Les auteurs proposent d'utiliser une variante de la méthode greedy, adaptée afin de tenir compte de la dépendance temporelle.

Borne d'erreur. Pour chaque pas de temps t_k, une borne d'erreur sur $\|u(t_k) - \widetilde{u}(t_k)\|$ est établie, où $\|\cdot\|$ est la norme $L^2(0,1)$. Les auteurs déplorent la croissance exponentielle de cette borne d'erreur en fonction du temps et constatent que cette croissance est encore plus importante dans le cas des faibles viscosités.

Paramétrisation complète

L'autre référence existante est [59]. L'approche est très similaire à celle décrite plus haut, à ceci près qu'elle permet la paramétrisation de la condition initiale, de la condition aux limites de Dirichlet et du forçage sous la forme (1.6.1).

La paramétrisation de la condition de Dirichlet se fait en se ramenant d'abord, par translation, au cas où cette condition est nulle.

Signalons que les résultats numériques ayant trait à la borne d'erreur ne sont pas présentés dans cette publication (c'est la vraie erreur qui est représentée, et le calcul de cette vraie erreur nécessite de calculer la « vraie » solution numérique, et c'est précisément ce que l'on souhaite éviter en utilisant une réduction de modèle). Par ailleurs, les tests numériques sont faits avec des conditions aux limites très proches (numériquement indiscernables) d'une condition nulle.

1.6.3 Notre approche

Phase offline/online. Nous choisissons, c'est là une des différences entre notre approche et celles présentées plus haut, d'intégrer la condition aux limites de Dirichlet sous *forme faible*, c'est-à-dire que nous modifions la forme faible de l'équation de Burgers (4.1.6) en une forme faible *pénalisée* (4.1.9) imposant une valeur au bord proche de celle souhaitée. L'intérêt de cette technique, comparée à l'approche décrite ci-dessus, est qu'elle introduit moins de termes dans la décomposition affine des problèmes variationnels (le nombre Q dans 1.3.3), ce qui diminue les temps de calcul offline et online, ainsi que le besoin en espace de stockage des éléments calculés lors de la phase offline.

Le reste de la réduction est similaire aux approches existantes : la forme faible pénalisée (4.1.9) est discrétisée en espace pour obtenir (4.1.11), puis en temps (section 4.1.2). Enfin, la méthode de Newton est appliquée pour

obtenir l'équation linéarisée (4.1.14), qui est ensuite réduite dans la section 4.2, en suivant le principe donné en section 1.3.3.

Choix de base. Nous considérons trois méthodes de choix de base réduite :

1. le choix basé sur la POD, qui est une adaptation de la méthode POD présentée en section 1.3.3 ; ce choix est décrit en section 4.2.3 ;
2. le choix basé sur un algorithme glouton (greedy), qui est utilisé dans [71] ; l'algorithme est rebaptisé « local greedy » et est détaillé section 4.2.3 ;
3. le choix hybride « POD-Greedy », qui prend sa source dans [46] et que nous décrivons section 4.2.3.

Borne d'erreur. Un aspect important de cette contribution est la borne d'erreur *a posteriori*, qui est obtenue par une méthode différente de celle existant dans la littérature, et qui permet d'effectuer une majoration plus fine. Cette borne d'erreur, ainsi que sa méthode de calcul, est décrite en détail dans la section 4.3.

Résultats obtenus. Les résultats numériques, présentés dans la section 4.4, montrent notamment que la réduction de l'équation de Burgers permet un gain en temps de calcul significatif, tel qu'un gain de 85% en temps de calcul, au prix d'une erreur relative certifiée de *un pour mille* (section 4.4.2). Par ailleurs, notre borne d'erreur est comparée avec la borne existante (section 4.4.5), et la comparaison est clairement à l'avantage de notre approche. Enfin, les trois procédures de choix de base sont comparées en termes de borne d'erreur relative moyenne et de borne d'erreur maximale (Figure 4.7). Nous voyons que le choix POD donne le meilleur résultat en termes de borne moyenne, alors que le greedy local permet de minimiser la borne maximale. Ce résultat, peu surprenant au regard de la construction de ces méthodes, est confirmé par la Figure 4.6 (même si pour $n = 10$ la méthode POD semble en-dessous de la méthode greedy, il faut remarquer qu'à ce stade, l'erreur relative, de l'ordre de 10^{-6} est inférieure à la précision numérique).

1.7 Introduction au Chapitre 5 Estimation d'erreur sortie-dépendante pour la méthode base réduite

Le chapitre 5 fait l'objet d'une publication soumise [Janon4]. Il a pour but d'apporter une réponse à la question 5, à savoir le développement d'une borne d'erreur métamodèle dépendant de la sortie (quantité d'intérêt) considérée.
Cette section d'introduction au chapitre 5 procède dans le même ordre que la section précédente : description de la problématique, présentation des approches existantes, résumé de notre approche et résultats obtenus.

1.7.1 Problématique

Avant de décrire la problématique de ce chapitre, commençons par introduire son contexte et ses notations.
Dans le chapitre 5, nous considérons un espace de dimension finie X, nous notons μ le paramètre d'entrée et nous considérons la solution $u(\mu) \in X$ du système d'équations linéaires :

$$A(\mu)u(\mu) = f(\mu),$$

pour une matrice $A(\mu)$ et un vecteur $f(\mu)$ pouvant s'écrire :

$$A(\mu) = \sum_{q=1}^{Q} \Theta_q(\mu) A_q, \quad f(\mu) = \sum_{q'=1}^{Q'} \gamma_{q'}(\mu) f_{q'}.$$

Typiquement, ce système d'équations est obtenu en discrétisant la forme faible d'une équation aux dérivées partielles, dans l'espace de grande dimension X. L'application de la méthode base réduite (voir 1.3.3) à cette équation mène au système d'équations réduit :

$$\widetilde{A}(\mu)\widetilde{u}(\mu) = \widetilde{f}(\mu),$$

d'inconnue $\widetilde{u}(\mu)$ appartenant à un espace réduit $\widetilde{X} \subset X$ fixé ($\widetilde{X}$ est l'espace engendré par la base réduite, choisie par exemple par l'une des méthodes présentée en section 1.3.3).

Remarquons que, comme dans le chapitre précédent, le vecteur d'entrée est noté μ et non X, et que la sortie du modèle n'est pas notée f mais s :

$$s(\mu) = \sigma(u(\mu)).$$

Cette sortie est supposée linéaire en $u(\mu)$. Autrement dit, σ est une forme linéaire de X.
On définit $\widetilde{s}$ par :

$$\widetilde{s}(\mu) = \sigma(\widetilde{u}(\mu)).$$

Une fois qu'ont été calculés, grâce à une procédure offline/online dont la procédure online est de complexité indépendante de $\dim X$, les coefficients de $\widetilde{u}(\mu)$ dans la base réduite engendrant $\widetilde{X}$, le scalaire $\widetilde{s}(\mu)$ est rapide à calculer. Il est donc naturel de le considérer comme « sortie réduite » du modèle, et d'utiliser $\widetilde{s}$ comme métamodèle en lieu et place de s.
La problématique principale du chapitre est celle de la question 5 de la section 1.1, à savoir de donner une borne explicitement calculable, via une procédure offline/online avec phase online de complexité indépendante de $\dim X$, pour l'erreur de métamodèle $|s(\mu) - \widetilde{s}(\mu)|$. Dans notre contexte, une telle borne d'erreur sortie-dépendante est motivée par le fait qu'en analyse de sensibilité, on considère le modèle uniquement au travers de cette quantité d'intérêt.
Nous souhaitons également pouvoir utiliser cette borne pour certifier le calcul des indices de Sobol à partir du métamodèle, rejoignant en cela notre question 3.

1.7.2 Approches existantes

Approche « borne Lipschitz »

La première possibilité est d'utiliser la norme duale de s :

$$L = \sup_{v \in X, v \neq 0} \frac{s(v)}{\|v\|}$$

et d'utiliser la linéarité de s pour écrire la borne suivante :

$$|s(\mu) - \widetilde{s}(\mu)| \leq L \, \|u(\mu) - \widetilde{u}(\mu)\| \leq L \varepsilon_u(\mu),$$

où $\varepsilon_u(\mu)$ désigne la borne supérieure sur $\|u(\mu) - \widetilde{u}(\mu)\|$ donnée par la méthode base réduite (voir section 1.3.3).

Nous avons donné le nom de « borne Lipschitz » à cette borne, puisque L coïncide avec la constante de Lipschitz de s dans le cas où s est linéaire. Dans le cas où s n'est pas linéaire, cette borne est toujours valable si s est L-lipschitzienne.

Approche basée sur l'adjoint

Dans cette section nous supposons, pour simplifier, que la matrice $A(\mu)$ est symétrique pour tout μ. Nous supposons également que $A(\mu)$ est définie positive.
L'autre approche disponible dans la littérature, décrite dans [72] et prenant sa source dans [76], est de considérer la solution $u^a(\mu)$ du problème adjoint :

$$A(\mu)u^a(\mu) = -l, \tag{1.7.1}$$

où l est le vecteur de X représentant la forme linéaire s :

$$s = \langle l, \cdot \rangle .$$

Le problème 1.7.1 peut se voir appliquer la méthode base réduite, ce qui permet de calculer une solution réduite adjointe $\widetilde{u}^a(\mu)$.
L'approche basée sur l'adjoint consiste à considérer comme sortie réduite la sortie réduite « corrigée » :

$$\widetilde{s}^a(\mu) = \sigma(\widetilde{u}(\mu)) + \langle A(\mu)\widetilde{u}(\mu) - f(\mu), \widetilde{u}^a(\mu) \rangle .$$

Cette sortie corrigée admet la borne d'erreur suivante :

$$|s(\mu) - \widetilde{s}^a(\mu)| \leq \frac{\|r(\mu)\|_* \, \|r^a(\mu)\|_*}{\alpha(\mu)}, \tag{1.7.2}$$

où $r(\mu)$ et $r^a(\mu)$ désignent les formes linéaires sur X suivantes :

$$r(\mu)(v) = v^t A(\mu)\widetilde{u}(\mu) - v^t f(\mu) \quad r^a(\mu)(v) = v^t A(\mu)\widetilde{u}^a(\mu) + v^t l,$$

où, pour toute forme linéaire ϕ sur X,

$$\|\phi\|_* = \sup_{v \in X, v \neq 0} \frac{\phi(v)}{\|v\|},$$

et où $\alpha(\mu)$ est la constante de coercivité de $A(\mu)$:

$$\alpha(\mu) = \inf_{v,w \in X, v \neq 0, w \neq 0} \frac{v^t A(\mu) w}{\|v\| \, \|w\|}.$$

Remarquons que $\alpha(\mu) > 0$ par définie-positivité de $A(\mu)$.
En pratique, la borne (1.7.2) est calculable, grâce à une procédure offline/online de calcul des normes des résidus $\|r(\mu)\|$ et $\|r^a(\mu)\|$ (voir [Janon5] §3.2 et [72]), et d'une borne inférieure sur $\alpha(\mu)$ (voir [Janon5] §3.3 et [72]).

1.7.3 Notre approche

Borne d'erreur

Nous pouvons faire les remarques suivantes sur les deux approches proposées plus haut :

– d'une part, la borne Lipschitz est une borne optimale, parmi les bornes qui ne dépendent que de la borne ε_u sur $\|u - \widetilde{u}\|$;
– d'autre part, la borne basée sur l'adjoint donne de meilleurs résultats que la borne Lipschitz mais elle nécessite l'application de la méthode base réduite sur le problème adjoint, ce qui double le temps de calcul nécessaire aux phases offline et online.

La borne d'erreur que nous proposons au chapitre 5 est intermédiaire entre ces deux approches, c'est-à-dire qu'elle fait entrer en jeu d'autres quantités que ε_u, afin d'obtenir une borne d'erreur plus performante que la borne Lipschitz, tout en ne nécessitant pas la considération du problème adjoint, afin de ne pas augmenter le temps de calcul outre mesure.

L'idée principale est de remarquer que les deux bornes précédentes se basent toutes sur un calcul offline/online de la norme du résidu $\|r(\mu)\|_*$, alors qu'il est également possible de donner une procédure offline/online efficace pour calculer le produit scalaire de $r(\mu)$ avec un vecteur déterminé durant la phase offline.

Nous proposons donc de choisir une base orthonormée $\Phi = \{\phi_1, \ldots, \phi_{\dim X}\}$ de X, de tronquer cette base en ne gardant que $N < \dim X$ vecteurs, et de borner $|s - \widetilde{s}|$ en fonction des N produits scalaires :

$$\langle r(\mu), \phi_1 \rangle, \ldots, \langle r(\mu), \phi_N \rangle .$$

Plus spécifiquement, nous obtenons une borne en probabilité (relativement à μ), c'est-à-dire que nous explicitons une quantité $\varepsilon(\mu, \alpha, N, \Phi)$ telle que :

$$P\left(|s - \widetilde{s}| \geq \varepsilon(\mu, \alpha)\right) \leq \alpha, \tag{1.7.3}$$

pour un niveau de risque $\alpha \in]0; 1[$ fixé.

L'expression de $\varepsilon(\mu, \alpha, N, \Phi)$ est la suivante :

$$\varepsilon(\mu, \alpha, N, \Phi) = T_1(\mu, N, \Phi) + \frac{T_2(N, \Phi)}{\alpha},$$

où $T_1(\mu, N, \Phi)$ et $T_2(N, \Phi)$ sont définies section 5.1.2. La justification de (1.7.3) fait l'objet du théorème 5 du chapitre 5.

Dans la suite du chapitre, nous donnons une heuristique pour le choix de Φ, basée sur le théorème 6 chapitre 5, et, dans la section 5.1.3, nous décrivons la procédure offline/online de calcul et d'estimation de T_1 et T_2.

Application à l'analyse de sensibilité

La borne d'erreur que nous obtenons est de nature probabiliste. La méthode développée au chapitre 2 nécessitant une borne déterministe, il convient de l'adapter afin de tenir compte du risque probabiliste de la borne métamodèle. C'est ce qui est fait dans la section 5.2, où, dans le théorème 7, nous corrigeons le risque de l'intervalle de confiance combiné décrit au chapitre 2 en fonction du risque de la borne d'erreur.

Résultats obtenus

L'évaluation de notre méthode s'est faite sur deux cas test (sections 5.3 et 5.4).
Dans les deux cas test, nous constatons (figures 5.1 et 5.2) que la borne que nous proposons donne de meilleurs résultats (en termes de borne d'erreur moyennée sur un échantillon de paramètres de tests) que la borne Lipschitz et que la borne basée sur le dual. Cette dernière nécessitant plus de calculs que les autres bornes, nous avons dû corriger la taille de base réduite utilisée (en abscisse sur les figures) afin de tenir compte de cette différence de complexité entre les méthodes. Par ailleurs, nous prenons des niveaux de risques petits ($\alpha = 10^{-2}$ ou $\alpha = 10^{-4}$) afin de ne pas trop biaiser la comparaison de notre borne avec risque α avec les autres, qui sont déterministes.
Enfin, nous montrons (table 5.1) que, comme le niveau de risque peut être choisi très petit sans déteriorer significativement notre borne, la correction du niveau de l'intervalle de confiance proposée dans le théorème 7 permet d'utiliser la borne que nous proposons pour faire de l'analyse de sensibilité certifiée.

Chapter 2

Uncertainties assessment in global sensitivity indices estimation from metamodels

Résumé: L'analyse de sensibilité globale nécessite de nombreuses exécutions du modèle considéré et, de ce fait, elle est souvent impraticable pour les modèles complexes et/ou nécessitant beaucoup de ressources de calcul afin d'être évalués. L'approche métamodèle consiste à remplacer le modèle original par un modèle approchant qui est beaucoup plus rapide à exécuter. Cet article traite de la perte d'information lors de l'estimation des indices de sensibilité due à l'approximation par le métamodèle. Nous présentons une méthode permettant une analyse robuste de l'erreur, ouvrant ainsi la voie à des gains significatifs en temps de calcul, sans contrepartie en termes de précision et de rigueur. La méthodologie proposée est illustré sur deux types de métamodèles différents: les métamodèles obtenus par base réduite et ceux obtenus par interpolation à noyau (interpolation RKHS).

Abstract: Global sensitivity analysis is often impracticable for complex and resource intensive numerical models, as it requires a large number of runs. The metamodel approach replaces the original model by an approximated code that is much faster to run. This paper deals with the information loss in the estimation of sensitivity indices due to the metamodel approximation. A method for providing a robust error assessment is presented, hence enabling significant time savings without sacrificing on precision and rigor.

The methodology is illustrated on two different types of metamodels: one based on reduced basis, the other one on RKHS interpolation.

2.1 Introduction

Many mathematical models use a large number of poorly-known parameters as inputs. The impact of parameter uncertainty on the model output is important for the user of these models. This problem can be tackled by considering the uncertain input parameters as random variables, whose probability distribution reflects the practitioner's belief in the precision of the parameter values. Model output, as function of the model inputs, is then a random variable. Its probability distribution, uniquely determined by the model and the distribution of the inputs, can give detailed and valuable information about the behavior of the output when input parameters vary: range of attained values, mean value and dispersion about the mean, most probable values (modes), *etc.*

Sensitivity analysis aims to identify the sensitive parameters, that is, parameters for which a small variation implies a large variation of the model output. In global sensitivity analysis, one makes use of the probability distribution of the outputs to define (amongst other sensitivity measures) *sensitivity indices* (also known as *Sobol indices*). The sensitivity index of an output with respect to an input variable is the fraction of output variance which can be "explained" by the variation of the input variable, either alone (one then speaks about main effect), or in conjunction with other input variables (total effect). This way, input variables can be sorted by the order of importance they have on the output. One can also consider the proportion of variance due to the variation of groups of two or more inputs, although main effects and total effects are generally sufficient to produce a satisfying sensitivity analysis; see, e.g., [48, 88] for more information about uncertainty and sensitivity analysis.

Once these indices have been defined, the question of their effective calculation remains open. For most models, an exact, analytic computation is not attainable (even expressing an output as an analytic function of the inputs is infeasible) and one has to use numerical approximations.

A robust, popular way to obtain such approximations is Monte Carlo estimation. This method simulates randomness in inputs by sampling a large number of parameter values (from the selected input distribution). The model output is then computed for each sampled value of the parameters. This way, one obtains a sample of outputs, under the conjugate action of the

model and the input distribution. A suitable statistical estimator can then be applied to form a numerical estimate of the sensitivity index based on the sample of outputs. The Monte Carlo approach to Sobol indices computation is described in [98], together with improvements in [49, 86].

A major drawback of the Monte Carlo estimation is that a large number of model outputs have to be evaluated for the resulting approximation of the sensitivity index to be accurate enough to be useful. In complex models, in which a simulation for one single value of the parameters may take several minutes, using these methods "as-is" is impracticable. In those cases, one generally makes use of a *surrogate model* (also known as *reduced model*, *emulator*, *metamodel* or *response surface*). The surrogate model has to approximate the original model (called the *full* model) well, while being many times faster to evaluate. The sensitivity index is then calculated *via* a sample of outputs, each generated by a call to the surrogate model, thus accelerating the overall computation. The aim of this paper is to quantify accuracy loss due to the use of a metamodel combined to a Monte-Carlo method to compute sensitivity indices.

The sensitivity index produced by Monte Carlo estimation with a surrogate model is tainted by a twofold error. Firstly, our Monte-Carlo sampling procedure assimilates the whole (generally infinite) population of possible inputs with the finite, randomly chosen, sample; this produces *sampling*, or *Monte-Carlo error*. Secondly, using a surrogate model biases the estimation of the Sobol index, as what is actually estimated is sensitivity of surrogate output, and not the full one; we call this bias the *metamodel error*.

A variation on the bootstrap, which addresses sampling error as well as metamodel error, has been proposed in [100]; in this work, the authors propose to use a bootstrap strategy on the metamodel residuals to estimate metamodel error and Monte-Carlo error simultaneously. In [66], confidence intervals for the Sobol index are estimated using the conditional distribution of the Kriging predictor given the learning sample. This approach is limited to Kriging metamodels. Finally, the paper by [11] makes use of the reduced-basis output error bound to certify computation of the expectation and the variance of a model output (and not, as in this paper, the Sobol indices) with neglected sampling error.

In this paper, we want to make a rigorous sensitivity analysis, so it is important to assess the magnitude of these two combined errors on the estimated

sensitivity indices. We will also use such assessment to help with the choice of correct approximation parameters (Monte-Carlo sample size and metamodel fidelity) to achieve a desired precision in estimated indices.

We estimate sampling error by using a classical method, which comes at a moderate numerical cost: bootstrap resampling ([33, 1]). Based on statistical estimation theory, the bootstrap technique involves the generation of a sample of sensitivity index estimator replications, whose empirical distribution serves as approximation of the true (unknown) estimator distribution, in order to produce asymptotic confidence intervals that give good results in many practical cases.

To estimate metamodel error, we will use the surrogate models coming with *error bounds*, that is, computable (or at least estimable) upper bounds on the error between the original and the surrogate outputs. The reduced basis (RB) method ([72, 45, 110, 44]) is a method leading, in some cases, to such rigorously certified metamodels; it is applicable when the original model is a discretization of a partial differential equation (PDE) depending on the input parameters. Kriging ([92]) – also known as Gaussian process metamodelling, which is equivalent to RKHS (reproducing kernel Hilbert space) interpolation ([94]) – also provides error indicators. In contrast to the RB method, Kriging/RKHS interpolation only requires "blackbox" training data, i.e. a (finite) set of input-output pairs from the original model. This makes it more versatile and easier to use, at the expense of rigor in the error bounds and quality in the approximation.

Our new approach is based on the separation of the estimation of the metamodel and the sampling error. The advantages brought by our approach are: its rigorousness (the impact of the use of a surrogate model is provably bounded); its efficiency (our bounds are rather sharp, and go to zero when metamodel errors decrease); its moderate computational requirements (time is better spent on making a precise computation than at measuring precision) and its versatility with respect to metamodel choice (the user can choose any metamodel that comes with computable error bound or error indicator). In other words, our method allows us to estimate sensitivity indices by using a metamodel which greatly speeds up computation times, while rigorously keeping track of the precision of the estimation. The model has to exhibit some regularity which can be captured by a metamodel. Moreover, the metamodel error bound should not be too pessimistic.

This paper is organized as follows: in the first part, we describe the prerequisites for our approach: we give the definition of the standard Monte Carlo estimator of the sensitivity indices of interest; in the second part, we present our confidence interval estimation technique for the sensitivity index, which accounts for the two sources of error described earlier (sampling and metamodel). In the third part, we present applications of our method for reduced-basis and RKHS/Kriging interpolation metamodels, and compare our method with the method described in [100].

2.2 Review of sensitivity indices

2.2.1 Definition

In order to define sensitivity indices, we choose a probability distribution for the input variables, considering each input variable X_i $(i = 1, \dots, p)$ as a random variable with known distribution; the model output $Y = f(X_1, \dots, X_p)$ (assumed to be square-integrable, non a.s. constant and scalar: multiple outputs can be treated separately) is thus for $\mathbf{X} = (X_1, \dots, X_p)$ a $\sigma(\mathbf{X})$-measurable random variable (assuming that f is a $\sigma(\mathbf{X})$-measurable function). We further assume that the X_i's are mutually independent. We also fix throughout all this paper an input variable of interest $1 \leq i \leq p$. We define the *first-order main effect* of X_i on Y by:

$$S_i = \frac{\mathrm{Var}\mathbb{E}(Y|X_i)}{\mathrm{Var}Y} \tag{2.2.1}$$

S_i is the sensitivity index in which we are interested in this paper but other indices (total effect, high-order effects) exist and our methodology can readily be extended to these indices.

2.2.2 Monte-Carlo estimator

We are interested in the following Monte-Carlo estimator for S_i ([49, 86]): a sample size $N \in \mathbb{N}$ being given, let $\left\{\mathbf{X}^k\right\}_{k=1,\dots,N}$ and $\left\{\mathbf{X}'^k\right\}_{k=1,\dots,N}$ be two random i.i.d. samples of size N each, drawn from the distribution of the input vector $\mathbf{X}$.

For $k = 1, \dots, N$, we note:

$$y_k = f(\mathbf{X}^k) \tag{2.2.2}$$

and:

$$y'_k = f(X_1'^k, \ldots, X_{i-1}'^k, X_i^k, X_{i+1}'^k, \ldots, X_p'^k) \tag{2.2.3}$$

The Monte-Carlo estimator of S_i is then given by:

$$\widehat{S_i} = \frac{\overline{yy'} - \overline{y} \times \overline{y'}}{\overline{y^2} - \overline{y}^2} \tag{2.2.4}$$

where, for any vector $v = (v_1, \ldots, v_N)$,

$$\overline{v} = \frac{1}{N} \sum_{k=1}^{N} v_i$$

It can be shown that $\widehat{S_i}$ is a strongly consistent estimator of S_i.
Remark: our methodology can be extended to higher order Sobol indices by choosing primed output sample $\{y'_k\}$ appropriately. More specifically, for a subset of variables $I \subset \{1, \ldots, p\}$, the closed index with respect to I, defined by:

$$S_I = \frac{\mathrm{Var}\mathbb{E}(Y|(X_i)_{i\in I})}{\mathrm{Var}Y}$$

is estimated by using:

$$y'_k = f(\mathbf{X}_I'^k), \text{ where } X_{I,i}'^k = \begin{cases} X_i^k \text{ if } i \in I \\ X_i'^k \text{ else} \end{cases}.$$

and standard high-order indices can be treated by subtracting the effects of the proper subsets of I.

2.3 Quantification of the two types of error in index estimation

We now present our method for estimating the two types of error that occur in Monte-Carlo sensitivity index estimation on a reduced-basis metamodel. In Sections 2.3.1 and 2.3.2, we show two approaches for taking metamodel error in account. In Section 2.3.3, we review the bootstrap, which we will use for the treatment of sampling error. Section 2.3.4 shows how to combine metamodel error and Monte-Carlo estimation in order to provide the final index interval estimation.

2.3.1 Metamodel error in index estimation

We now denote by $\widetilde{f} : \mathcal{P} \subseteq \mathbb{R}^p \to \mathbb{R}$ the metamodel output approximating $f : \mathcal{P} \to \mathbb{R}$, and by ε the pointwise error bound that certifies the metamodel approximation, i.e. we have:

$$\left|f(\mathbf{x}) - \widetilde{f}(\mathbf{x})\right| \leq \varepsilon(\mathbf{x}) \quad \forall \mathbf{x} \in \mathcal{P}$$

For a pair of samples $\left(\{\mathbf{X}^k\}_{k=1,\ldots,N}, \{\mathbf{X}'^k\}_{k=1,\ldots,N}\right)$ of inputs, we can use our metamodel output $\widetilde{f}$ and our metamodel error bound ε to compute, for $k = 1, \ldots, N$:

$$\widetilde{y}_k = \widetilde{f}(\mathbf{X}^k), \quad \widetilde{y}'_k = \widetilde{f}(X_1'^k, \ldots, X_{i-1}'^k, X_i^k, X_{i+1}'^k, \ldots, X_p'^k) \tag{2.3.1}$$

and:

$$\varepsilon_k = \varepsilon(\mathbf{X}^k), \quad \varepsilon'_k = \varepsilon(X_1'^k, \ldots, X_{i-1}'^k, X_i^k, X_{i+1}'^k, \ldots, X_p'^k) \tag{2.3.2}$$

In this section, we find accurate, explicitly and efficiently computable bounds $\widehat{S}_i^m$ and $\widehat{S}_i^M$, depending only on $\widetilde{y}_k, \widetilde{y}'_k, \varepsilon_k$ and ε'_k so that:

$$\widehat{S}_i^m \leq \widehat{S}_i \leq \widehat{S}_i^M \tag{2.3.3}$$

In other words, we want lower and upper bounds on the full model based sensitivity index estimator $\widehat{S}_i$ computable from surrogate model calls.
We now define, for any $\mathbf{z} = (z_1, \ldots, z_N, z'_1, \ldots, z'_N) \in \mathbb{R}^{2N}$ and any $a, t, t' \in \mathbb{R}$ the R function by:

$$R(a; \mathbf{z}, t, t') = \sum_{k=1}^{N} \left(z'_k - (a(z_k - t) + t')\right)^2.$$

Let $\mathbf{y} = (y_1, \ldots, y_N, y'_1, \ldots, y'_N)$. By setting first derivative of R with respect to a to zero, making use of the convexity of $R(\cdot; \mathbf{y}, \overline{y}, \overline{y}')$ and using the definition (2.2.4) of $\widehat{S}_i$, one easily shows that:

$$\widehat{S}_i = \operatorname*{argmin}_{a \in \mathbb{R}} R(a; \mathbf{y}, \overline{y}, \overline{y}'). \tag{2.3.4}$$

In other words, $\widehat{S}_i$ is the slope of the linear least squares regression of the $\{y'_k\}_k$ on the $\{y_k\}_k$. The key of our approach is to bound $R(a; \mathbf{y}, \overline{y}, \overline{y}')$ between two quantities which are computable without knowing $\mathbf{y}$, nor $\overline{y}$ and $\overline{y}'$ and by using (2.3.4) to deduce bounds for any realization (ie., evaluation

on a sample) of $\widehat{S}_i$ which depend only on metamodel outputs and error bounds, that is $y_k, \varepsilon_k, y'_k, \varepsilon'_k, k = 1, \ldots, N$.
Recall that:

$$\widetilde{y}_k = \widetilde{f}(\mathbf{X}^k), \;\; \widetilde{y}'_k = \widetilde{f}(X_1'^k, \ldots, X_{i-1}'^k, X_i^k, X_{i+1}'^k, \ldots, X_p'^k)$$

and:

$$\varepsilon_k = \varepsilon(\mathbf{X}^k), \;\; \varepsilon'_k = \varepsilon(X_1'^k, \ldots, X_{i-1}'^k, X_i^k, X_{i+1}'^k, \ldots, X_p'^k).$$

Define:

$$R_{inf}(a; \widetilde{\mathbf{y}}, \boldsymbol{\varepsilon}, t, t') = \sum_{k=1}^{N} \left\{ \inf_{z_k \in [\widetilde{y}_k - \varepsilon_k; \widetilde{y}_k + \varepsilon_k], z'_k \in [\widetilde{y}'_k - \varepsilon'_k; \widetilde{y}'_k + \varepsilon'_k]} \left(z'_k - (a(z_k - t) + t')\right)^2 \right\} \tag{2.3.5}$$

and:

$$R_{sup}(a; \widetilde{\mathbf{y}}, \boldsymbol{\varepsilon}, t, t') = \sum_{k=1}^{N} \left\{ \sup_{z_k \in [\widetilde{y}_k - \varepsilon_k; \widetilde{y}_k + \varepsilon_k], z'_k \in [\widetilde{y}'_k - \varepsilon'_k; \widetilde{y}'_k + \varepsilon'_k]} \left(z'_k - (a(z - t) + t')\right)^2 \right\} \tag{2.3.6}$$

where $\widetilde{\mathbf{y}} = (\widetilde{y}_1, \ldots, \widetilde{y}_N, \widetilde{y}'_1, \ldots, \widetilde{y}'_N)$, $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_N, \varepsilon'_1, \ldots, \varepsilon'_N)$.
It is clear that:

$$R_{inf}(a; \widetilde{\mathbf{y}}, \boldsymbol{\varepsilon}, t, t') \leq R(a; \mathbf{y}, t, t') \leq R_{sup}(a; \widetilde{\mathbf{y}}, \boldsymbol{\varepsilon}, t, t') \quad \forall a, t, t' \in \mathbb{R} \tag{2.3.7}$$

Note that R, R_{inf} and R_{sup} are quadratic polynomials in a. We name $\alpha, \beta, \gamma, \alpha_{inf}, \beta_{inf}, \gamma_{inf}, \alpha_{sup}, \beta_{sup}$ and γ_{sup} their respective coefficients. In other words, we have:

$$R(a; \mathbf{y}, t, t') = \alpha a^2 + \beta a + \gamma$$

$$R_{inf}(a; \widetilde{\mathbf{y}}, \boldsymbol{\varepsilon}, t, t') = \alpha_{inf} a^2 + \beta_{inf} a + \gamma_{inf} \tag{2.3.8}$$

$$R_{sup}(a; \widetilde{\mathbf{y}}, \boldsymbol{\varepsilon}, t, t') = \alpha_{sup} a^2 + \beta_{sup} a + \gamma_{sup} \tag{2.3.9}$$

These coefficients are computable by evaluating

$$R_{inf}(a; \widetilde{\mathbf{y}}, \boldsymbol{\varepsilon}, t, t') \text{ and } R_{sup}(a; \widetilde{\mathbf{y}}, \boldsymbol{\varepsilon}, t, t')$$

for three different values of a and solve for the interpolating quadratic functions. These coefficients depend on t and t', as well on $\mathbf{y}$ (for α, β, γ) and $\widetilde{\mathbf{y}}$ and $\boldsymbol{\varepsilon}$ (for the other coefficients). We do not explicitly write this dependence until the last part of our discussion.

Using (2.3.7) we see that the quadratic function of a:

$$(\alpha_{inf} - \alpha)a^2 + (\beta_{inf} - \beta)a + \gamma_{inf} - \gamma$$

is negative or zero; hence it takes a non-positive value for $a = 0$, and has a non-positive discriminant:

$$\gamma_{inf} - \gamma \leq 0 \tag{2.3.10}$$

$$(\beta_{inf} - \beta)^2 \leq 4(\alpha_{inf} - \alpha)(\gamma_{inf} - \gamma) \tag{2.3.11}$$

As $(\beta_{inf} - \beta)^2 \geq 0$, Equations (2.3.10) and (2.3.11) above imply that $\alpha_{inf} - \alpha \leq 0$, and that:

$$\beta_{inf} - \delta_{inf} \leq \beta \leq \beta_{inf} + \delta_{inf}$$

for $\delta_{inf} = 2\sqrt{(\alpha_{inf} - \alpha)(\gamma_{inf} - \gamma)}$.
We now suppose that $\alpha_{inf} > 0$. As α_{inf} is computable from $\widetilde{y}_k, \widetilde{y}'_k, \varepsilon_k$ and ε'_k, one can practically check if this condition is met. If it is not the case, our bound can not be used. We expect that if the metamodel error is not too large, we will have $\alpha_{inf} \approx \alpha$ and, as $\alpha > 0$, the hypothesis $\alpha_{inf} > 0$ is realistic.
So, under this supplementary assumption, we have:

$$\operatorname*{argmin}_a R(a; \mathbf{y}, t, t') = -\frac{\beta}{2\alpha} \geq -\frac{\beta_{inf} + \delta_{inf}}{2\alpha_{inf}}$$

Now using the second part of (2.3.7) and the same reasoning on the non-positive quadratic function of a: $R(a; \mathbf{y}, t, t') - R_{sup}(a; \widetilde{\mathbf{y}}, \boldsymbol{\varepsilon}, t, t')$, we find that: $\alpha \leq \alpha_{sup}$, and: $\beta_{sup} - \delta_{sup} \leq \beta \leq \beta_{sup} + \delta_{sup}$. Hence,

$$\operatorname*{argmin}_a R(a; \mathbf{y}, t, t') \leq -\frac{\beta_{sup} - \delta_{sup}}{2\alpha_{sup}}$$

where $\delta_{sup} = 2\sqrt{(\alpha - \alpha_{sup})(\gamma - \gamma_{sup})}$. This comes without supplementary assumptions, because $\alpha_{sup} \geq \alpha$ and $\alpha > 0$, or else $R(\cdot; \mathbf{y}, t, t')$ would have no minimum (even if the case $\alpha = 0$ can seldom occur due to sampling fluctuations, the hypothesis Y not constant a.s. ensures that increasing N and/or changing the sample will lead to a case where $\alpha > 0$).
As we clearly have δ_{inf} and δ_{sup} smaller than (or equal to)

$$\widehat{\delta} := 2\sqrt{(\alpha_{inf} - \alpha_{sup})(\gamma_{inf} - \gamma_{sup})},$$

we deduce that:

$$-\frac{\beta_{inf}(t,t')+\widehat{\delta}(t,t')}{2\alpha_{inf}(t,t')} \leq \underset{a}{\operatorname{argmin}}\, R(a;\mathbf{y},t,t') \leq -\frac{\beta_{sup}(t,t')-\widehat{\delta}(t,t')}{2\alpha_{sup}(t,t')}$$

where we have made explicit the dependencies in t and t'.
To finish, it is easy to see that we have:

$$\overline{\mathcal{P}} := [\overline{\widetilde{y}} - \overline{\varepsilon}; \overline{\widetilde{y}} + \overline{\varepsilon}] \ni \overline{y} \tag{2.3.12}$$

and:

$$\overline{\mathcal{P}}' := [\overline{\widetilde{y}'} - \overline{\varepsilon}'; \overline{\widetilde{y}'} + \overline{\varepsilon}'] \ni \overline{y}' \tag{2.3.13}$$

(where $\overline{\widetilde{y}}, \overline{\widetilde{y}'}, \overline{\varepsilon}$ and $\overline{\varepsilon'}$ denote, respectively, the empirical means of $(\widetilde{y}_k)_k$, $(\widetilde{y}'_k)_k$, $(\varepsilon_k)_k$ and $(\varepsilon'_k)_k$) so that:

$$\min_{t\in\overline{\mathcal{P}},t'\in\overline{\mathcal{P}}'} \left(-\frac{\beta_{inf}(t,t')+\widehat{\delta}(t,t')}{2\alpha_{inf}(t,t')}\right) \leq \widehat{S}_i = \underset{a}{\operatorname{argmin}}\, R(a;\mathbf{y},t,t')$$

and:

$$\widehat{S}_i = \underset{a}{\operatorname{argmin}}\, R(a;\mathbf{y},t,t') \leq \max_{t\in\overline{\mathcal{P}},t'\in\overline{\mathcal{P}}'} \left(-\frac{\beta_{sup}(t,t')-\widehat{\delta}(t,t')}{2\alpha_{sup}(t,t')}\right)$$

Hence, (2.3.3) is verified with:

$$\widehat{S}_i^m = \min_{t\in\overline{\mathcal{P}},t'\in\overline{\mathcal{P}}'} \left(-\frac{\beta_{inf}(t,t')+\widehat{\delta}(t,t')}{2\alpha_{inf}(t,t')}\right), \quad \widehat{S}_i^M = \max_{t\in\overline{\mathcal{P}},t'\in\overline{\mathcal{P}}'} \left(-\frac{\beta_{sup}(t,t')-\widehat{\delta}(t,t')}{2\alpha_{sup}(t,t')}\right) \tag{2.3.14}$$

It is clear that $\widehat{S}_i^m$ and $\widehat{S}_i^M$ are computable without knowing the y_ks and y'_ks.
In practice, we compute approximate values of $\widehat{S}_i^m$ and $\widehat{S}_i^M$ by replacing the min and max over $\overline{\mathcal{P}} \times \overline{\mathcal{P}'}$ by the min and max over a finite sample $\Xi \subset \overline{\mathcal{P}} \times \overline{\mathcal{P}'}$.

2.3.2 Metamodel error in index estimation: a smoothed alternative

The bounds we presented in the last section are sometimes uninteresting, as, when ε is large enough with respect to f, the bounds $\widehat{S_i^m}$ and $\widehat{S_i^M}$ are not tight enough to be useful: it happens for example that $[0;1] \subset [\widehat{S_i^m}; \widehat{S_i^M}]$. Such a case is the RKHS interpolation metamodel example we present in Section 2.4.2.

In this section, we present an alternative to the bound described in the previous section. This alternative is interesting when ε is only moderately small with respect to f.

We begin by defining $\psi_i(\mathbf{z})$ as the Monte-Carlo sensitivity index estimator for variable i using $\mathbf{z} = (z_1, \ldots, z_N, z'_1, \ldots, z'_N)$ as function evaluations sample:

$$\psi_i(\mathbf{z}) = \frac{\overline{zz'} - \overline{z}\,\overline{z'}}{\overline{z^2} - \overline{z}^2}.$$

It is clear that:

$$\psi_i(y_1, \ldots, y_N, y'_1, \ldots, y'_N) = \widehat{S}_i,$$

where $\{y_k\}$ and $\{y'_k\}$ are defined in (2.2.2) and (2.2.3).

Also let:

$$\mathcal{Z} = \prod_{k=1}^{N} [\widetilde{y}_k - \varepsilon_k; \widetilde{y}_k + \varepsilon_k] \times \prod_{k=1}^{N} [\widetilde{y}'_k - \varepsilon'_k; \widetilde{y}'_k + \varepsilon'_k].$$

Now, since:

$$\forall k = 1, \ldots, N \quad y_k \in [\widetilde{y}_k - \varepsilon_k; \widetilde{y}_k + \varepsilon_k] \text{ and } y'_k \in [\widetilde{y}'_k - \varepsilon'_k; \widetilde{y}'_k + \varepsilon'_k],$$

we certainly have $(y_1, \ldots, y_N, y'_1, \ldots, y'_N) \in \mathcal{Z}$ and hence:

$$\min_{\mathbf{z} \in \mathcal{Z}} \psi_i(\mathbf{z}) \leq \widehat{S}_i \leq \max_{\mathbf{z} \in \mathcal{Z}} \psi_i(\mathbf{z}). \tag{2.3.15}$$

Minimizers and maximizers of optimization problems in (2.3.15) often display a very irregular (nonsmooth) behavior (as a function of the sampled inputs $\{\mathbf{x}_1, \ldots, \mathbf{x}_N, \mathbf{x}'_1, \ldots, \mathbf{x}'_N\}$), even for a smooth output f. This leads to overly pessimistic bounds in (2.3.15). To overcome this difficulty, we propose to ensure smoothness of the solution by introducing a penalty factor $\lambda \geq 0$ and to take:

$$S_i^{m,\lambda} = \psi_i(\mathbf{z}^{m,\lambda}) \quad S_i^{M,\lambda} = \psi_i(\mathbf{z}^{M,\lambda})$$

where:

$$\mathbf{z}^{m,\lambda} = \operatorname*{argmin}_{\mathbf{z} \in \mathcal{Z}} (\psi_i(\mathbf{z}) + \lambda \Pi(\mathbf{z})) \quad \mathbf{z}^{M,\lambda} = \operatorname*{argmax}_{\mathbf{z} \in \mathcal{Z}} (\psi_i(\mathbf{z}) - \lambda \Pi(\mathbf{z})) \tag{2.3.16}$$

for $\Pi(\mathbf{z})$ a (non-negative) indicator of the smoothness of the function which take $\mathbf{z}$ as values when evaluated on the input sample $\{\mathbf{x}_1, \ldots, \mathbf{x}_N, \mathbf{x}'_1, \ldots, \mathbf{x}'_N\}$. For instance, one may take:

$$\Pi(\mathbf{z}) = \frac{1}{2N} \sum_{k=1}^{2N} \left(z_k - z_k^S\right)^2 \tag{2.3.17}$$

with $\{z_k^S\}$ a kernel-smoothed version of $\{z_k\}$:

$$z_k^S = \frac{\sum_{k'=1}^{2N} K\left(\frac{\|\mathbf{x}_{k'}-\mathbf{x}_k\|}{h}\right) z_{k'}}{\sum_{k'=1}^{2N} K\left(\frac{\|\mathbf{x}_{k'}-\mathbf{x}_k\|}{h}\right)} \tag{2.3.18}$$

for K an appropriate kernel (preferably with compact support, for reasons of efficiency of the implementation), h a suitable bandwidth (in the sum we have set $\mathbf{x}_{k+N} = \mathbf{x}'_k$ for $k = 1, \ldots, N$), and $\|\cdot\|$ the Euclidean norm on $\mathbb{R}^p$. The main difficulty is that these optimization problems are in potentially large dimension $2N$. However, the gradient of the objective functions are analytically available – under (2.3.17) and (2.3.18) – and can be cheaply evaluated, so we choose to use a L-BFGS quasi-Newton algorithm, as implemented in `L-BFGS-B` ([118]), to solve problems (2.3.16).

Concerning the choice of λ and h, we opt for the following: h should be chosen so as to have a reasonable proportion (say, 1 to 5 percent) of "neighbor" points (that is to say, couples of points in $\{\mathbf{x}_k\}_{k=1,\ldots,N}$ having a significantly nonzero value on the kernel K). For the choice of λ: we plot, for a chosen h held fixed, the length $\left|S_i^{M,\lambda} - S_i^{m,\lambda}\right|$ as function of λ, and then choose λ to be the abscissa of the bottom-left corner of the "L-shaped" curve obtained.

2.3.3 Sampling error : bootstrap confidence intervals

Sampling error, due to the Monte-Carlo evaluation of the variances in (2.2.1), can be quantified through an approximate confidence interval calculated using bootstrap ([1]).

We use the bias-corrected (BC) percentile method presented in [31, 32]. The principle of this method can be summed up the following way: let $\widehat{\theta}(X_1, \ldots, X_n)$ be an estimator for an unknown parameter θ in a reference population $\mathcal{P}$. We generate a random i.i.d. n-sample $\{x_1, \ldots, x_n\}$ from $\mathcal{P}$, then we repeatedly, for $b = 1, \ldots, B$, randomly draw $\{x_1[b], \ldots, x_n[b]\}$ with replacement from this sample and get a *replication* of $\widehat{\theta}$ by computing $\widehat{\theta}[b] = \widehat{\theta}(x_1[b], \ldots, x_n[b])$. This way we obtain a set $\mathcal{R} = \{\widehat{\theta}[1], \ldots, \widehat{\theta}[B]\}$ of replications of $\widehat{\theta}$.

We now show how this sample can be used to estimate a confidence interval for θ. We denote by Φ the standard normal cdf:

$$\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} \exp\left(-\frac{t^2}{2}\right) \mathrm{d}t$$

and by Φ^{-1} its inverse.
Using $\mathcal{R}$ and the point estimate $\widehat{\theta} = \widehat{\theta}(x_1, \ldots, x_n)$, a "bias correction constant" z_0 can be estimated:

$$\widehat{z_0} = \Phi^{-1}\left(\frac{\#\{\widehat{\theta}[b] \in \mathcal{R} \text{ s.t. } \widehat{\theta}[b] \leq \widehat{\theta}\}}{B}\right)$$

where $\#A$ denotes the number of elements in the set A.
Then, for $\beta \in]0;1[$, we define the "corrected quantile estimate" $\widehat{q}(\beta)$:

$$\widehat{q}(\beta) = \Phi(2\widehat{z_0} + z_\beta)$$

where z_β satisfies $\Phi(z_\beta) = \beta$.
The central BC bootstrap confidence interval of level $1-\alpha$ is then estimated by the interval whose endpoints are the $\widehat{q}(\alpha/2)$ and $\widehat{q}(1-\alpha/2)$ quantiles of $\mathcal{R}$.
The BC bootstrap procedure is known to be robust to non-normality of $\widehat{\theta}$.
The key advantage of bootstrapping our sensitivity estimators is that we do not require supplementary model evaluations to estimate a confidence interval; hence the computational overhead for getting a confidence interval (versus pointwise estimation only) remains quite modest.

2.3.4 Combined confidence intervals

From Section 3.1 to 3.3, we have seen how to separately assess sampling error and metamodel error. To take both sources of error into account simultaneously, we propose using bootstrap confidence intervals (see Section 2.3.3) by calculating B bootstrap replications of $\widehat{S}_i^m$ and $\widehat{S}_i^M$, where, for $b = 1, \ldots, B$ each bootstrap pair $\left(\widehat{S}_i^m[b]; \widehat{S}_i^M[b]\right)$ is computed using $(\widetilde{y}_k)_{k\in L_b}, (\widetilde{y}'_k)_{k\in L_b}$ as surrogate output samples, and associated error bounds $(\widetilde{\varepsilon}_k)_{k\in L_b}, (\widetilde{\varepsilon}'_k)_{k\in L_b}$, where L_b is a list of N integers sampled with replacement from $\{1, \ldots, N\}$. Here the bounds $\widehat{S}_i^m$ and $\widehat{S}_i^M$ can be computed using either the method described in Section 2.3.1, or the one described in Section 2.3.2.
The BC bootstrap confidence interval procedure (see Section 2.3.3) can then be used to produce $(1-\alpha)$-level confidence intervals $[\widehat{S}^m_{i,\alpha/2}; \widehat{S}^m_{i,1-\alpha/2}]$ for S_i^m, and $[\widehat{S}^M_{i,\alpha/2}; \widehat{S}^M_{i,1-\alpha/2}]$ for S_i^M. We then take as combined confidence interval of level $1-\alpha$ for S_i the range $[\widehat{S}^m_{i,\alpha/2}; \widehat{S}^M_{i,1-\alpha/2}]$. This interval accounts for sampling and metamodels error simultaneously: let, for $b = 1, \ldots, B$, $\widehat{S}_i[b]$

be the b^{th} bootstrap replication, computed using the true outputs $(y_k)_{k\in L_b}$. As, for each b, we have:

$$\widehat{S}_i^m[b] \leq \widehat{S}_i[b] \leq \widehat{S}_i^M[b],$$

it follows that

$$\widehat{S}_{i,\alpha/2}^m \leq \widehat{S}_{i,\alpha/2}, \text{ and } \widehat{S}_{i,1-\alpha/2} \leq \widehat{S}_{i,1-\alpha/2}^M,$$

where $\widehat{S}_{i,\alpha/2}$ and $\widehat{S}_{i,1-\alpha/2}$ are the endpoints of the $(1-\alpha)$ BC-confidence interval computed using $(\widehat{S}_i[b])_b$ as replications. It follows that $[\widehat{S}_{i,\alpha/2}; \widehat{S}_{i,1-\alpha/2}]$ is contained in $[\widehat{S}_{i,\alpha/2}^m; \widehat{S}_{i,1-\alpha/2}^M]$ and, hence, the level of the latter interval is greater than the level of the former. As the level of the $[\widehat{S}_{i,\alpha/2}; \widehat{S}_{i,1-\alpha/2}]$ is (asymptotically) equal to $1-\alpha$, the asymptotic level of the combined confidence interval $[\widehat{S}_{i,\alpha/2}^m; \widehat{S}_{i,1-\alpha/2}^M]$ should theoretically be, by design, greater than $1-\alpha$.

2.4 Applications

2.4.1 Application to a reduced basis metamodel

The reduced basis method ([72, 78]) can be applied when the output is a functional of the discretized solution of a partial differential equation (PDE). In particular cases (for instance [72] for elliptic equations, [57] for viscous Burgers equation, [108] for parabolic equations), it has been shown that the reduced basis output error bounds are accurate and useful. In these cases, due to the intrusive nature of the reduced basis approach and "problem-dependent" considerations made during the construction of the metamodel, one can expect fastly-convergent, tight and fully justified error bounds for the sensitivity indices.

In this section, we test our combined confidence interval procedure described in Sections 2.3.1 and 2.3.4, and compare it with Monte-Carlo estimation on the full model (with bias-corrected bootstrap to assess sampling error). Our criteria for comparison are the CPU times needed to compute the intervals and the lengths of these intervals (the smaller the better). Note that the error bounds given by the reduced basis method are sufficiently small so that the "smoothness" bounds described in Section 2.3.2 give similar results to the method of Section 2.3.1.

In all our tests we take $\alpha = .05$ and $B = 2000$ bootstrap replications. We checked that this value of B is large enough by increasing B (ie., setting $B = 4000$), and notice that our results remain significantly unchanged.

Model set-up

Let u, a function of space $x \in [0;1]$ (note that space variable x is unrelated to input parameter vector $\mathbf{x}$) and time $t \in [0,T]$ ($T > 0$ is a fixed (i.e., known) parameter) satisfying the *viscous Burgers' equation*:

$$\frac{\partial u}{\partial t} + \frac{1}{2}\frac{\partial}{\partial x}(u^2) - \nu\frac{\partial^2 u}{\partial x^2} = \psi \tag{2.4.1}$$

where $\nu \in \mathbb{R}_*^+$ is the *viscosity*, and $\psi \in C^0\left([0,T], L^2(]0,1[)\right)$ is the *source term*.

For u to be well-defined, we also prescribe initial value $u_0 \in H^1(]0,1[)$ (ie. u_0 is in the Sobolev space of square-integrable functions whose first derivative is square-integrable), and continuous boundary values $b_0, b_1 \in C^0([0,T])$.

The initial u_0 and boundary values b_0 and b_1 are parametrized the following way:

$$\begin{aligned}
b_0(t) &= (u_{0m})^2 + \sum_{l=1}^{n(b_0)} A_l^{b_0} \sin(\omega_l^{b_0} t) \\
b_1(t) &= u_0(1) + \sum_{l=1}^{n(b_1)} A_l^{b_1} \sin(\omega_l^{b_1} t) \\
\psi(t,x) &= \psi_m + \sum_{l=1}^{n_T(\psi)} \sum_{p=1}^{n_S(\psi)} A_{lp}^{\psi} \sin(\omega_l^{\psi T} t) \sin(\omega_p^{\psi S} x) \\
u_0(x) &= (u_{0m})^2 + \sum_{l=1}^{n(u_0)} A_l^{u_0} \sin(\omega_l^{u_0} x)
\end{aligned}$$

The values of the angular frequencies $\omega_l^{b_0}$, $\omega_l^{b_1}$, $\omega_l^{\psi T}$, $\omega_p^{\psi S}$ and $\omega_l^{u_0}$, as well as their cardinalities $n(b_0)$, $n(b_1)$, $n_T(\psi)$, $n_S(\psi)$ and $n(u_0)$ are fixed (known), while our uncertain parameters, namely: viscosity ν, coefficients ψ_m and u_{0m}, and amplitudes $\left(A_l^{b_0}\right)_{l=1,\dots,n(b_0)}$, $\left(A_l^{b_1}\right)_{l=1,\dots,n(b_1)}$, $\left(A_{lp}^{\psi}\right)_{l=1,\dots,n_T(\psi);p=1,\dots,n_S(\psi)}$ and $\left(A_l^{u_0}\right)_{l=1,\dots,n(u_0)}$ live in some Cartesian product of intervals $\mathcal{P}$ defined by:

$$\mathcal{P} = \Big\{\mathbf{x} = (\nu, A_1^{b_0}, \ldots, A_{n(b_0)}^{b_0}, A_1^{b_1}, \ldots, A_{n(b_1)}^{b_0}, \psi_m, A_{11}^{\psi}, A_{12}^{\psi}, \ldots, A_{1,n_S(\psi)}^{\psi}, A_{2,1}^{\psi}, \ldots, A_{2,n_S(\psi)}^{\psi}, \ldots, A_{n_T(\psi),n_S(\psi)}^{\psi}, u_{0m}, A_1^{u_0}, \ldots, A_{n(u_0)}^{u_0})\Big\} \quad (2.4.2)$$

The solution $u = u(\mathbf{x})$ depends on the parameter vector $\mathbf{x}$ above.
The "full" model is obtained by discretizing the initial-boundary value problem, using a discrete time grid $\{t_k = k\Delta t\}_{k=0,\ldots,T/\Delta t}$, where $\Delta t > 0$ is the time step, and, space-wise, using $\mathbf{P}^1$ Lagrange finite elements built upon an uniform subdivision of $[0;1]$: $\{x_i = i/\mathcal{N}\}$, for $i = 0, \ldots, \mathcal{N}$. Our full output is:

$$f(\mathbf{x}) = f(u(\mathbf{x})) = \frac{1}{\mathcal{N}} \sum_{i=0}^{\mathcal{N}} u(t = T, x = x_i)$$

The reduced basis method is then applied to yield a surrogate solution $\widetilde{u}$ of (2.4.1), as well as an error bound ε_u on u. The reader can refer to [57] for full details on discretization, reduction and derivation of the RB error bound for this model. The main idea of the reduced basis method is to project u onto a well-chosen subspace of X whose dimension n (the reduced basis size) is smaller than the dimension of the finite element space that can be used to solve (2.4.1) numerically.
In our numerical experiments, we take $\mathcal{N} = 60$, $\Delta t = .01$, $T = .05$, $n_S(\psi) = n_T(\psi) = n(b_0) = n(b_1) = 0$, $n(u_0) = 1$, $\omega_1^{u_0} = 0.5$, $A_1^{u_0} = 5$ and $\psi_m = 1$.
The two input parameters are independent and uniformly sampled. The table below contains the ranges for them, and also the "true" values of the sensitivity indices, which have been calculated (in more than 14h CPU time) using a Monte-Carlo simulation with large sample size $N = 4 \times 10^6$ (so as to BC bootstrap confidence intervals of length $< 10^{-2}$) on the full model:

Parameter	Range	Confidence interval for sensitivity index
ν	[1 ; 20]	[0.0815;0.0832]
u_{0m}	[0 ; 0.3]	[0.9175;0.9182]

For benchmarking purposes, as they were computed using the true model with a large Monte-Carlo sample size, we can take the following values:

$$S_\nu = 0.082 \text{ and } S_{u_{0m}} = 0.918 \quad (2.4.3)$$

as "true" sensitivity indices.

Empirical coverage of combined confidence intervals

We test the performance of our combined confidence intervals by assessing their empirical *coverage*. This empirical coverage is measured by computing, for each input variable, 100 combined confidence intervals (each time with different input sample), and counting the proportion of intervals containing the true values in (2.4.3). The reduced basis size is $n = 9$; Monte-Carlo sample size is $N = 300$. Results are gathered in the table below:

Parameter	Mean confidence interval	Empirical coverage
ν	[0.0139;0.2083]	0.91
u_{0m}	[0.8421;0.9491]	0.87

Convergence benchmark

Figure 2.1 shows the lower $\widehat{S^m}$ and upper $\widehat{S^M}$ bounds (defined in Section 2.3.1) for different reduced basis sizes (hence different metamodel precisions) but fixed sample of size $N = 2000$, as well as the bootstrap confidence intervals computed using the procedure presented in Section 2.3.4. This figure exhibits the fast convergence of our bounds to the true value of the sensitivity index as the reduced basis size increases. We also see that the part of the error due to sampling remains constant, as sample size stays the same. We also notice that the true values lie between $\widehat{S}^m_{i,.025}$ and $\widehat{S}^M_{i,1-.025}$, and not always between $\widehat{S}^m_i$ and $\widehat{S}^M_i$; this is due to the Monte-Carlo part of the error, which is not taken into account by the $[\widehat{S}^m_i; \widehat{S}^M_i]$ bounds.

Choice of n and N

In practice, one wants to estimate sensitivity indices with a given precision (*ie.* to produce $(1-\alpha)$-level confidence intervals with prescribed length), and has no *a priori* indication on how to choose N and n to do so. Moreover, for one given precision, there may be multiple choices of suitable couples (N, n), balancing between sampling and metamodel error. Increasing N and/or n will increase the overall time for computation and improve the precision of the calculation (thanks to reduction in sampling error for increased N, or reduction in metamodel error for increased n).

We wish to choose the best compromise, that is, the one that gives the smallest computation time; for the reduced basis method, this time is roughly proportional to $N \times n^3$ (as a number proportional to N of linear systems of

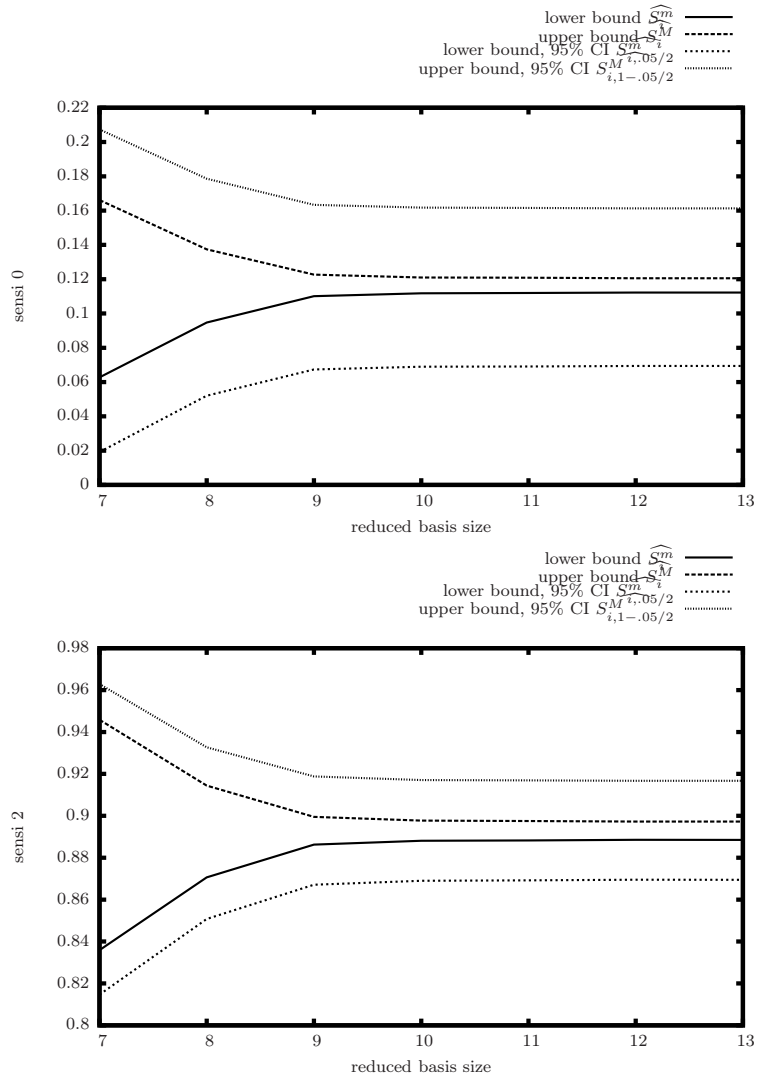


Figure 2.1: Convergence benchmark for sensitivity indices estimation in the Burgers' model, left: variable ν, right: variable u_{0m}. We plotted, for a fixed sample size $N = 2000$, estimator bounds $\widehat{S}_i^m$ and $\widehat{S}_i^M$ defined in (2.3.1), and endpoints $\widehat{S}_{i,.025}^m$ and $\widehat{S}_{i,1-.025}^M$ of the 95% confidence interval, for different reduced basis sizes.

size n have to be solved). On the other hand, the length of the combined confidence interval can be modelled as the sum of two terms:

$$\frac{Z_\alpha}{\sqrt{N}} + \frac{C}{a^n} \tag{2.4.4}$$

where Z_α, C are positive constants and $a > 1$ is a constant. The first term accounts for sampling error; its form is heuristically deduced from central limit theorem. The second accounts for metamodel error; its exponential decay is backed up by numerical experiments as well as theoretical works (e.g. [12]). The parameters Z_α, C and a can be found by doing "calibration runs" with a fixed small sample size N and different reduced basis sizes $n_1, \ldots, n_K$, and by regressing the obtained confidence interval lengths using (2.4.4).

Minimizing the computation time under the constraint that error is equal to a given "target precision" leads to a straightforward constrained optimization problem (P), which can be solved, after estimation of Z_α, C and n, to find (after rounding to the closest integer) the optimal n and N for a desired precision.

The results of the optimization procedure, for two different target precisions, can be found in the table below. We also present the mean confidence interval length:

$$\frac{1}{p}\sum_{i=1}^{p}\left(S^M_{i,1-\alpha/2} - S^m_{i,\alpha/2}\right),$$

to check that the obtained confidence intervals (CI) have the desired length.

	Target precision= 0.04	Target precision= 0.02
N	8600	22000
n	11	11
CI for S_ν	[0.0691614 ; 0.119096]	[0.0659997 ; 0.0937285]
CI for $S_{u_{0m}}$	[0.890985 ; 0.919504]	[0.914266 ; 0.926452]
Mean CI length	0.0389	0.0199

To check for the optimality of the choices of n and N, we ran the estimation procedure for different n and N but (approximately) fixed time budget $N \times n^3$. Results are in Table 2.1. Only the choice in the middle column, in italics, is the result of the optimization procedure described in this paragraph. We show that it leads to better (smaller) confidence intervals than the other choices.

N	5210	*8600*	23000
n	13	*11*	8
$N \times n^3$	11446370	*11446600*	11776000
Mean CI length	0.0416	*0.0389*	0.165

Table 2.1: Mean confidence interval lengths for different choices of n and N.

Full-scale example

We now present an example with more parameters. We take $\mathcal{N} = 300$, $\Delta t = .01$, $T = .05$, $n(\psi) = 0$, $n(u_0) = 5$, $n(b_0) = n(b_1) = 2$, $\psi_m = 1$, $\omega_1^{u_0} = .5$, $\omega_2^{u_0} = 1$, $\omega_3^{u_0} = 1.5$, $\omega_4^{u_0} = 2$, $\omega_5^{u_0} = 2.5$, $\omega_1^{b_0} = \omega_1^{b_1} = .3$ and $\omega_2^{b_0} = \omega_2^{b_1} = .6$. Input parameters are assumed independent and uniformly distributed in the ranges below:

Parameter(s)	Range
ν	[1 ; 20]
$u_{0m}, A_i^{u_0}$ $(i = 1, \ldots, 5)$	[0; 0.2]
$A_1^{b_0}$	[1;1.3]
$A_2^{b_0}$	[0;0.2]
$A_i^{b_1}$ $(i = 1, 2)$	[0;0.2]
f_m	[0;0.2]

We used a reduced basis of size $n = 13$. The combined confidence intervals obtained, for $N = 7000$, have mean length ≈ 0.08. Their computation required $N \times (p + 1) = 91000$ calls to the metamodel; these calls took 85 s of CPU time. By extrapolating the time necessary to generate the 200 (full) model evaluations for the choice of the reduced basis, we found that creating bootstrap confidence intervals of length 0.08 using the full model would require approximately 514 s. The use of a metamodel hence enabled a 6x speedup.

2.4.2 Application to a RKHS interpolation metamodel

We test our method using a RKHS interpolation metamodel based on the Ishigami function:

$$f(X_1, X_2, X_3) = \sin X_1 + 7 \sin^2 X_2 + 0.1 X_3^4 \sin X_1$$

for $(X_j)_{j=1,2,3}$ i.i.d. uniform in $[-\pi; \pi]$.

Learning sample size n	Variable (true index)	Mean 95% combined conf. int.	Mean length	Coverage
110	X_1 (0.3139)	[0.207;0.759]	0.552	0.94
110	X_2 (0.4424)	[0.0169;0.561]	0.545	0.98
110	X_3 (0)	[0;0.356]	0.357	1
130	X_1 (0.3139)	[0.262;0.626]	0.364	0.83
130	X_2 (0.4424)	[0.083;0.491]	0.408	0.93
130	X_3 (0)	[0;0.256]	0.257	1
160	X_1 (0.3139)	[0.274;0.509]	0.235	0.92
160	X_2 (0.4424)	[0.216;0.486]	0.27	0.92
160	X_3 (0)	[0;0.180]	0.18	1

Table 2.2: Obtained confidence intervals for the RKHS interpolation metamodel.

The analytical values of sensitivities are known:

$$S_1 = 0.3139, \quad S_2 = 0.4424, \quad S_3 = 0.$$

We refer to 2.5 for details on the RKHS methodology and on the method of computation of the error indicator.

The experimental design $\mathcal{D}$, of size denoted by n, have been generated using maximin latin hypercube samples (using `maximinLHS` of R `lhs` package ([17]). The R package `mlegp` ([27]) is used as our RKHS interpolation/Kriging toolbox.

Smoothing parameters are set to: $\lambda = .6$, $h = .2$. We computed, for different learning sample sizes n, the empirical coverages and mean lengths of the combined confidence intervals (with the "smoothed" alternative described in Section 2.3.2). Monte-Carlo sample size is $N = 1000$, and $B = 300$ bootstrap replicates are computed. Finally, as Sobol indices are always bounded by 0 and 1, the reported confidence interval is the intersection of the computed interval with $[0; 1]$ (note that one can instead shift the interval to be a subset of $[0; 1]$ while maintaining its original length as done in [100]). Results are given in Table 2.2.

The bound described in Section 2.3.1 has been tested on this example but did not give interesting results (ie. the produced intervals satisfied $[\widehat{S}_i^m; \widehat{S}_i^M] \supset [0; 1]$). The reason is that the RKHS error indicator remains large, even for the larger n values for which the metamodel construction is practicable.

We compared these results with the ones obtained with `CompModSA`, a software package implementing the methodology described in [100]. We used

Variable (true index)	Mean 95% Combined conf. int.	Mean length	Coverage
X_1 (0.3139)	[0.0164222;0.218084]	0.202	0.02
X_2 (0.4424)	[0.0877679;0.351651]	0.264	0.11
X_3 (0)	[0.00750793;0.173349]	0.166	0.82

Table 2.3: Confidence intervals for the RKHS interpolation metamodel obtained with `CompModSA`.

as parameters: surface='mlegp' (Kriging metamodel), `n.mc.T=0` (we do not want any total index computation), `n.mc.S=1000` (sample size), `n.samples=1` (one run), and `n.CI=300` (generate confidence intervals using 300 bootstrap replications). We contributed a patch for `CompModSA`, available at `http://ljk.imag.fr/membres/Alexandre.Janon/compmodsa.php`, which adds to `sensitivity` the option `CI.S`, set to `TRUE` to compute bootstrap confidence intervals for the main effect index. The results, for a learning sample of size $n = 160$, are shown in Table 2.3.
This comparison clearly shows that our method is able to account for metamodel error, so as to keep the actual coverage of the produced confidence interval close to the expected one. We have been unable to fully compare our approach with the one of [66], as the lengths of the confidence interval obtained with this approach were not available; however we can state that, in this example, our method produces intervals of correct coverages for every variable and every training set size we have tested.

Conclusion and perspectives

We presented a methodology to quantify the impact of the sampling error and the metamodel error on the sensitivity indices computation, when the metamodel provides a pointwise error bound (or, at least, an error indicator) on the output of interest. Sampling error is handled by a classic bootstrap procedure, while metamodel error is managed using bounds on the sensitivity index estimator. Quantification of those two types of errors permits a certification on the performed estimation We applied our method on two types of metamodels: intrusive (reduced basis) and non-intrusive (RKHS interpolation). On the applications we see that our approach performs well both for reduced-basis and RKHS interpolation. Our method can also be applied with other metamodels, as soon as an assessment for

pointwise metamodel error exists.

2.5 Appendix – RKHS interpolation and error bound

We now briefly recall the RKHS (reproducing kernel Hilbert space) interpolation method, and refer to [94] for the details. RKHS interpolation is known to be (see [95]) equivalent to interpolation by Kriging, also known as Gaussian process metamodelling. Using n evaluations of f, one can build a *training sample* consisting of n input-output pairs $\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$, where $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ is the *experimental design* and $y_i = f(\mathbf{x}_i)$ for $i = 1, \ldots, n$. Let $R(\cdot, \cdot)$ be a positive definite kernel; the RKHS interpolator of $\mathcal{D}$ with respect to R is:

$$\widetilde{f}(\mathbf{x}) = \mathbf{k}(\mathbf{x})^T \Sigma_S^{-1} \mathbf{y}_S$$

where:

$$\Sigma_S = \begin{pmatrix} R(\mathbf{x}_1, \mathbf{x}_1) & R(\mathbf{x}_1, \mathbf{x}_2) & \ldots & R(\mathbf{x}_1, \mathbf{x}_n) \\ R(\mathbf{x}_2, \mathbf{x}_1) & R(\mathbf{x}_2, \mathbf{x}_2) & \ldots & R(\mathbf{x}_2, \mathbf{x}_n) \\ \vdots & \vdots & \ddots & \vdots \\ R(\mathbf{x}_n, \mathbf{x}_1) & R(\mathbf{x}_n, \mathbf{x}_2) & \ldots & R(\mathbf{x}_n, \mathbf{x}_n) \end{pmatrix}$$

$$\mathbf{k}(\mathbf{x}) = \left(R(\mathbf{x}_1, \mathbf{x}), \ldots, R(\mathbf{x}_n, \mathbf{x})\right)^T, \qquad \mathbf{y}_S = \left(y_1, \ldots, y_n\right)^T$$

In practice, R is assumed to belong to a parametrized family; for instance, R is a gaussian kernel:

$$R(\mathbf{x}, \mathbf{y}) = \exp\left(-\sum_{j=1}^{p} \theta_j \left(x_j - y_j\right)^2\right) \tag{2.5.1}$$

whose parameters $\theta_1, \ldots, \theta_p$ are estimated from $\mathcal{D}$ by minimizing some contrast function.

If f belongs to the RKHS associated with the R kernel, there exist a constant C, depending only on f and R, so that:

$$\left|f(\mathbf{x}) - \widetilde{f}(\mathbf{x})\right| \leq C\sqrt{\sigma_Z^2(\mathbf{x})} \tag{2.5.2}$$

where:

$$\sigma_Z^2(\mathbf{x}) = R(\mathbf{x}, \mathbf{x}) - \mathbf{k}(\mathbf{x})^T \Sigma_S^{-1} \mathbf{k}(\mathbf{x})$$

We propose to estimate the constant C, which is, up to a multiplicative constant, the norm of f in the RKHS associated with R, by:

$$\widehat{C} = \max_{i=1,\ldots,n^\tau} \frac{\left|f(\mathbf{x}_i^\tau) - \widetilde{f}(\mathbf{x}_i^\tau)\right|}{\sqrt{\sigma_Z^2(\mathbf{x}_i^\tau)}}$$

where $\{\mathbf{x}_i^\tau\}_{i=1,\dots,n^\tau}$ is a "test sample" that does not contain any point of the experimental design (so as to ensure that $\sigma_Z^2(\mathbf{x}_i^\tau) \neq 0$ for all $i = 1, \dots, n^\tau$). For a continuous f, one easily sees that $\widehat{C}$ converges, as $\{\mathbf{x}_i^\tau\}$ fills the parameter set $\mathcal{P}$, to the smallest C so that (2.5.2) holds.

Acknowledgements: We wish to thank Jean-Claude Fort for a suggestion which we exploited to perform our computation of the least square metamodel-induced error bound. We also thank Anestis Antoniadis and Ingrid Van Keilegom for advice on bootstrap methodology, and Robert Miller for language remarks. We finally thank anonymous referees for helpful comments about the overall presentation of the paper. – This work has been partially supported by the French National Research Agency (ANR) through COSINUS program (project COSTA-BRAVA n°ANR-09-COSI-015).

Chapter 3

Asymptotic normality and efficiency of two Sobol index estimators

Résumé: De nombreux modèles mathématiques font intervenir plusieurs paramètres qui ne sont pas tous connus précisément. L'analyse de sensibilité globale se propose de sélectionner les paramètres d'entrée dont l'incertitude a le plus d'impact sur la variabilité d'une quantité d'intérêt, sortie du modèle. Un des outils statistiques pour quantifier l'influence de chacune des entrées sur la sortie est l'indice de sensibilité de Sobol. Nous considérons l'estimation statistique de cet indice à l'aide d'un nombre fini d'échantillons de sorties du modèle: nous présentons deux estimateurs de cet indice et énonçons un théorème central limite pour chacun d'eux. Nous démontrons que l'un de ces deux estimateurs est optimal en terme de variance asymptotique. Nous généralisons également nos résultats au cas où la vraie sortie du modèle n'est pas observée, mais où seule une version dégradée (bruitée) de la sortie est disponible.

Abstract: Many mathematical models involve input parameters, which are not precisely known. Global sensitivity analysis aims to identify the parameters whose uncertainty has the largest impact on the variability of a quantity of interest (output of the model). One of the statistical tools used to quantify the influence of each input variable on the output is the Sobol sensitivity index. We consider the statistical estimation of this index

from a finite sample of model outputs: we present two estimators and state a central limit theorem for each. We show that one of these estimators has an optimal asymptotic variance. We also generalize our results to the case where the true output is not observable, and is replaced by a noisy version.

Introduction

Many mathematical models encountered in applied sciences involve a large number of poorly-known parameters as inputs. It is important for the practitioner to assess the impact of this uncertainty on the model output. An aspect of this assessment is sensitivity analysis, which aims to identify the most sensitive parameters, that is, parameters having the largest influence on the output. In global stochastic sensitivity analysis (see for example [88] and references therein) the input variables are assumed to be independent random variables. Their probability distributions account for the practitioner's belief about the input uncertainty. This turns the model output into a random variable, whose total variance can be split down into different partial variances (this is the so-called Hoeffding decomposition see [109]). Each of these partial variances measures the uncertainty on the output induced by each input variable uncertainty. By considering the ratio of each partial variance to the total variance, we obtain a measure of importance for each input variable that is called the *Sobol index* or *sensitivity index* of the variable [97]; the most sensitive parameters can then be identified and ranked as the parameters with the largest Sobol indices.

Once the Sobol indices have been defined, the question of their effective computation or estimation remains open. In practice, one has to estimate (in a statistical sense) those indices using a finite sample (of size typically in the order of hundreds of thousands) of evaluations of model outputs [48]. Indeed, many Monte Carlo or quasi Monte Carlo approaches have been developed by the experimental sciences and engineering communities. This includes the FAST methods (see for example [23], [105] and references therein) and the Sobol pick-freeze (SPF) scheme (see [97, 98]). In SPF a Sobol index is viewed as the regression coefficient between the output of the model and its pick-freezed replication. This replication is obtained by holding the value of the variable of interest (frozen variable) and by sampling the other variables (picked variables). The sampled replications are then combined to produce an estimator of the Sobol index. In this paper we study very deeply this Monte Carlo method in the general framework where one or more variables can be frozen. This allows to define sensitivity indices with respect to a general random input living in a probability space (groups of variables, random vectors, random processes...). In this work, we study and compare two

Sobol index estimators based on the SPF scheme; the first estimator, denoted by S_N^X, is well-known, the second, denoted by T_N^X has not, to our best knowledge, been considered in the literature so far. For both estimators, we show convergence and give the rate of convergence; we also show that T_N^X is optimal (in terms of asymptotic variance) amongst regular estimators which are functions of the pick-freezed replications – this feature is called *asymptotic efficiency* and is a generalization of the notion of minimum variance unbiased estimator (see [109] chapters 8 and 25 or [52] for more details).

The SPF method requires many (typically, around one thousand times the number of input variables) evaluations of the model output. In many interesting cases, an evaluation of the model output is made by a complex computer code (for instance, a numerical partial differential equation solving algorithm) whose running time is not negligible (typically in the order of the second or the minute) for one single evaluation. When thousands of such evaluations have to be made, one generally replaces the original *exact* model by a faster-to-run *metamodel* (also known in the literature as *surrogate model* or *response surface* [10]) which is an approximation of the true model. Well-known metamodels include Kriging [92], polynomial chaos expansion [102] and reduced bases [72, 57], to name a few. When a metamodel is used, the estimated Sobol indices are tainted by a twofold error: *sampling error*, due to the replacement of the original, infinite population of all the possible inputs by a finite sample, and *metamodel error*, due to the replacement of the original model by an approximative metamodel.

The goal of this paper is to study the asymptotic behavior of these two errors on Sobol index estimation in the double limit where the sample size goes to infinity and the metamodel converges to the true model. Some work has been done on the non-asymptotic error quantification in Sobol index estimation in earlier papers [100, 66, 58] by means of confidence intervals which account for both sampling and metamodel errors. In this paper, we give necessary and sufficient conditions on the rate of convergence of the metamodel to the exact model for asymptotic normality of a natural Sobol index estimator to hold. The asymptotic normality allows us to produce asymptotic confidence intervals in order to assess the quality of our estimation. We also give sufficient conditions for a metamodel-based estimator to be asymptotically efficient. Asymptotic efficiency of an other Sobol index estimator has already been considered in [25]. In this work, the authors

were interested in the asymptotic efficiency for local polynomial estimates of Sobol indices. Our approach proposes an estimator which has a simpler form, is less computationally intensive and is more precise in practice. Moreover, we derive results also in the case where the full model is replaced by a metamodel.

This paper is organized as follows: in the first section, we set up the notation, review the definition of Sobol indices and give two estimators of interest. In the second section, we prove asymptotic normality and asymptotic efficiency when the sample of outputs comes from the true model. These two properties are generalized in the third section where metamodel error is taken into account. The fourth section gives numerical illustrations on benchmark models and metamodels.

3.1 Definition and estimation of Sobol indices

3.1.1 Exact model

The output $Y \in \mathbb{R}$ is a function of independent random input variables $X \in \mathbb{R}^{p_1}$ and $Z \in \mathbb{R}^{p_2}$. In other words, Y and (X, Z) are linked by the relation

$$Y = f(X, Z) \tag{3.1.1}$$

where f is a deterministic function defined on $\mathcal{P} \subset \mathbb{R}^{p_1+p_2}$. We denote by $p = p_1 + p_2$ the total number of inputs of f.

In the paper X' will denote an independent copy of X. We also write $Y^X = f(X, Z')$.

We assume that Y is square integrable and non deterministic ($\mathrm{Var}Y \neq 0$). We are interested in the following Sobol index:

$$S^X = \frac{\mathrm{Var}\left(\mathbb{E}(Y|X)\right)}{\mathrm{Var}(Y)} \in [0; 1]. \tag{3.1.2}$$

This index quantifies the influence on the X input on the output Y: a value of S^X that is close to 1 indicates that X is highly influent on Y.

Remark 1. *All the results in this paper readily apply when X is multidimensional. In this case, S^X is usually called the* closed sensitivity index *of X (see [91]).*

3.1.2 Estimation of S^X

The next lemma shows how to express S^X using covariances. This will lead to a natural estimator which has already been considered in [49].

Lemma 1. *Assume that the random variables X and Z are square integrable. Then*

$$\mathrm{Var}(\mathbb{E}(Y|X)) = \mathrm{Cov}(Y, Y^X).$$

In particular

$$S^X = \frac{\mathrm{Cov}\left(Y, Y^X\right)}{\mathrm{Var}(Y)}. \tag{3.1.3}$$

Proof. On one hand, since $Y \overset{\mathcal{L}}{=} Y^X$ (that is, Y and Y^X have the same distribution), we have

$$\mathrm{Cov}(Y, Y^X) = \mathbb{E}(YY^X) - \mathbb{E}(Y)\mathbb{E}(Y^X) = \mathbb{E}(YY^X) - \mathbb{E}(Y)^2.$$

On the other hand, Y and Y^X are independent conditionally on X, so that

$$\mathbb{E}(YY^X) = \mathbb{E}(\mathbb{E}(YY^X|X)) = \mathbb{E}(\mathbb{E}(Y|X)\mathbb{E}(Y^X|X)) = \mathbb{E}(\mathbb{E}(Y|X)^2).$$

□

Remark 2. *Using a classical regression result, we see that*

$$S^X = \underset{a\in\mathbb{R}}{\mathrm{argmin}} \left\{ \mathbb{E}\left((Y^X - \mathbb{E}(Y^X)) - a(Y - \mathbb{E}(Y)) \right)^2 \right\}. \tag{3.1.4}$$

A first estimator. In view of Lemma 1, we are now able to define a first natural estimator of S^X (all sums are taken for i from 1 to N):

$$S_N^X = \frac{\frac{1}{N}\sum Y_i Y_i^X - \left(\frac{1}{N}\sum Y_i\right)\left(\frac{1}{N}\sum Y_i^X\right)}{\frac{1}{N}\sum Y_i^2 - \left(\frac{1}{N}\sum Y_i\right)^2},$$

where, for $i = 1, \ldots, N$:

$$Y_i = f(X_i, Z_i), \quad Y_i^X = f(X_i, Z_i'),$$

and $\{(X_i, Z_i)\}_{i=1,\ldots,N}$ and $\{(X_i, Z_i')\}_{i=1,\ldots,N}$ are two independent and identically distributed (i.i.d.) samples of the distribution of (X, Z), with $\{Z_i\}_i$ independent of $\{Z_i'\}_i$.

This estimator has been considered in [49], where it has been showed to be a practically efficient estimator.

A second estimator. We can take into account the observation of $\{Y_i^X\}_{1\leq i\leq N}$ to make an estimation of $\mathbb{E}(Y)$ and $\mathrm{Var}(Y)$ which is expected to perform better than any other based on $\{Y_i\}_{1\leq i\leq N}$ only. We propose the following estimator:

$$T_N^X = \frac{\frac{1}{N}\sum Y_i Y_i^X - \left(\frac{1}{N}\sum\left[\frac{Y_i+Y_i^X}{2}\right]\right)^2}{\frac{1}{N}\sum\left[\frac{Y_i^2+(Y_i^X)^2}{2}\right] - \left(\frac{1}{N}\sum\left[\frac{Y_i+Y_i^X}{2}\right]\right)^2}. \tag{3.1.5}$$

To our best knowledge, this estimator has not been considered in the literature. We will clarify what we mean when saying that T_N^X performs better than S_N^X in Proposition 7, Section 3.2.2 and Subsection 3.4.1.

3.2 Asymptotic properties: exact model

3.2.1 Consistency and asymptotic normality

Throughout all the paper, we denote by $\mathcal{N}_k(\mu,\Sigma)$ the k-dimensional Gaussian distribution with mean μ and covariance matrix Σ, and, given any sequence of random variables $\{R_n\}_{n\in\mathbb{N}}$, we note

$$\overline{R}_N = \frac{1}{N}\sum_{n=1}^{N} R_n.$$

Proposition 5 (Consistency)**.** *We have:*

$$S_N^X \xrightarrow[N\to\infty]{a.s.} S^X \tag{3.2.1}$$

$$T_N^X \xrightarrow[N\to\infty]{a.s.} S^X. \tag{3.2.2}$$

Proof. The result is a straightforward application of the strong law of large numbers and that $\mathbb{E}(Y)=\mathbb{E}(Y^X)$ and $\mathrm{Var}(Y)=\mathrm{Var}(Y^X)$. □

Proposition 6 (Asymptotic normality)**.** *Assume that* $\mathbb{E}(Y^4)<\infty$. *Then*

$$\sqrt{N}\left(S_N^X - S^X\right) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}_1\left(0, \frac{\mathrm{Var}\left((Y-\mathbb{E}(Y))\left[(Y^X-\mathbb{E}(Y)) - S^X(Y-\mathbb{E}(Y))\right]\right)}{(\mathrm{Var}Y)^2}\right) \tag{3.2.3}$$

and

$$\sqrt{N}\left(T_N^X - S^X\right) \underset{N\to\infty}{\overset{\mathcal{L}}{\to}} \mathcal{N}_1\left(0, \sigma_T^2\right) \tag{3.2.4}$$

where

$$\sigma_T^2 = \frac{\mathrm{Var}\left((Y - \mathbb{E}(Y))(Y^X - \mathbb{E}(Y)) - S^X/2\left((Y - \mathbb{E}(Y))^2 + (Y^X - \mathbb{E}(Y))^2\right)\right)}{(\mathrm{Var}Y)^2}.$$

Proof of (3.2.3). We begin by noticing that S_N^X is invariant by any centering (translation) of the Y_i and Y_i^X. To simplify the next calculations, we suppose that they have been recentred by $-\mathbb{E}(Y)$. By setting:

$$U_i = \left((Y_i - \mathbb{E}(Y))(Y_i^X - \mathbb{E}(Y)), \quad Y_i - \mathbb{E}(Y), \quad Y_i^X - \mathbb{E}(Y), \quad (Y_i - \mathbb{E}(Y))^2\right)^T, \tag{3.2.5}$$

this implies that:

$$S_N^X = \Phi(\overline{U}_N)$$

with:

$$\Phi(x, y, z, t) = \frac{x - yz}{t - y^2}$$

The central limit theorem gives that:

$$\sqrt{N}\left(\overline{U}_N - \mu\right) \underset{N\to\infty}{\overset{\mathcal{L}}{\to}} \mathcal{N}_4\left(0, \Gamma\right)$$

where Γ is the covariance matrix of U_1 and:

$$\mu = \begin{pmatrix} \mathrm{Cov}(Y, Y^X) \\ 0 \\ 0 \\ \mathrm{Var}(Y) \end{pmatrix}.$$

The so-called Delta method [109] (Theorem 3.1) gives:

$$\sqrt{N}\left(S_N^X - S^X\right) \underset{N\to\infty}{\overset{\mathcal{L}}{\to}} \mathcal{N}_1(0, g^T \Gamma g)$$

where:

$$g = \nabla\Phi(\mu).$$

Note that since by assumption $\mathrm{Var}Y \neq 0$, Φ is differentiable at μ, so that the application of the Delta method is justified. By differentiation, we get that, for any x, y, z, t so that $t \neq y^2$:

$$\nabla\Phi(x, y, z, t) = \left(\frac{1}{t - y^2}, \quad \frac{-z(t - y^2) + (x - yz)\cdot 2y}{(t - y^2)^2}, \quad -\frac{y}{t - y^2}, \quad -\frac{x - yz}{(t - y^2)^2}\right)^T$$

so that, by using (3.1.3):

$$g = \left(\frac{1}{\mathrm{Var}Y}, \quad 0, \quad 0, \quad -\frac{S^X}{\mathrm{Var}Y} \right)^T.$$

Hence

$$\begin{aligned} g^T \Gamma g &= \frac{\mathrm{Var}\left((Y-\mathbb{E}(Y))(Y^X-\mathbb{E}(Y))\right)}{(\mathrm{Var}Y)^2} + \frac{(S^X)^2}{(\mathrm{Var}Y)^2}\mathrm{Var}\left((Y-\mathbb{E}(Y))^2\right) \\ &\quad -2\frac{S^X}{(\mathrm{Var}Y)^2}\mathrm{Cov}\left((Y-\mathbb{E}(Y))(Y^X-\mathbb{E}(Y)),(Y-\mathbb{E}(Y))^2\right) \\ &= \frac{1}{(\mathrm{Var}Y)^2}\Big(\mathrm{Var}\left((Y-\mathbb{E}(Y))(Y^X-\mathbb{E}(Y))\right) + \mathrm{Var}\left(S^X\left((Y-\mathbb{E}(Y))^2\right)\right) \\ &\quad -2\mathrm{Cov}\left((Y-\mathbb{E}(Y))(Y^X-\mathbb{E}(Y)), S^X(Y-\mathbb{E}(Y))^2\right)\Big) \\ &= \frac{\mathrm{Var}\left((Y-\mathbb{E}(Y))\left[(Y^X-\mathbb{E}(Y)) - S^X(Y-\mathbb{E}(Y))\right]\right)}{(\mathrm{Var}Y)^2}, \end{aligned}$$

which is the announced result.

Proof of (3.2.4). As in the previous point, it is easy to check that T_N^X is invariant with respect to translations of Y_i and Y_i^X by $-\mathbb{E}(Y)$. Thus, $T_N^X = \Psi\left(\overline{W}_N\right)$ with:

$$\Psi(x,y,z) = \frac{x-(y/2)^2}{z/2-(y/2)^2}$$

and:

$$W_i = \Big((Y_i-\mathbb{E}(Y))(Y_i^X-\mathbb{E}(Y)), \quad (Y_i-\mathbb{E}(Y)) + (Y_i^X-\mathbb{E}(Y)), \\ (Y_i-\mathbb{E}(Y))^2 + (Y_i^X-\mathbb{E}(Y))^2\Big)^T. \quad (3.2.6)$$

By the central limit theorem,

$$\sqrt{N}\left(\overline{W}_N - \begin{pmatrix} \mathrm{Cov}(Y,Y^X) \\ 0 \\ 2\mathrm{Var}Y \end{pmatrix}\right) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}_3(0,\Sigma)$$

where Σ is the covariance matrix of W_1.

The Delta method for Ψ gives:

$$\sqrt{N}\left(T_N^X - S^X\right) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}_1(0, g^T\Sigma g)$$

where g is now:

$$g = \nabla\Psi(\mathrm{Cov}(Y, Y^X), \quad 0, \quad 2\mathrm{Var}Y).$$

We have, for any x, y, z so that $z \neq y^2/2$:

$$\nabla\Psi(x,y,z) = \Big(\frac{1}{z/2-(y/2)^2}, \quad \frac{-y\left(z/2-(y/2)^2-x-(y/2)^2\right)}{\left(z/2-(y/2)^2\right)^2}, \\ -\frac{1}{2}\frac{x-(y/2)^2}{\left(z/2-(y/2)^2\right)^2}\Big)^T.$$

Hence

$$g = \left(\frac{1}{\mathrm{Var}Y}, \quad 0, \quad -\frac{1}{2}\frac{S^X}{\mathrm{Var}Y}\right)^T$$

and we have

$$\begin{aligned} g^T\Sigma g &= \frac{1}{(\mathrm{Var}Y)^2}\mathrm{Var}\left((Y-\mathbb{E}(Y))(Y^X-\mathbb{E}(Y))\right) \\ &\quad +\frac{1}{4}\frac{(S^X)^2}{(\mathrm{Var}Y)^2}\mathrm{Var}\left((Y-\mathbb{E}(Y))^2+(Y^X-\mathbb{E}(Y))^2\right) \\ &\quad -2\frac{1}{(\mathrm{Var}Y)^2}\frac{1}{2}S^X \\ &\quad \times\mathrm{Cov}\left((Y-\mathbb{E}(Y))(Y^X-\mathbb{E}(Y)),(Y-\mathbb{E}(Y))^2+(Y^X-\mathbb{E}(Y))^2\right) \\ &= \frac{1}{(\mathrm{Var}Y)^2} \\ &\quad \times\mathrm{Var}\left((Y-\mathbb{E}(Y))(Y^X-\mathbb{E}(Y))-\frac{S^X}{2}\left((Y-\mathbb{E}(Y))^2+(Y^X-\mathbb{E}(Y))^2\right)\right). \quad \square \end{aligned}$$

Proposition 7. *The asymptotic variance of T_N^X is always less than or equal to the asymptotic variance of S_N^X, with equality if and only if $S^X = 0$ or $S^X = 1$.*

To prove this Proposition, we need the following immediate Lemma:

Lemma 2. *Y and Y^X are exchangeable random variables, ie.*

$$(Y, Y^X) \stackrel{\mathcal{L}}{=} (Y^X, Y).$$

Proof of Proposition 7. The difference between the asymptotic variances is

equal to:

$$
\begin{aligned}
&- 2S^X \Bigg[\mathrm{Cov}\left((Y-\mathbb{E}(Y))(Y^X-\mathbb{E}(Y)), (Y-\mathbb{E}(Y))^2\right) \\
&\quad - \mathrm{Cov}\left((Y-\mathbb{E}(Y))(Y^X-\mathbb{E}(Y)), \frac{1}{2}\left((Y-\mathbb{E}(Y))^2 + (Y^X-\mathbb{E}(Y))^2\right)\right)\Bigg] \\
&\qquad + (S^X)^2 \Bigg[\mathrm{Var}\left((Y-\mathbb{E}(Y))^2\right) - \frac{1}{4}\left(2\mathrm{Var}\left((Y-\mathbb{E}(Y))^2\right)\right. \\
&\qquad \left. + 2\mathrm{Cov}\left((Y-\mathbb{E}(Y))^2, (Y^X-\mathbb{E}(Y))^2\right)\right] + o(1). \quad (3.2.7)
\end{aligned}
$$

Thanks to exchangeability of Y and Y^X, we have that:

$$
\begin{aligned}
&\mathrm{Cov}\left((Y-\mathbb{E}(Y))(Y^X-\mathbb{E}(Y)), (Y-\mathbb{E}(Y))^2\right) \\
&\qquad = \mathrm{Cov}\left((Y-\mathbb{E}(Y))(Y^X-\mathbb{E}(Y)), (Y^X-\mathbb{E}(Y))^2\right)
\end{aligned}
$$

hence the first term of the right-hand side of (3.2.7) is zero.
For the second term, we use Cauchy-Schwarz inequality to see that:

$$
\begin{aligned}
\mathrm{Cov}\left((Y-\mathbb{E}(Y))^2, (Y^X-\mathbb{E}(Y))^2\right) &\leq \sqrt{\mathrm{Var}\left((Y-\mathbb{E}(Y))^2\right)\mathrm{Var}\left((Y^X-\mathbb{E}(Y))^2\right)} \\
&= \mathrm{Var}\left((Y-\mathbb{E}(Y))^2\right)
\end{aligned}
$$

so the second term is always non-negative. This proves that the asymptotic variance of S_N^X is greater than the asymptotic variance of T_N^X.
For the equality case, we notice that $S^X = 0$ implies the equality of the asymptotic variances. If $S^X \neq 0$, equality holds if and only if there is equality in Cauchy-Schwarz, ie. there exists $k \in \mathbb{R}$ so that:

$$
(Y-\mathbb{E}(Y))^2 = k(Y^X-\mathbb{E}(Y))^2
$$

by taking expectations and using $\mathrm{Var}Y = \mathrm{Var}Y^X$ we see that $k = 1$ necessarily, hence $Y = Y^X$ almost surely, and $S^X = 1$ thanks to (3.1.3). □

3.2.2 Asymptotic efficiency

In this section we study the asymptotic efficiency of S_N^X and T_N^X. This notion (see [109], Section 25 for its definition) extends the notion of Cramér-Rao bound to the semiparametric setting and enables to define a criteria of optimality for estimators, called asymptotic efficiency.

Let $\mathcal{P}$ be the set of all cumulative distribution functions (cdf) of exchangeable random vectors in $L^2(\mathbb{R}^2)$. It is clear that the cdf Q of a random vector of $L^2(\mathbb{R}^2)$ is in $\mathcal{P}$ if and only if Q is symmetric:

$$Q(a,b) = Q(b,a) \quad \forall (a,b) \in \mathbb{R}^2.$$

Let P be the cdf of (Y, Y^X). We have $P \in \mathcal{P}$ thanks to Lemma 2.

Proposition 8 (Asymptotic efficiency)**.** *$\{T_N^X\}_N$ is asymptotically efficient for estimating S^X in $\mathcal{P}$.*

We will use the following Lemma, which is also of interest in its own right:

Lemma 3 (Asymptotic efficiency in $\mathcal{P}$)**.** *1. Let $\Phi_1 : \mathbb{R} \to \mathbb{R}$ be a function in $L^2(P)$. The sequence of estimators $\left\{\Phi_N^1\right\}_N$ given by:*

$$\Phi_N^1 = \frac{1}{N} \sum \frac{\Phi_1(Y_i) + \Phi_1(Y_i^X)}{2}$$

is asymptotically efficient for estimating $\mathbb{E}(\Phi_1(Y))$ in $\mathcal{P}$.

2. Let $\Phi_2 : \mathbb{R}^2 \to \mathbb{R}$ be a symmetric function in $L^2(P)$. The sequence $\left\{\Phi_N^2\right\}_N$ given by:

$$\Phi_N^2 = \frac{1}{N} \sum \Phi_2 \left(Y_i, Y_i^X\right)$$

is asymptotically efficient for estimating $\mathbb{E}(\Phi_2(Y, Y^X))$ in $\mathcal{P}$.

Proof of Lemma 3. Let, for $g \in L^2(P)$ and $t \in \mathbb{R}$, P_t^g be the cdf satisfying:

$$\mathrm{d}P_t^g = (1 + tg)\mathrm{d}P.$$

It is clear that $\{P_t^g\}_{t \in \mathbb{R}} \subset \mathcal{P}$ if and only if $g \in \dot{\mathcal{P}}_P$, where:

$$\dot{\mathcal{P}}_P = \{g \in L^2(P) \text{ s.t. } \mathbb{E}(g(Y, Y^X)) = 0 \text{ and } g(a,b) = g(b,a) \forall (a,b) \in \mathbb{R}^2\}$$

is the tangent set of $\mathcal{P}$ at P.
Let, for $Q \in \mathcal{P}$:

$$\Psi_1(Q) = \mathbb{E}_Q\left(\Phi_1(Y)\right) \quad \text{and} \quad \Psi_2(Q) = \mathbb{E}_Q\left(\Phi_2(Y, Y^X)\right).$$

We recall that $\mathbb{E}_Q$ denotes the expectation obtained by assuming that the random vector (Y, Y^X) follows the Q distribution.
Following [109] Section 25.3, we compute the efficient influence functions of Ψ_1 and Ψ_2 with respect to $\mathcal{P}$ and the tangent set $\dot{\mathcal{P}}_P$. These empirical influence functions are related to the minimal asymptotic variance of a regular estimator sequence whose observations lie in $\mathcal{P}$ (op.cit., Theorems 25.20 and 25.21). Let $g \in \dot{\mathcal{P}}_P$.

1. We have

$$\begin{aligned}\frac{\Psi_1(P_t^g)-\Psi_1(P)}{t} &= \mathbb{E}_P\left(\Phi_1(Y)g(Y,Y^X)\right)\\ &= \mathbb{E}_P\left[\left(\frac{\Phi_1(Y)+\Phi_1(Y^X)}{2}-\mathbb{E}(\Phi_1(Y))\right)g(Y,Y^X).\right]\end{aligned}$$

As:

$$\widetilde{\Psi_{1,P}}=\frac{\Phi_1(Y)+\Phi_1(Y^X)}{2}-\mathbb{E}(\Phi_1(Y))\in\dot{\mathcal{P}}_P,$$

it is the efficient influence function of Ψ_1 at P. Hence the efficient asymptotic variance is:

$$\mathbb{E}_P\left(\left(\widetilde{\Psi_{1,P}}\right)^2\right)=\frac{\mathrm{Var}\left(\Phi_1(Y)+\Phi_1(Y^X)\right)}{4}.$$

As, by the central limit theorem, $\left\{\Phi_N^1\right\}$ clearly achieves this efficient asymptotic variance, it is an asymptotically efficient estimator of $\Psi_1(P)$.

2. We have:

$$\begin{aligned}\frac{\Psi_2(P_t^g)-\Psi_2(P)}{t} &= \mathbb{E}_P\left(\Phi_2(Y,Y^X)g(Y,Y^X)\right)\\ &= \mathbb{E}_P\left[\left(\Phi_2(Y,Y^X)-\mathbb{E}(\Phi_2(Y,Y^X))\right)g(Y,Y^X)\right].\end{aligned}$$

Thanks to the symmetry of Φ_2, we have that

$$\widetilde{\Psi_{2,P}}=\Phi_2(Y,Y^X)-\mathbb{E}(\Phi_2(Y,Y^X))$$

belongs to $\dot{\mathcal{P}}_P$, hence it is the efficient influence function of Ψ_2. So the efficient asymptotic variance is:

$$\mathbb{E}_P\left(\left(\widetilde{\Psi_{2,P}}\right)^2\right)=\mathrm{Var}\left(\Phi_2(Y,Y^X)\right),$$

and this variance is achieved by $\left\{\Phi_N^2\right\}$. □

Proof of Proposition 8. By Lemma 3, we get that:

$$U_N=\left(\frac{1}{N}\sum_{i=1}^N Y_iY_i^X,\quad \frac{1}{N}\sum_{i=1}^N\frac{Y_i+Y_i^X}{2},\quad \frac{1}{N}\sum_{i=1}^N\frac{Y_i^2+(Y_i^X)^2}{2}\right) \tag{3.2.8}$$

is asymptotically efficient, componentwise, for estimating

$$U=\left(\mathbb{E}(YY^X),\quad \mathbb{E}(Y),\quad \mathbb{E}(Y^2)\right) \tag{3.2.9}$$

in $\mathcal{P}$.
Using Theorem 25.50 (efficiency in product space) of [109], we can deduce joint efficiency from this componentwise efficiency.
Now, let Ψ be the function defined by:

$$\Psi(x, y, z) = \frac{x - y^2}{z - y^2}$$

and Ψ is differentiable on:

$$\mathbb{R}^3 \setminus \left\{(x, y, z)/z \neq y^2\right\},$$

Theorem 25.47 (efficiency and Delta method) of [109] implies that $\{\Psi(U_N)\}$ is asymptotically efficient for estimating $\Psi(U)$ in $\mathcal{P}$. The conclusion follows, as $\Psi(U_N) = T_N^X$ and $\Psi(U) = S^X$. □

3.3 Asymptotic properties: metamodel

3.3.1 Metamodel-based estimation

As said in the introduction, we often are in a situation where the exact output f is too costly to be evaluated numerically (thus, Y and Y^X are not observable variables in our estimation problem) and has to be replaced by a metamodel $\widetilde{f}$, which is a faster to evaluate approximation of f. We view this approximation as a perturbation of the exact model by some function δ:

$$\widetilde{Y} = \widetilde{f}(X, Z) = f(X, Z) + \delta,$$

where the perturbation $\delta = \delta(X, Z, \xi)$ is also a function of a random variable ξ independent from X and Z.
We also define, as before

$$\widetilde{Y}^X = \widetilde{f}(X, Z').$$

Assuming again that $\widetilde{Y}$ is non deterministic and in L^2, we can consider the following vector of Sobol indices with respect to the metamodel:

$$\widetilde{S}^X = \frac{\mathrm{Var}(\mathbb{E}(\widetilde{Y}|Z))}{\mathrm{Var}(\widetilde{Y})} \tag{3.3.1}$$

and its estimators:

$$\widetilde{S}_N^X = \frac{\frac{1}{N}\sum \widetilde{Y}_i \widetilde{Y}_i^X - \left(\frac{1}{N}\sum \widetilde{Y}_i\right)\left(\frac{1}{N}\sum \widetilde{Y}_i^X\right)}{\frac{1}{N}\sum \widetilde{Y}_i^2 - \left(\frac{1}{N}\sum \widetilde{Y}_i\right)^2} \tag{3.3.2}$$

$$\widetilde{T}_N^X = \frac{\frac{1}{N}\sum \widetilde{Y}_i\widetilde{Y}_i^X - \left(\frac{1}{N}\sum\left[\frac{\widetilde{Y}_i+\widetilde{Y}_i^X}{2}\right]\right)^2}{\frac{1}{N}\sum\left[\frac{\widetilde{Y}_i^2+(\widetilde{Y}_i^X)^2}{2}\right] - \left(\frac{1}{N}\sum\left[\frac{\widetilde{Y}_i+\widetilde{Y}_i^X}{2}\right]\right)^2}. \tag{3.3.3}$$

The goal of this section is to give sufficient conditions on the perturbation δ for $\widetilde{S}_N^X$ and $\widetilde{T}_N^X$ to satisfy asymptotic normality (Subsection 3.3.2), and $\widetilde{T}_N^X$ to be asymptotically efficient (Subsection 3.3.3), with respect to the Sobol index of the *true* model S^X.

3.3.2 Consistency and asymptotic normality

In the first Subsection (3.3.2) we suppose that the error term δ does not depend on N. In this case, if the Sobol index of the exact model is different from the Sobol index of the metamodel, then neither consistency nor asymptotic normality are possible. In the second subsection (3.3.2), we let δ depend on N and we give conditions for consistency and asymptotic normality to hold.

First case : δ does not depend on N

Proposition 9. *If $\widetilde{S}^X - S^X \neq 0$ then neither $\widetilde{S}_N^X$ nor $\widetilde{T}_N^X$ are consistent for estimating S^X.*

Proof. We have

$$\widetilde{S}_N^X - S^X = (\widetilde{S}_N^X - \widetilde{S}^X) + (\widetilde{S}^X - S^X).$$

The first term converges to 0 almost surely by Proposition 5 applied to $\widetilde{S}_N^X$. However, the second is nonzero by assumption.
The proof for the $\widetilde{T}_N^X$ estimator is exactly the same. □

This Proposition shows that it is impossible to have asymptotic normality for $\widetilde{S}_N^X$ and $\widetilde{T}_N^X$ in any nontrivial case if δ does not vanish (in some sense) asymptotically. This justifies the consideration of cases where δ depends on N, and this is the object of the next subsection.

Second case : $\mathrm{Var}\,\delta_N$ converges to 0 as $N \to \infty$

We now assume that the perturbation δ is a function of the sample size N. This entails that $\widetilde{f}$, as well as $\widetilde{Y}$, $\widetilde{Y}^X$ and $\widetilde{S}^X$ depend on N. We emphasize

this dependence by using the notations δ_N, $\widetilde{f}_N$, $\widetilde{Y}_N$, $\widetilde{Y}_N^X$. We keep, however, using the notations $\widetilde{S}_N^X$ and $\widetilde{T}_N^X$ for the estimators of $\widetilde{S}^X$ defined at (3.3.2) and (3.3.3).
We further assume (until the end of the chapter) that $\delta_N \xrightarrow[N\to+\infty]{L^2} c$ for some constant c.

Proposition 10. *We have* $\widetilde{S}^X \underset{N\to+\infty}{\longrightarrow} S^X$.

Proof. We clearly have that $\widetilde{Y}_N \xrightarrow[N\to+\infty]{L^2} Y + c$.
We deduce that:

$$\mathrm{Var}\left(\widetilde{Y}_N\right) \underset{N\to+\infty}{\longrightarrow} \mathrm{Var}(Y+c) = \mathrm{Var}(Y)$$

and

$$\mathbb{E}(\widetilde{Y}_N|Z) \underset{N\to+\infty}{\longrightarrow} \mathbb{E}(Y|Z) + c \text{ in } L^2.$$

From this last convergence we get

$$\mathrm{Var}\left(\mathbb{E}(\widetilde{Y}_N|Z)\right) \underset{N\to+\infty}{\longrightarrow} \mathrm{Var}\left(\mathbb{E}(Y|Z)\right).$$

This proves that $\widetilde{S}^X = \mathrm{Var}\left(\mathbb{E}(\widetilde{Y}_N|Z)\right)/\mathrm{Var}\left(\widetilde{Y}_N\right)$ converges to

$$S^X = \mathrm{Var}\left(\mathbb{E}(Y|Z)\right)/\mathrm{Var}(Y)$$

when N goes to $+\infty$. □

Proposition 11. *Assume there exist* $s > 0$ *and* $C > 0$ *such that*

$$\forall N, \quad \mathbb{E}\left(\left|\widetilde{Y}_N\right|^{4+s}\right) < C. \tag{3.3.4}$$

Then

$$\sqrt{N}\left(\widetilde{S}_N^X - \widetilde{S}^X\right) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}_1\left(0, \sigma_S^2\right) \tag{3.3.5}$$

$$\sqrt{N}\left(\widetilde{T}_N^X - \widetilde{S}^X\right) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}_1\left(0, \sigma_T^2\right) \tag{3.3.6}$$

where σ_S^2 *and* σ_T^2 *are the asymptotic variances of* S_N^X *and* T_N^X *given, respectively, in* (3.2.3) *and* (3.2.4).

Proof. **Proof of** (3.3.5)**.** Let

$$\widetilde{U}_{N,i} = \left((\widetilde{Y}_{N,i} - \mathbb{E}(Y))(\widetilde{Y}_{N,i}^X - \mathbb{E}(Y)), \widetilde{Y}_{N,i} - \mathbb{E}(Y), \widetilde{Y}_{N,i}^X - \mathbb{E}(Y), \left(\widetilde{Y}_{N,i} - \mathbb{E}(Y)\right)^2\right)$$

and

$$\overline{\widetilde{U}}_N := \frac{1}{N}\sum_{i=1}^{N}\widetilde{U}_{N,i}.$$

Using the Lindeberg-Feller central limit theorem (see e.g. [109] 2.27, with $Y_{N,i} = \widetilde{U}_{N,i}/\sqrt{N}$), we get:

$$\sqrt{N}\left(\overline{\widetilde{U}}_N - \mathbb{E}\left(\widetilde{U}_{1,1}\right)\right) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}_4(0,\Gamma)$$

where Γ is the covariance matrix of the U_1 vector defined in (3.2.5).
The use of this central limit theorem is justified by the fact that, under assumption (3.3.4) of uniform boundedness of moments of $\widetilde{Y}_N$, there are $s' > 0$ and C' such that:

$$\forall N,\ \ \mathbb{E}(||U_{N,i}||^{2+s'}) < C'$$

where $\|\cdot\|$ is the standard Euclidean norm.
This ensures

$$\forall \varepsilon > 0,\ \ \mathbb{E}(||\tilde{U}_{N,i}||^2 \mathbf{1}_{||\tilde{U}_{N,i}|| > \varepsilon\sqrt{N}}) \to 0.$$

Then

$$\mathbb{E}(||\tilde{U}_{N,i}||^2 \mathbf{1}_{||\tilde{U}_{N,i}|| > \varepsilon\sqrt{N}}) = \mathbb{E}\left(\frac{||\tilde{U}_{N,i}||^{2+s'}}{||\tilde{U}_{N,i}||^{s'}}\mathbf{1}_{||\tilde{U}_{N,i}|| > \varepsilon\sqrt{N}}\right) \leq \frac{C'}{\varepsilon^{s'} N^{s'/2}}.$$

This shows that for each i, $\left\{\left\|\tilde{U}_{N,i}\right\|^2\right\}_N$ is uniformly integrable, hence, the variance-covariance matrix of $\tilde{U}_{N,i}$ converges to Γ when $N \to +\infty$. As $\widetilde{U}_{N,i} \xrightarrow[N\to+\infty]{\mathbb{P}} U_i$, the same convergence holds in L^2 and the covariance matrices of $\widetilde{U}_{N,i}$ converge (as $N \to +\infty$) to Γ, the covariance matrix of U_i.
We conclude the proof by applying the Delta method as for the exact model (cf. the proof of Proposition 6).

Proof of (3.3.6). We set:

$$\widetilde{W}_{N,i} = \Big((\widetilde{Y}_{N,i} - \mathbb{E}(Y))(\widetilde{Y}^X_{N,i} - \mathbb{E}(Y)),\quad (\widetilde{Y}_{N,i} - \mathbb{E}(Y)) + (\widetilde{Y}^X_{N,i} - \mathbb{E}(Y)),$$
$$(\widetilde{Y}_{N,i} - \mathbb{E}(Y))^2 + (\widetilde{Y}^X_{N,i} - \mathbb{E}(Y))^{2T}.$$

As in the previous point, the Lindeberg-Feller theorem can be applied to $\left\{\widetilde{W}_{N,i}\right\}$ to yield the convergence:

$$\sqrt{N}\left(\overline{\widetilde{W}}_N - \mathbb{E}\left(\widetilde{W}_{1,1}\right)\right) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}_3(0,\Sigma)$$

where Σ is the covariance matrix of W_1 defined in (3.2.6). The conclusion follows again by an application of the Delta method as in the proof of Proposition 6. □

We are actually interested in the asymptotic distribution of $\sqrt{N}\left(\widetilde{S}_N^X - S^X\right)$. In the remaining of the subsection, we will show that this convergence depends on the rate of convergence to 0 of $\mathrm{Var}(\delta_N)$.

Theorem 1. *Let:*

$$C_{\delta,N} = 2\mathrm{Var}(Y)^{1/2}\left[\mathrm{Corr}(Y,\delta_N^X) - \mathrm{Corr}(Y,Y^X)\mathrm{Corr}(Y,\delta_N)\right] + \mathrm{Var}(\delta_N)^{1/2}\left[\mathrm{Corr}(\delta_N,\delta_N^X) - \mathrm{Corr}(Y,Y^X)\right],$$

for $\delta_N^X = \delta_N(X,Z',\xi')$, and, given any L^2 random variables A and B of nonzero variance:

$$\mathrm{Corr}(A,B) = \frac{\mathrm{Cov}(A,B)}{\sqrt{\mathrm{Var}A\,\mathrm{Var}B}}.$$

Assume that $C_{\delta,N}$ does not converge to 0.

1. *If $\mathrm{Var}(\delta_N) = o\left(\frac{1}{N}\right)$, then asymptotic normalities of $\widetilde{S}_N^X$ and $\widetilde{T}_N^X$ for S^X hold, i.e.*

$$\sqrt{N}(\widetilde{S}_N^X - S^X) \underset{N\to+\infty}{\longrightarrow} \mathcal{N}(0,\sigma_S^2) \tag{3.3.7}$$

and:

$$\sqrt{N}(\widetilde{T}_N^X - S^X) \underset{N\to+\infty}{\longrightarrow} \mathcal{N}(0,\sigma_T^2). \tag{3.3.8}$$

2. *If $N\mathrm{Var}(\delta_N) \to \infty$, then $\widetilde{S}_N^X$ and $\widetilde{T}_N^X$ are not asymptotically normal for S^X.*

3. *If $C_{\delta,N}$ converges to a positive constant C and $\mathrm{Var}(\delta_N) = \frac{\gamma}{CN} + o\left(\frac{1}{N}\right)$, then:*

$$\sqrt{N}(\widetilde{S}_N^X - S^X) \underset{N\to+\infty}{\longrightarrow} \mathcal{N}(\gamma,\sigma_S^2),$$

and:

$$\sqrt{N}(\widetilde{T}_N^X - S^X) \underset{N\to+\infty}{\longrightarrow} \mathcal{N}(\gamma,\sigma_T^2).$$

Remark 3. *Obviously, if $C_{\delta,N}$ converges to 0, then asymptotic normalities of $\widetilde{S}_N^X$ and $\widetilde{T}_N^X$ hold under weaker assumptions on* $\mathrm{Var}(\delta_N)$.

Proof of Theorem 1. The following decompositions:

$$\sqrt{N}(\widetilde{S}_N^X - S^X) = \sqrt{N}(\widetilde{S}_N^X - \widetilde{S}^X) + \sqrt{N}(\widetilde{S}^X - S^X) \tag{3.3.9}$$

$$\sqrt{N}(\widetilde{T}_N^X - S^X) = \sqrt{N}(\widetilde{T}_N^X - \widetilde{S}^X) + \sqrt{N}(\widetilde{S}^X - S^X) \tag{3.3.10}$$

make obvious that if $\sqrt{N}(\widetilde{S}^X - S^X)$ goes to some constant κ then

$$\sqrt{N}(\widetilde{S}_N - S) \xrightarrow[N\to+\infty]{\mathcal{L}} \mathcal{N}(\kappa, \sigma_S^2)$$

and:

$$\sqrt{N}(\widetilde{T}_N - S) \xrightarrow[N\to+\infty]{\mathcal{L}} \mathcal{N}(\kappa, \sigma_T^2).$$

The second point of the theorem is now clear from the proof of Proposition 9.

The remaining of the theorem is an immediate consequence of Lemma 4 below. □

Lemma 4. *We have:*

$$\sqrt{N}\left(\widetilde{S}^X - S^X\right) = O\left((N\mathrm{Var}(\delta_N))^{1/2}\right).$$

Proof. We have:

$$\begin{aligned}
\widetilde{S}^X - S^X &= \frac{\mathrm{Cov}(\widetilde{Y}_N, \widetilde{Y}_N^X)}{\mathrm{Var}\widetilde{Y}_N} - \frac{\mathrm{Cov}(Y, Y^X)}{\mathrm{Var}(Y)} \\
&= \frac{\mathrm{Cov}(Y, Y^X) + 2\mathrm{Cov}(Y, \delta_N^X) + \mathrm{Cov}(\delta_N, \delta_N^X)}{\mathrm{Var}(Y) + 2\mathrm{Cov}(Y, \delta_N) + \mathrm{Var}(\delta_N)} - \frac{\mathrm{Cov}(Y, Y^X)}{\mathrm{Var}(Y)} \\
&= \frac{\mathrm{Var}(Y)\left(2\mathrm{Cov}(Y, \delta_N^X) + \mathrm{Cov}(\delta_N, \delta_N^X)\right)}{\mathrm{Var}(Y)\left(\mathrm{Var}(Y) + 2\mathrm{Cov}(Y, \delta_N) + \mathrm{Var}(\delta_N)\right)} \\
&\quad - \frac{\mathrm{Cov}(Y, Y^X)\left(2\mathrm{Cov}(Y, \delta_N) + \mathrm{Var}(\delta_N)\right)}{\mathrm{Var}(Y)\left(\mathrm{Var}(Y) + 2\mathrm{Cov}(Y, \delta_N) + \mathrm{Var}(\delta_N)\right)} \\
&= \frac{\mathrm{Var}(\delta_N)^{1/2} C_\delta}{\mathrm{Var}(Y) + 2\mathrm{Cov}(Y, \delta_N) + \mathrm{Var}(\delta_N)}
\end{aligned}$$

and:

$$\mathrm{Var}(Y) + 2\mathrm{Cov}(Y, \delta_N) + \mathrm{Var}(\delta_N) = \mathrm{Var}(Y) + o(1).$$

Finally, $C_{\delta,N}$ is uniformly bounded because $\mathrm{Var}(\delta_N)$ goes to 0 and $\mathrm{Var}(Y)$ is a constant. □

3.3.3 Asymptotic efficiency

Proposition 12 (Asymptotic efficiency for the metamodel)**.** *Assume*

1. $\exists s > 0, C > 0$ *s.t.* $\forall N$, $\mathbb{E}\left(|Y|^{4+s}\right) < C$ *and* $\mathbb{E}\left(|\widetilde{Y}|^{4+s}\right) < C$ *;*
2. $N\mathrm{Var}(\delta_N) \to 0$ *;*
3. $\sqrt{N}\mathbb{E}(\delta_N) \to 0$.

Then $\left\{\widetilde{T}_N^X\right\}$ *is asymptotically efficient for estimating* S^X.

Remark 4. *By Minkowski inequality, the first hypothesis implies* $\mathbb{E}(\delta_N^{4+s}) < 2C^{\frac{1}{4+s}}$ *and the asymptotic normality by Lemma 4 and Theorem 1.*

The proposition above will be proved using the following lemma.

Lemma 5. *For all* $N \in \mathbb{N}^*$*, let* $(Z_{N,i})_{i=1,\dots,N}$ *be a sequence of i.i.d variables such that*

1. $\sqrt{N}\mathbb{E}(Z_{N,i}) \underset{N\to+\infty}{\longrightarrow} 0$*;*
2. $\mathrm{Var}(Z_{N,i}) \underset{N\to+\infty}{\longrightarrow} 0$.

Then

$$\frac{1}{\sqrt{N}}\sum_{i=1}^{N} Z_{N,i} \underset{N\to+\infty}{\overset{\mathbb{P}}{\longrightarrow}} 0.$$

Proof. The result follows after the following decomposition:

$$\frac{1}{\sqrt{N}}\sum_{i=1}^{N} Z_{N,i} = \sqrt{N}\left(\frac{1}{N}\sum_{i=1}^{N} Z_{N,i} - \mathbb{E}(Z_{N,1})\right) + \sqrt{N}\mathbb{E}(Z_{N,1}). \qquad \square$$

Proof of Proposition 12. Let U_N and U be the vectors defined in the proof of Proposition 8, in (3.2.8) and (3.2.9), respectively, and:

$$\widetilde{U}_N = \left(\frac{1}{N}\sum_{i=1}^{N}\widetilde{Y}_{N,i}\widetilde{Y}_{N,i}^X, \quad \frac{1}{N}\sum_{i=1}^{N}\frac{\widetilde{Y}_{N,i}+\widetilde{Y}_{N,i}^X}{2}, \quad \frac{1}{N}\sum_{i=1}^{N}\frac{\widetilde{Y}_{N,i}^2+(\widetilde{Y}_{N,i}^X)^2}{2}\right).$$

We will show that:

$$\sqrt{N}\left(U_N - \widetilde{U}_N\right) \underset{N\to\infty}{\overset{\mathbb{P}}{\to}} 0. \tag{3.3.11}$$

By Theorem 25.23 of [109] and the fact that (U_N) is asymptotically efficient for U (shown in the proof of Proposition 8), this implies that $\left(\widetilde{U}_N\right)$ is asymptotically efficient for U, and the end of the proof of Proposition 8 shows the announced result.

To prove (3.3.11), it is sufficient to prove componentwise convergence. We will treat the second and the third components, as the result holds in the same way for the other.
For the second component, we have

$$\frac{1}{\sqrt{N}}\sum_{i=1}^{N}(\widetilde{Y}_{N,i}-Y_i)=\frac{1}{\sqrt{N}}\sum_{i=1}^{N}\delta_{N,i}$$

goes to 0 (in probability) by the previous lemma. The same holds for $\frac{1}{\sqrt{N}}\sum_{i=1}^{N}(\widetilde{Y}^X_{N,i}-Y_i^X)$.
For the third component, we have

$$\frac{1}{\sqrt{N}}\sum_{i=1}^{N}(\widetilde{Y}^2_{N,i}-Y_i^2)=2\frac{1}{\sqrt{N}}\sum_{i=1}^{N}\delta_{N,i}Y_i+\frac{1}{\sqrt{N}}\sum_{i=1}^{N}\delta^2_{N,i}.$$

Now by assumption,

$$\sqrt{N}\mathbb{E}(\delta_{N,i}Y_i)\leq\sqrt{N\mathbb{E}(\delta^2_{N,i})\mathbb{E}(Y_i^2)}=\sqrt{N(\mathrm{Var}(\delta_{N,i})+\mathbb{E}(\delta_{N,i})^2)\mathbb{E}(Y_i^2)}\to 0,$$

and by Cauchy-Schwarz inequality,

$$\mathrm{Var}(\delta_{N,i}Y_i)=\mathbb{E}(\delta^2_{N,i}Y_i^2)-(\mathbb{E}(\delta_{N,i}Y_i))^2\leq\sqrt{\mathbb{E}(\delta^4_{N,i})\mathbb{E}(Y_i^4)}+\mathbb{E}(\delta^2_{N,i})\mathbb{E}(Y_i^2)\leq C\mathbb{E}(\delta_N^4)^{1/2}.$$

By assumption, for all i, $\delta_{N,i}\xrightarrow[N\to+\infty]{\mathbb{P}}0$. Hence, the same convergence holds about $\delta^4_{N,i}$. Since δ_N is in L^{4+s}, then $\left\{\delta_N^4\right\}_N$ is uniformly integrable and we get the convergence of $\mathbb{E}(\delta_N^4)$ to 0 when $N\to+\infty$.
We conclude by the lemma above. Again, the same convergence occurs for $\frac{1}{\sqrt{N}}\sum_{i=1}^{N}(\widetilde{(Y^X_{N,i})^2}-(Y_i^X)^2)$. □

3.4 Numerical illustrations

In this section, we illustrate the asymptotic results of Sections 3.2.1 and 3.3.2 when the exact model is the Ishigami function [55]:

$$f(X_1,X_2,X_3)=\sin X_1+7\sin^2 X_2+0.1X_3^4\sin X_1 \tag{3.4.1}$$

for $(X_j)_{j=1,2,3}$ are i.i.d. uniform random variables in $[-\pi;\pi]$. In this case, all the integrability conditions are satisfied (we even have $Y\in L^\infty$).

The Sobol index of f with respect to input variable X_1 is S^X defined in (3.1.2) for $X = X_1$ and $Z = (X_2, X_3)$; we denote it by S^1. Similarly, S^2 (resp. S^3) is S^X obtained taking $X = X_2$ and $Z = (X_1, X_3)$ (resp. $X = X_3$ and $Z = (X_1, X_2)$).

Exact values of these indices are analytically known:

$$S^1 = 0.3139, \quad S^2 = 0.4424, \quad S^3 = 0.$$

For a sample size N, a risk level $\alpha \in]0;1[$ and for each input variable, a confidence interval for S^X (S^X being one of S^1, S^2 or S^3) of confidence level $1-\alpha$ can be estimated – using evaluations of the true model f – by approximating the distribution of S_N^X (or T_N^X) by its Gaussian distribution given in Proposition 3.2.3, using empirical estimators of the asymptotic variances stated in this Proposition.

In the case where only a perturbated model (metamodel) $\widetilde{f}_N = f + \delta_N$ is available, a confidence interval can still be estimated by using the $\widetilde{S}_N^X$ (or $\widetilde{T}_N^X$) estimator.

Thanks to Proposition 11, the level of the resulting confidence interval should be close to $1-\alpha$ for sufficiently large values of N if (and only if) $\mathrm{Var}\delta_N$ decreases sufficiently quickly with N.

The levels of the obtained confidence interval can be estimated by computing a large number R of confidence interval replicates, and by considering the empirical coverage, that is, the proportion of intervals containing the true index value; it is well known that this empirical coverage strongly converges to the level of the interval as R goes to infinity.

In the next subsections, we present the estimations of the levels of the confidence interval for the Ishigami model (3.4.1) using the true model (Subsection 3.4.1), and, with various synthetic model perturbations (Subsections 3.4.2 and 3.4.3), as well as RKHS (Kriging) metamodels (Subsection 3.4.4) and nonparametric regression metamodels (Subsection 3.4.5). We begin by comparing S_N and T_N on the exact model (Subsection 3.4.1), then we illustrate the generalization to the metamodel case on the widespread estimator S_N; the condition to ensure asymptotic normality in the metamodel is the same for S_N and T_N. All simulations have been made with $R = 1000$ and $\alpha = 0.05$.

3.4.1 Exact model

Figure 3.1 shows the empirical coverage of the asymptotic confidence interval built using the S_N^X estimator, plotted as a function of the sample size N. The theoretical level 0.95 is represented with a dotted line. Figure 3.2 does the same using the T_N^X estimator.

We see that the coverages get closer to the target level 0.95 as N increases, thereby assessing the reliability of the asymptotic confidence interval.

Figure 3.3 compares the efficiency of S_N^X and T_N^X by plotting the confidence interval lengths for the two estimators, as functions of the sample size. As the lengths for both estimators are $O(1/\sqrt{N})$, we plot the lengths multiplied by $\sqrt{N}$. We see that T_N^X always produce smaller confidence intervals, except for X_3 where the lengths are sensibly the same; this conclusion fully agrees with Proposition 7.

3.4.2 Gaussian-perturbated model

We consider a perturbation $\widetilde{f}_N$ of the original output f:

$$\widetilde{f}_N = f + \frac{5\xi}{N^{\beta/2}}$$

where $\beta > 0$ and ξ is a standard Gaussian.

The perturbation $\delta_N = 5\frac{\xi}{N^{\beta/2}}$ leads to $\mathrm{Var}\delta_N \propto N^{-\beta}$. Since:

$$C_\delta = O\left(\mathrm{Var}(\delta_N)^{1/2}\right) = O\left(N^{-\beta/2}\right),$$

the proof of Theorem 1 shows that $\widetilde{S}_N$ is asymptotically normal for S if $\beta > 1/2$. For indices relative to X_1 and X_2, this sufficient condition is also necessary, as C_δ is actually equivalent to $N^{-\beta/2}$. For X_3, we have $C_\delta = 0$ so that $\widetilde{S}_N$ is asymptotically normal for S for any positive β.

This is illustrated for $N = 50000$ in Figure 3.4. We see that the empirical coverages of the confidence interval for S^1 and S^2 jump to 0.95 near $\beta = 1/2$, while, for S^3, this coverage is always close to 0.95.

3.4.3 Weibull-perturbated model

We now take a different perturbation of the output:

$$\widetilde{f}_N = f + \frac{5WX_3^2}{N^{\beta/2}}$$

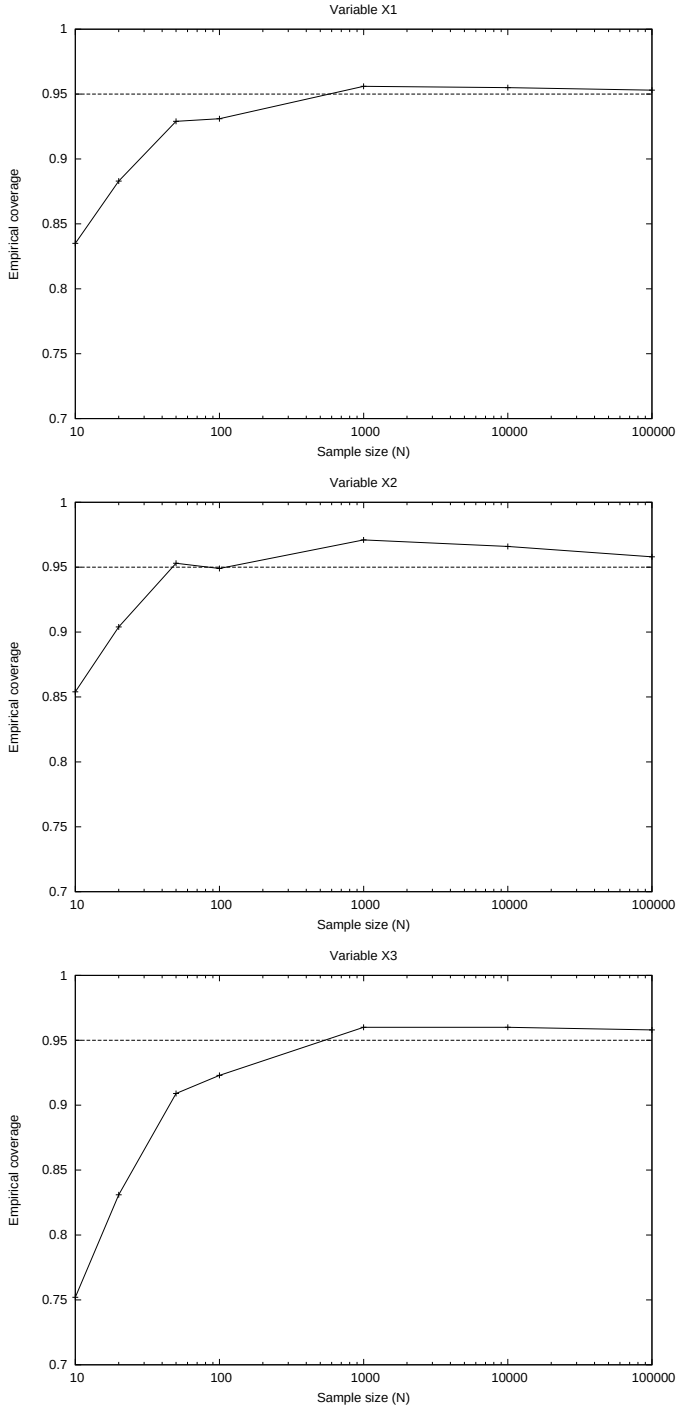


Figure 3.1: Empirical coverages of asymptotic confidence intervals for S^1 (left), S^2 (center) and S^3 (right), as a function of the sample size. The S_N estimator is used.

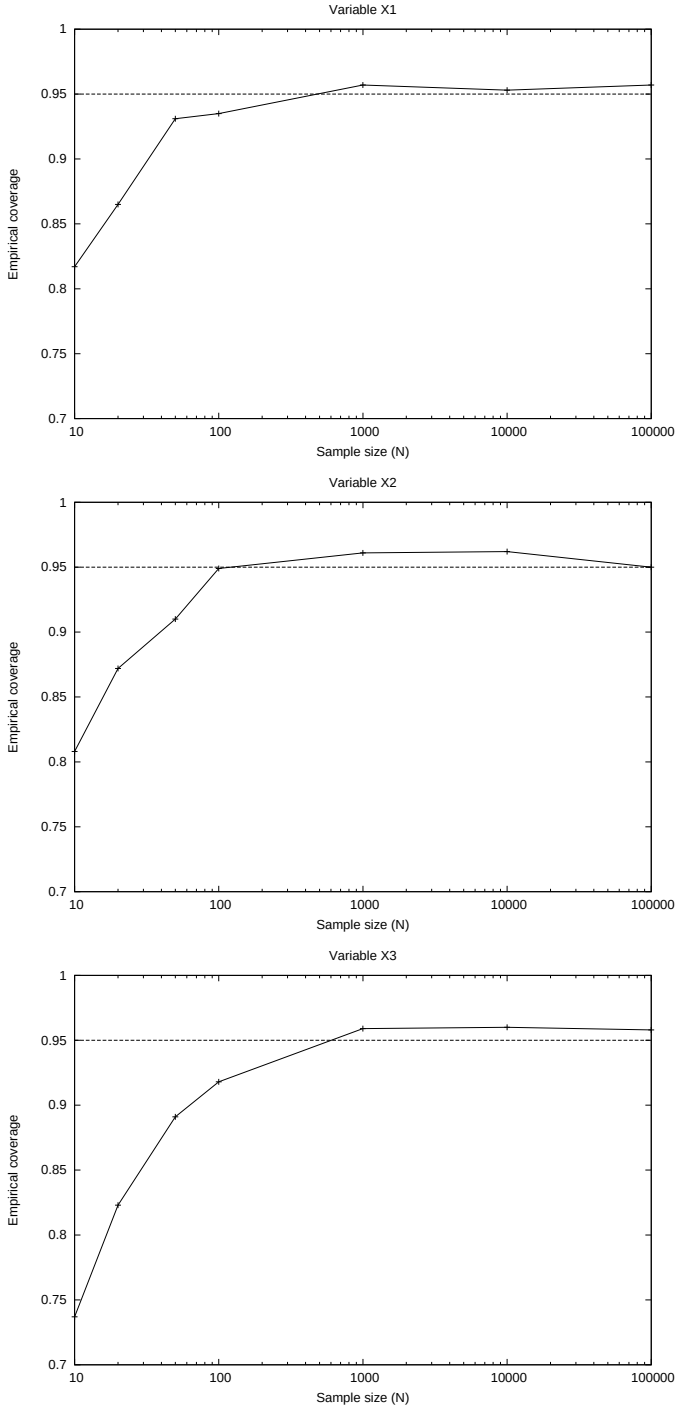


Figure 3.2: Empirical coverages of asymptotic confidence intervals for S^1 (left), S^2 (center) and S^3 (right), as a function of the sample size (for the exact model). The T_N estimator is used.

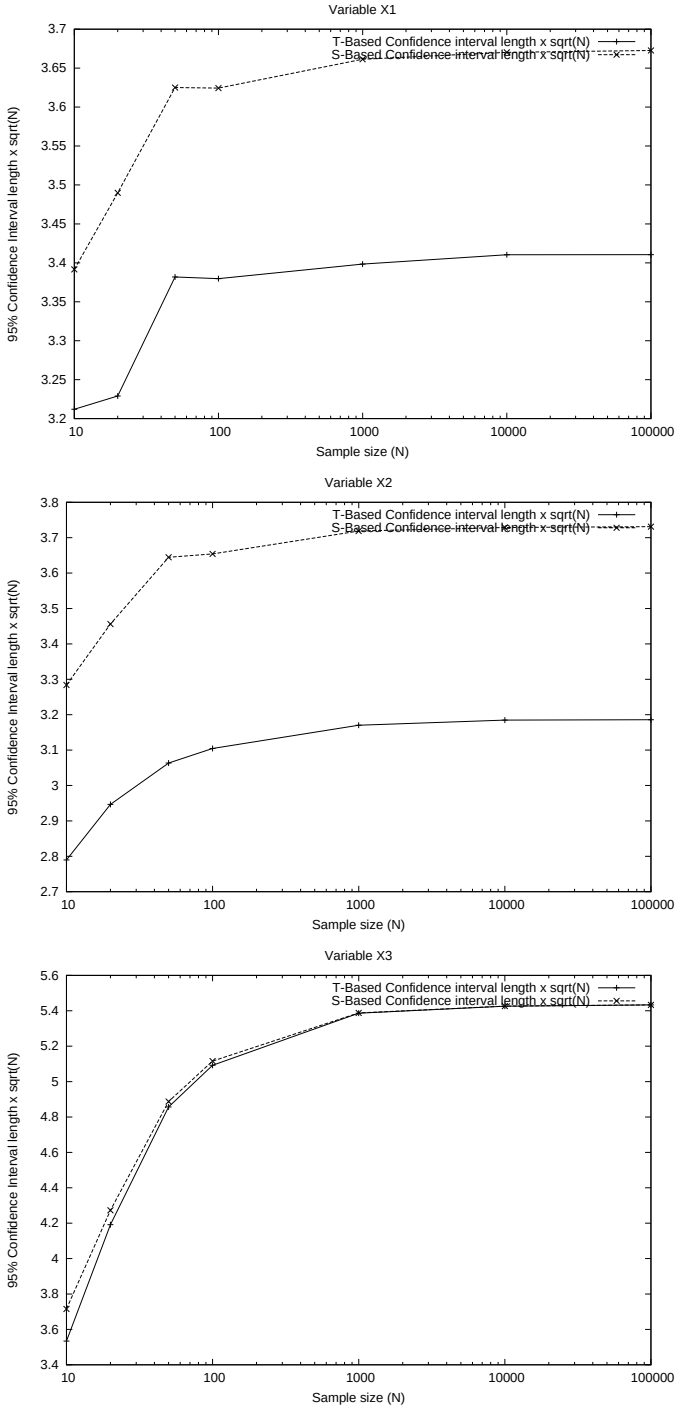


Figure 3.3: Lengths (rescaled by $\sqrt{N}$) of the estimated 95% confidence intervals for S^1 (left), S^2 (center) and S^3 (right), as functions of the sample size (for the exact model). In solid line: length of the interval built from T_N estimator; in dotted line: length of the interval built from S_N estimator.

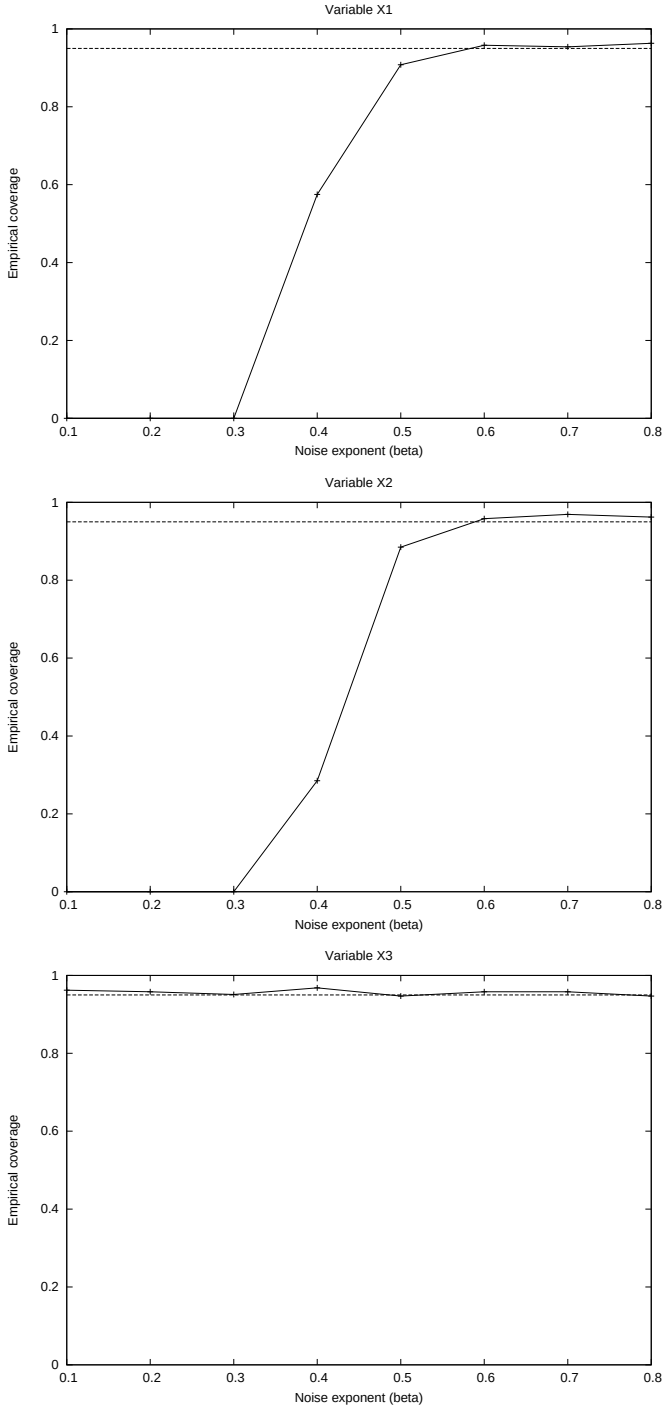


Figure 3.4: Empirical coverages of the asymptotic confidence intervals for S^1, S^2 and S^3, as a function of β (for the Gaussian-perturbated model).

where W is Weibull-distributed with scale parameter $\lambda = 1$ and shape parameter $k = 1/2$. Here, the perturbation depends on the inputs and, as for every input variable, $C_{\delta,N}$ does not converge to zero, Theorem 1 states in particular that $\widetilde{S}_N$ is asymptotically normal for S for $\beta > 1$. Again, this property is suggested for $N = 50000$ by the plot in Figure 3.5.

3.4.4 RKHS metamodel

In this part, we discuss the use of a reproducing kernel Hilbert space (RKHS) interpolator [92, 94, 95] as metamodel $\widetilde{f}$. Such metamodels (also known as Kriging, or Gaussian process metamodels) are widely used when performing sensitivity analysis of time-expensive computer codes [66]. The interpolator depends on a learning sample $\{(d_1, f(d_1)), \ldots, (d_n, f(d_n))\}$, where the design points $\mathcal{D} = \{d_i\}_{i=1,\ldots,n} \subset \mathcal{P}$ are generally chosen according to a space-filling design, for instance the so-called maximin LHS (latin hypercube sampling) designs. Increasing the learning sample size n will increase the necessary number of evaluations of the true model f (each evaluation being potentially very computationally demanding) to build the learning sample, but will also enhance the quality of the interpolation (i.e. reduce metamodel error).
The error analysis of the RKHS method [94, 65] shows that there exist positive constants $\mathcal{C}$ and $\mathcal{K}$, depending on f, so that:

$$\forall u \in \mathcal{P}, \quad \left| f(u) - \widetilde{f}(u) \right| \leq \mathcal{C} e^{-\mathcal{K}/h_{\mathcal{D},\mathcal{P}}}$$

where:

$$h_{\mathcal{D},\mathcal{P}} = \sup_{u \in \mathcal{P}} \min_{d \in \mathcal{D}} \|d - u\|$$

for a given norm $\|\cdot\|$ on $\mathcal{P}$.
The quantity $h_{\mathcal{D},\mathcal{P}}$ can be linked to the number of points $n^*(\varepsilon)$ in an optimal covering of $\mathcal{D}$:

$$n^*(\varepsilon) = \min\{p \in \mathbb{N}^* \mid \exists (d_1, \ldots, d_p) \in \mathcal{P} \text{ s.t.} \\ \forall u \in \mathcal{P}, \exists i \in \{1, \ldots, p\} \text{ satisfying } \|u - d_i\| \leq \varepsilon\}.$$

In other words, $n^*(\varepsilon)$, known as the *covering number of* $\mathcal{P}$, is the smallest size of a design $\mathcal{D}$ satisfying $h_{\mathcal{D},\mathcal{P}} \leq \varepsilon$.
It is known that, when $\mathcal{P}$ is a compact subset of $\mathbb{R}^p$ (in our context, $p = p_1 + p_2$ is the number of input parameters), there exist constants A and B

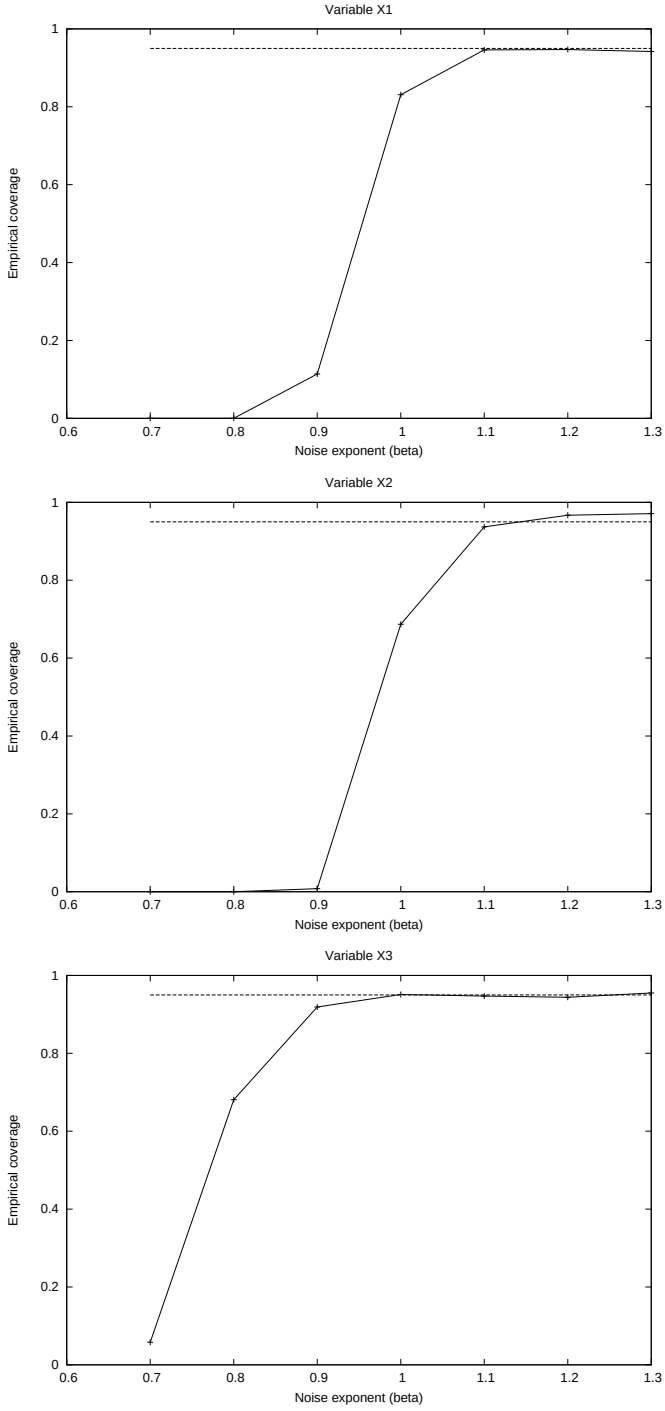


Figure 3.5: Empirical coverages of the asymptotic confidence intervals for S^1, S^2 and S^3, as a function of β (for the Weibull-perturbated model).

so that:

$$A\varepsilon^{-p} \leq n^*(\varepsilon) \leq B\varepsilon^{-p}.$$

Hence, assuming that an optimal design of size n is chosen, we have, for a constant B':

$$h_{\mathcal{D},\mathcal{P}} \leq B' n^{-1/p}$$

and we have the following pointwise metamodel error bound, for constants C and K':

$$\forall u \in \mathcal{P}, \quad \left| f(u) - \widetilde{f}(u) \right| \leq \mathcal{C} e^{-\mathcal{K}' n^{1/p}}$$

which obviously leads to an integrated error bound on the variance of the metamodel error:

$$\mathrm{Var}\delta \leq C e^{-k n^{1/p}}$$

for suitable constants C and k.

Numerical illustration

We illustrate the properties of the RKHS-based sensitivity analysis using the Ishigami function (3.4.1) as true model, maximin LHSes for design points selection. RKHS interpolation also depends on the choice of a kernel, which we choose Gaussian all the way through. All simulations have been made with the `R` software [82], together with the `lhs` package [17] for design sampling and the `mlegp` package [27] for Kriging.

Figure 3.6, which shows an estimation (based on a sample of 1000 metamodel errors) of the (logarithm of) variance of metamodel error, plotted against the cubed root of the learning sample size $n^{1/3}$. Using an exponential regression, we find that:

$$\mathrm{Var}(\delta) \approx \widehat{C} e^{-\widehat{k} n^{1/3}} \tag{3.4.2}$$

where:

$$\widehat{k} = 1.91$$

Now, if we let the learning sample size n depend on the Monte-Carlo sample size N by the relation:

$$n = (a \ln N)^3$$

for $a > 0$, Theorem 1 suggests that the metamodel-based estimators of the sensitivity indices are asymptotically normal if and only if $N^{-a\widehat{k}+1} \to 0$ when $N \to +\infty$, that is $a > \dfrac{1}{\widehat{k}}$, or

$$a > 0.52, \tag{3.4.3}$$

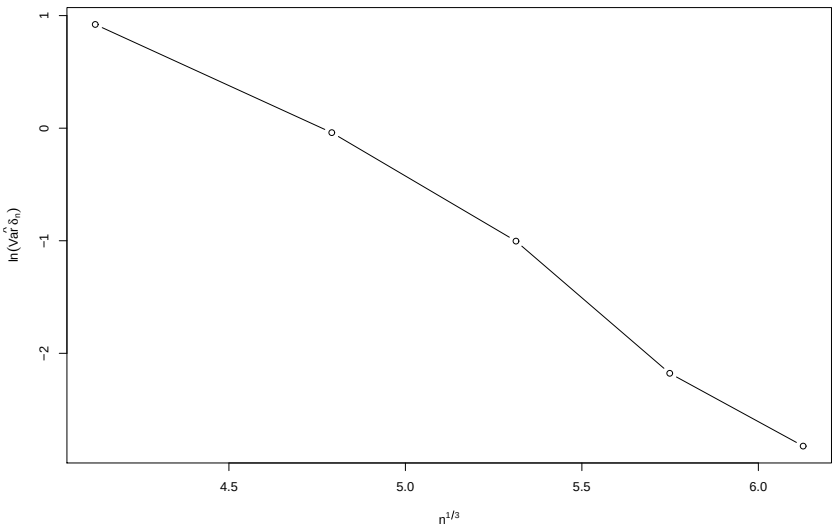


Figure 3.6: Estimation of the Kriging metamodel error variance (log. scale) as function of the learning sample size n.

according to our numerical value for $\widehat{k}$.

Even if it has not been rigorously proved that this condition is necessary and sufficient (due to the estimation of k and the fact that (3.4.2) provably holds, possibly with different constants, as an upper bound), one should observe in practice that the behavior of the empirical confidence intervals for large values of N changes as this critical value of a is crossed. Table 3.1 below shows the results obtained for different subcritical and supercritical values of a (i.e., (3.4.3) does not hold, or hold, respectively), and provides a clear illustration of this fact.

3.4.5 Nonparametric regression

In this section, we consider the case where the true model f is not directly observable, but is only available through a finite set of *noisy* realisations of:

$$f_{\text{noisy}}(D_i) = f(D_i) + \varepsilon_i, \quad i = 1, \ldots, n$$

where $\mathcal{D} = (D_i = (X_i, Z_i))_{i=1,\ldots,n}$ are independent copies of (X, Z), and $\{\varepsilon_i\}_{i=1,\ldots,n}$ are independent, identically distributed centered random variables.

a	N	n	Coverage for S^1	Cov. for S^2	Cov. for S^3
.4	3000	33	0.1	0	0.7
.4	4000	37	0.08	0	0.78
.4	6000	43	0.26	0.3	0.88
.4	10000	51	0.28	0.18	0.78
.4	20000	77	0.28	0.1	0.59
.6	3000	111	0.79	0.37	0.9
.6	4000	124	0.8	0.7	0.94
.6	10000	169	0.92	0.82	0.94
.6	20000	210	0.93	0.85	0.95
.7	3000	177	0.93	0.88	0.93
.7	4000	196	0.9	0.91	0.94
.7	6000	226	0.94	0.93	0.97
.8	4000	293	0.95	0.95	0.95

Table 3.1: Estimation of the asymptotic coverages for the RKHS Ishigami metamodel. Empirical coverages are obtained using 100 confidence interval replicates. Theoretical coverage is 0.95.

As discussed in Section 3.3.2, one should expect that the Sobol index estimator computed on f_{noisy} are not asymptotically normal for the estimation of the Sobol indices of f (as $\text{Var}(\varepsilon_i)$ is fixed). This motivates the use of a smoothed estimate of f, which we will take as our perturbated model $\widetilde{f} = \widetilde{f}_{\mathcal{D}}$. We consider the Nadaraya-Watson estimator:

$$\widetilde{f}_{\mathcal{D}}(u) = \begin{cases} \dfrac{\sum_{i=1}^n K_h\left(u - D_i\right) f_{\text{noisy}}(D_i)}{\sum_{i=1}^n K_h(u - D_i)} \text{ if } \displaystyle\sum_{i=1}^n K_h(u - D_i) \neq 0 \\ 0 \text{ else.} \end{cases}$$

where K_h is a smoothing kernel of window $h \in \mathbb{R}^p$; for instance K_h is a Gaussian kernel:

$$K_h(v) = \exp\left(-\sum_{i=1}^p \frac{\|v_i\|^2}{h_i^2}\right) \tag{3.4.4}$$

where the norm $\|\cdot\|$ is the Euclidean norm on $\mathbb{R}^p$.

It is known that, under regularity conditions on f, and a n-dependent appropriate choice of h, the mean integrated square error (MISE) of $\widetilde{f}$ satisfies:

$$\int \mathbb{E}_{\mathcal{D}}\left(\left(f(u) - \widetilde{f}_{\mathcal{D}}(u)\right)^2\right) \mathrm{d}u \leq C' n^{-\gamma}, \tag{3.4.5}$$

for a positive constant C' and a positive γ (which depends only on the dimension p and the regularity of f), and where $\mathbb{E}_{\mathcal{D}}$ denotes expectation with respect to the random "design" $\mathcal{D}$.

Now, by Fubini-Tonelli's theorem, we have:

$$\int \mathbb{E}_{\mathcal{D}}\left(\left(f(u) - \widetilde{f}_{\mathcal{D}}(u)\right)^2\right) \mathrm{d}u = \mathbb{E}_{\mathcal{D}}\left(\int \left(f(u) - \widetilde{f}_{\mathcal{D}}(u)\right)^2 \mathrm{d}u\right). \tag{3.4.6}$$

By using (3.4.6), (3.4.5) and applying Markov's inequality to the positive random variable $\int \left(f(u) - \widetilde{f}_{\mathcal{D}}(u)\right)^2 \mathrm{d}u$, we have that, for any $\varepsilon > 0$,

$$\mathbb{P}\left(\left\{\mathcal{D} \,/\, \int \left(f(u) - \widetilde{f}_{\mathcal{D}}(u)\right)^2 \mathrm{d}u \leq \frac{C'}{\varepsilon} n^{-\gamma}\right\}\right) \geq 1 - \varepsilon.$$

Hence, for a fixed risk $\varepsilon > 0$, there exist $C > 0$ and $\gamma > 0$ so that:

$$\int \left(\widetilde{f}_{\mathcal{D}}(u) - f(u)\right)^2 \mathrm{d}u \leq C n^{-\gamma} \tag{3.4.7}$$

holds with probability greater than $1 - \varepsilon$ (with respect to the choice of $\mathcal{D}$).

We recall that the quantity we have to consider in order to study asymptotic normality of Sobol index estimator on the metamodel is:

$$\text{Var}(\delta) = \int \left(f(u) - \widetilde{f}_{\mathcal{D}}(u)\right)^2 \mathrm{d}u - \left(\int \left(f(u) - \widetilde{f}_{\mathcal{D}}(u)\right) \mathrm{d}u\right)^2$$

and that, obviously,

$$\mathrm{Var}(\delta) \leq \int \left(f(u) - \widetilde{f}_{\mathcal{D}}(u)\right)^2 \mathrm{d}u.$$

This gives, by making use of (3.4.7):

$$\mathrm{Var}\left(\delta\right) \leq Cn^{-\gamma} \tag{3.4.8}$$

with probability greater than $1-\varepsilon$.
In most cases of application, the design $\mathcal{D}$ is fixed. In view of (3.4.8), it is reasonable to suppose that there exist $C > 0$ and $\beta > 0$ so that:

$$\mathrm{Var}\left(\delta\right) \leq Cn^{-\beta}$$

and we make n depend on N by the following relation:

$$n = N^a,$$

for $a > 0$. By Theorem 1, the estimator sequence $\{\widetilde{S}_N\}$ is asymptotically normal provided that $N\mathrm{Var}(\delta_N) \to 0$, that is: $a > \dfrac{1}{\beta}$.

Numerical illustration

We now illustrate this property using the Ishigami function (3.4.1) as true model, and a Gaussian white noise ε_i of standard deviation 0.3 (yielding to a signal-to-noise ratio of 90%).
The nonparametric regressions are carried using a Gaussian kernel (3.4.4), the R package `np` [47], together with the extrapolation method of [83] for window selection and the `FIGtree` [69] C++ library for efficient Nadaraya-Watson evaluation based on fast gaussian transform.
Figure 3.7, which shows an estimation (based on a test sample of size 3000) of $\mathrm{Var}(\delta)$ in function of n, and a power regression shows that:

$$\mathrm{Var}\left(\delta\right) \approx Cn^{-\widehat{\beta}}$$

with $\widehat{\beta} = 0.86$. This gives an estimate of 1.16 as the critical a for asymptotic normality.
As in the RKHS case, we performed estimations of the coverages of the asymptotic confidence interval for several values of a and N; the results are gathered in Table 3.2. We see that, first, the condition $a > 1.16$ implies correct coverages, and, second, the condition also seems to be near-necessary

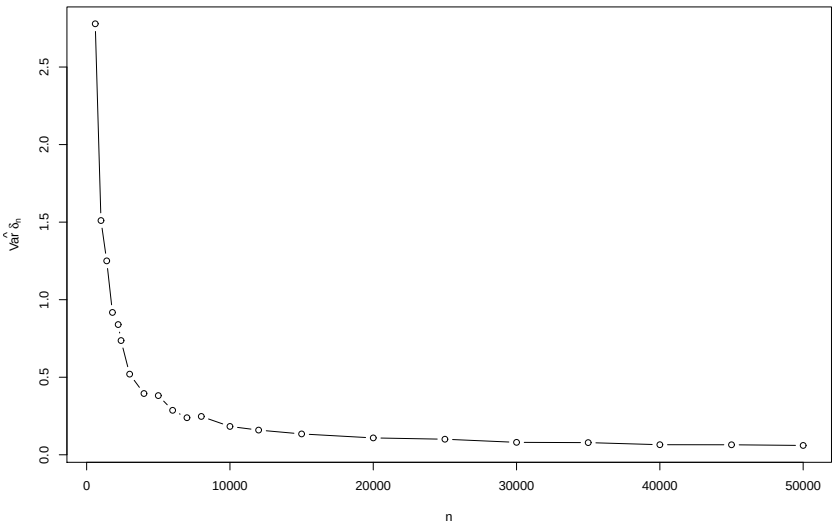


Figure 3.7: Estimation of the nonparametric regression error variance (log. scale) as function of the learning sample size n (Subsection 3.4.5).

a	N	n	Coverage for S^1	Cov. for S^2	Cov. for S^3
0.8	1000	252	0.25	0.01	0.94
0.8	2000	438	0.05	0.02	0.86
1.1	1000	1996	0.95	0.97	0.96
1.1	2000	4277	0.95	0.93	0.96
1.2	1000	3982	0.93	0.95	0.96
1.2	2000	9147	0.96	0.97	0.95
1.3	1000	7944	0.95	0.99	0.94
1.3	2000	19559	0.95	0.95	0.96

Table 3.2: Estimation of the asymptotic coverages for the Ishigami nonparametric regression. Empirical coverages are obtained using 100 confidence interval replicates. Theoretical coverage is 0.95.

to have asymptotic normality. We also remark that, for the asymptotic normality to hold, the necessary number of noisy model evaluations is asymptotically comparable to the Monte-Carlo sample size (while, in the RKHS case, the necessary number of true model evaluations was asymptotically negligible with respect to the Monte-Carlo sample size): this shows that the nonparametric regression is suitable in the case of noisy but abundant model evaluations, while RKHS interpolation is clearly preferable when the true model output is costly to evaluate (i.e. few model outputs are available).

Acknowledgements: This work has been partially supported by the French National Research Agency (ANR) through COSINUS program (project COSTA-BRAVA n°ANR-09-COSI-015).

Chapter 4

Certified reduced-basis solutions of viscous Burgers equation parametrized by initial and boundary values

Résumé: Nous présentons une procédure base réduite offline/online pour le problème aux limites donné par l'équation de Burgers visqueuse. Cette procédure permet le calcul rapide approché de solutions à ce problème, paramétrées par la viscosité et les données initiales et aux limites. Nous donnons également une borne d'erreur, nécessitant peu de ressources de calcul, qui certifie l'approximation proposée. Nous présentons des expérimentations numériques montrant que notre méthode permet une économie significative en temps de calcul, et que la borne d'erreur proposée est efficace.

Abstract: We present a reduced basis offline/online procedure for viscous Burgers initial boundary value problem, enabling efficient approximate computation of the solutions of this equation for parametrized viscosity and initial and boundary value data. This procedure comes with a fast-evaluated rigorous error bound certifying the approximation procedure. Our numerical experiments show significant computational savings, as well as efficiency of the error bound.

Introduction

This paper is set in the context of sensitivity analysis and uncertainty analysis in geophysical models. Such models typically involve a wide range of parameters, such as: source terms (climatic forcings, heat/wind/matter fluxes), boundary conditions (forcings, open boundaries), and the initial state of the system.

Their study generally leads to parametrized partial differential equations (PDEs). These equations often involve poorly-known parameters. Therefore, it is important to be able to measure the impact of a given parameter on the quality of the solution, and also to identify the "sensitive" parameters, that is, the parameters for which a small variation implies a large variation of the model solution. Due to their ability to perform global sensitivity analyses for nonlinear models, stochastic tools [48, 88] are rapidly expanding. These methods require "many queries," that is solving the parametrized PDE for a large (say, thousands) number of values of the parameters. When analytic solution to the PDE is not known (as it is often the case), one has to use a numerical method (such as finite difference or finite element) to compute an approximate value of the solution. Such methods lead to computer codes that could take a large time to produce an accurate-enough approximation — for a single value of the parameter. Having the "many-query" problem in mind, it is crucial to design a procedure that solves the equation for several values of the parameter faster than the naïve approach of calling the numerical code for each required instance of the parameter.

The reduced basis (RB) method is such a procedure; we split the overall computation into two successive parts: one part, the *offline* phase, makes use of the standard, computationally intensive numerical procedure used to solve the PDE to gather "knowledge" about solutions of the latter; and the other one, the *online* phase, where we rely on data collected during the offline phase to compute, for each desired instance of the parameter, a good approximation of the solution, for a per-instance cost that is orders of magnitude smaller than the cost of one run of the standard numerical code. The advantage is that, for a sufficiently large number of online evaluations, the fixed cost of the offline phase will be strongly dominated by the reduction in the marginal cost provided by the online procedure. This cost reduction is made possible by the fact that, in most cases, the desired solutions of the

PDE, for all the considered values of the parameter, lie in some manifold of functions that is "close" to a low-dimensional linear subspace. One goal of the offline phase is to find such a suitable subspace, so that the online procedure can look for the solution of the PDE as an element of the subspace — so as to reduce the number of degrees of freedom and thus the computational cost. One interesting feature of the RB approach is that it comes with an *online error bound*, that is a (provably) certified, natural norm, fast-computed (i.e. almost of the same complexity of the online phase) upper bound of the distance between the solution provided by the online phase (called the *reduced*, or *online*) solution and the one given by the standard, expensive numerical procedure (called the *full* or *reference* solution). This "certified RB" framework has been developed for *affinely* parametrized second-order elliptic linear PDEs in [72]. It has been extended to nonlinear, non-affinely parametrized, parabolic PDEs, see e.g. [45], [44] and applied to problems such as steady incompressible Navier-Stokes [110]. Moreover theoretical work has been done to ensure *a priori* convergence of the RB procedure [12].

In this paper, we are interested in the RB reduction of the time-dependent viscous Burgers equation (which will serve as a "test case" for the "real" equations modelling geophysical fluids we are interested in). Papers [46] and [84] extend certified RB methodology to linear initial-boundary value problems. The case of homogeneous Dirichlet boundary conditions, zero initial value and fixed (*i.e.*, not parametrized) source term has been treated in [111], [71] ; in these works, the only parameter was the viscosity coefficient. Parametrization of initial and Dirichlet boundary conditions (treated using a conversion to a homogeneous Dirichlet problem) has been done in [59], for a general multidimensional quadratically nonlinear equation. Our methodology allows parametrization of the viscosity, the initial state, the source term and and of the boundary conditions. Compared to the works cited above, we use a weak (penalization) treatment of the Dirichlet boundary conditions. We will see that this weak treatment is more favorable in terms of both computation and storage complexities. Besides, our paper features a new error bound, which has shown to be, in all the testcases we performed, much more efficient than the existing bound. In particular, we will show that our bound exhibits improved sharpness for low viscosities, where [71] pointed out the moderate efficiency of the presented *a posteriori* bound.

This paper is organised as follows: in the first part, we introduce the viscous Burgers equation, and present a standard numerical procedure used to solve it; in the second part, we expose our offline/online reduction procedure; in the third part, we develop a certified online error bound; finally in the fourth part we validate and discuss our results based on numerical experiments.

4.1 Model

In this section, we describe the model we are interested in. Subsection 4.1.1 introduces the viscous Burgers equation, while Subsection 4.1.2 presents the "full" numerical procedure on which our reduction procedure, described in Section 4.2, relies on.

4.1.1 Equation

We are interested in u, function of space $x \in [0;1]$ and time $t \in [0;T]$ (for $T > 0$), with regularity:
$u \in C^1\left([0,T], H^1(]0,1[)\right)$, satisfying the *viscous Burgers equation*:

$$\frac{\partial u}{\partial t} + \frac{1}{2}\frac{\partial}{\partial x}(u^2) - \nu\frac{\partial^2 u}{\partial x^2} = f \tag{4.1.1}$$

where $\nu \in \mathbb{R}_*^+$ ($\mathbb{R}$ denotes the set of real numbers, $\mathbb{R}_*^+$ the set of positive real numbers) is the *viscosity*, and $f \in C^0\left([0,T], L^2(]0,1[)\right)$ is the *source term*. For u to be well-defined, we also prescribe initial values $u_0 \in H^1(]0,1[)$:

$$u(t=0,x) = u_0(x) \quad \forall x \in [0;1] \tag{4.1.2}$$

and boundary values $b_0, b_1 \in C^0([0,T])$:

$$\begin{cases} u(t, x=0) = b_0(t) \\ u(t, x=1) = b_1(t) \end{cases} \qquad \forall t \in [0;T] \tag{4.1.3}$$

Where b_0, b_1 and u_0 are given functions, supposed to satisfy *compatibility conditions*:

$$u_0(0) = b_0(0) \quad \text{and} \quad u_0(1) = b_1(0) \tag{4.1.4}$$

This problem can be analyzed by means of the Cole-Hopf substitution (see [50] for instance), which turns (4.1.1) into the heat equation, leading to an integral representation of u.

4.1.2 Numerical resolution

We now describe the "expensive" numerical resolution of the problem described above that will serve as our reference for the reduction procedure described in the next section. We proceed in two steps: space discretization in paragraph 4.1.2 and time discretization in paragraph 4.1.2.

Space discretization

For space discretization, we use a $\mathbf{P}^1$ finite element procedure with weak (penalty) setting of the Dirichlet boundary conditions (4.1.3).
We first have to write the weak formulation of our PDE ; to do so, we multiply (4.1.1) by a function $v \in H^1(]0;1[)$ and integrate over $]0;1[$:

$$\int_0^1 \frac{\partial u}{\partial t}(t,x)v(x)\mathrm{d}x + \frac{1}{2}\int_0^1 \frac{\partial (u^2)}{\partial x}(t,x)v(x)\mathrm{d}x \\ -\nu\int_0^1 \frac{\partial^2 u}{\partial x^2}(t,x)v(x)\mathrm{d}x = \int_0^1 f(t,x)v(x)\mathrm{d}x \quad \forall v \in H^1(]0;1[)\ \forall t \in [0;T] \tag{4.1.5}$$

Next, we integrate by parts the second and the third integral appearing in the left hand side of the previous equation:

$$\int_0^1 \frac{\partial (u^2)}{\partial x}(t,x)v(x)\mathrm{d}x = -\int_0^1 u^2(t,x)\frac{\partial v}{\partial x}(x)\mathrm{d}x + \Big[u^2(t,\cdot)v\Big]_0^1$$
$$\int_0^1 \frac{\partial^2 u}{\partial x^2}(t,x)v(x)\mathrm{d}x = -\int_0^1 \frac{\partial u}{\partial x}(t,x)\frac{\partial v}{\partial x}(x)\mathrm{d}x + \left[\frac{\partial u}{\partial x}(t,\cdot)v\right]_0^1$$

Inserting this into (4.1.5), we get:

$$\int_0^1 \frac{\partial u}{\partial t}(t,x)v(x)\mathrm{d}x - \frac{1}{2}\int_0^1 u^2(t,x)\frac{\partial v}{\partial x}(x)\mathrm{d}x + \nu\int_0^1 \frac{\partial u}{\partial x}(t,x)\frac{\partial v}{\partial x}(x)\mathrm{d}x \\ +\frac{1}{2}\Big[u^2(t,\cdot)v\Big]_0^1 - \nu\left[\frac{\partial u}{\partial x}(t,\cdot)v\right]_0^1 = \int_0^1 f(t,x)v(x)\mathrm{d}x \quad \forall v \in H^1(]0;1[)\ \forall t \in [0;T] \tag{4.1.6}$$

To get rid of the two boundary terms arising in the integrations by parts, one usually restricts v to satisfy $v(0) = v(1) = 0$ so as to make the boundary terms disappear; the Dirichlet boundary conditions (4.1.3) are then incorporated "outside" of the weak formulation. However, the reduction framework we are to expose later requires the boundary conditions to be ensured by the weak formulation itself. The Dirichlet penalty method, presented in

[4], is a way of doing so, at the expense of a slight approximation error. This method entails replacement of boundary conditions (4.1.3) with the following conditions:

$$\begin{cases} -\frac{1}{2}u^2(t,x=0) + \nu\frac{\partial u}{\partial x}(t,x=0) = P\left(u(t,x=0) - b_0(t)\right) \\ \frac{1}{2}u^2(t,x=1) - \nu\frac{\partial u}{\partial x}(t,x=1) = P\left(u(t,x=1) - b_1(t)\right) \end{cases} \quad \forall t \in [0;T] \tag{4.1.7}$$

with a fixed *penalization constant* $P > 0$.

The intuitive idea underlying (4.1.7) is that it can clearly be rewritten as:

$$\begin{cases} u(t,x=0) = b_0(t) + \frac{1}{P}\left(-\frac{1}{2}u^2(t,x=0) + \nu\frac{\partial u}{\partial x}(t,x=0)\right) \\ u(t,x=1) = b_1(t) + \frac{1}{P}\left(\frac{1}{2}u^2(t,x=1) - \nu\frac{\partial u}{\partial x}(t,x=1)\right) \end{cases} \quad \forall t \in [0;T]$$

so that (4.1.3) is asymptotically verified for $P \to +\infty$. The reader can refer to [6] for rigorous *a priori* error estimates when using Dirichlet penalty in the linear elliptic case.

In practice, we can check if our approximation is sufficiently accurate by means of the following *a posteriori* procedure: we take for P some large value (typically $P = 10^7$), we compute (numerically) an approximate solution u_d, using the procedure we are currently describing, and we check if an indicator of the amount of failure in verification of (4.1.3) is small enough; such an indicator can be, for instance:

$$\varepsilon_b = \sup_t \left[\max\left(|u_d(t,x=0) - b_0(t)|, |u_d(t,x=1) - b_1(t)|\right)\right] \tag{4.1.8}$$

where the supremum is taken over all discrete time steps. If this indicator is larger than a prescribed tolerance, then P has to be increased. Our numerical results in Section 4.4 will assert this condition. We can then invoke the well-posedness of the boundary/initial value problem (specifically, continuous dependence on the boundary values) to ensure that the solution of (4.1.1), (4.1.2) and (4.1.7) will be close to the solution of (4.1.1), (4.1.2) and (4.1.3). This reasoning is analogous to the one made when omitting the approximation made when replacing exact boundary values by their discretized counterparts.

Going back to our weak formulation, we multiply the first line of (4.1.7) by

$v(0)$, the second one by $v(1)$ and add up these two equations. We get that:

$$\frac{1}{2}\left[u^2(t,\cdot)v\right]_0^1 - \nu\left[\frac{\partial u}{\partial x}(t,\cdot)v\right]_0^1 = P\left[(u(t,x=0)v(0) - b_0(t)v(0)) + (u(t,x=1)v(1) - b_1(t)v(1))\right]$$

Putting it back into (4.1.6) and isolating the terms not involving u on the right-hand side yields the following weak formulation:

$$\int_0^1 \frac{\partial u}{\partial t}v - \frac{1}{2}\int_0^1 u^2\frac{\partial v}{\partial x} + \nu\int_0^1 \frac{\partial u}{\partial x}\frac{\partial v}{\partial x} + P\left(u(t,x=0)v(0) + u(t,x=1)v(1)\right) = \int_0^1 f(t,\cdot)v + P\left(b_0(t)v(0) + b_1(t)v(1)\right) \quad \forall v \in H^1(]0;1[)\ \forall t \in [0;T] \tag{4.1.9}$$

that is:

$$\left\langle \frac{\partial u}{\partial t}(t,\cdot), v\right\rangle + c(u(t,\cdot),u(t,\cdot),v) + \nu a(u(t,\cdot),v) + B(u(t,\cdot),v) = \ell(v,t) + b_0(t)\beta_0(v) + b_1(t)\beta_1(v) \quad \forall v \in H^1(]0;1[), \forall t \in [0,T] \tag{4.1.10}$$

by introducing the following notations (for all $v, w, z \in H^1(]0;1[)$, $t \in [0;T]$):

$$\langle w, v\rangle = \int_0^1 wv \qquad a(w,v) = \int_0^1 \frac{\partial w}{\partial x}\frac{\partial v}{\partial x}$$
$$B(w,v) = P\left(w(0)v(0) + w(1)v(1)\right) \qquad \ell(v,t) = \int_0^1 f(t,\cdot)v$$
$$\beta_0(v) = Pv(0) \qquad \beta_1(v) = Pv(1)$$

and:

$$c(w,v,z) = -\frac{1}{2}\int_0^1 wv\frac{\partial z}{\partial x}$$

for every w, v and z in $H^1(]0;1[)$ for which $wv\dfrac{\partial z}{\partial x} \in L^1(]0;1[)$.
The weak formulation is then discretized with Lagrange $\mathbf{P^1}$ finite elements (see [80]) by choosing some integer $\mathcal{N}$ and considering a uniform subdivision of $[0;1]$ with $\mathcal{N}+1$ nodes: $\{x_i\}_{i=0,\ldots,\mathcal{N}}$ and, for each $i = 0,\ldots,\mathcal{N}$ we denote by ϕ_i the piecewise-affine "hat" function whose value is 1 on x_i and 0 on every other nodes.
We denote by X the linear subspace of $H^1(]0;1[)$ spanned by $\{\phi_i\}_{i=0,\ldots,\mathcal{N}}$. We also set $\|\cdot\|$ to be the $L^2(]0;1[)$ norm.

Every $\psi \in X$ can be written as $\psi = \sum_{j=0}^{\mathcal{N}} \psi_j \phi_j$, and we have $\psi(x_i) = \psi_i$ for all $i = 0, \dots, \mathcal{N}$. This justifies that π defined below is a projection of $H^1(]0;1[)$ onto X:

$$\pi : \begin{cases} H^1(]0;1[) \to X \\ \qquad \psi \mapsto \sum_{j=0}^{\mathcal{N}} \psi(x_j)\phi_j \end{cases}$$

The space discretization of our problem is the following: for all $t \in [0;T]$, find $u(t,\cdot) \in X$ so that :

$$\begin{cases} u(t=0,\cdot) = \pi(u_0) \\ \left\langle \dfrac{\partial u}{\partial t}(t,\cdot), v \right\rangle + c(u(t,\cdot), u(t,\cdot), v) + \nu a(u(t,\cdot), v) + B(u(t,\cdot), v) \\ \qquad\qquad = \ell_\pi(v,t) + b_0(t)\beta_0(v) + b_1(t)\beta_1(v) \quad \forall v \in X \end{cases} \tag{4.1.11}$$

where

$$\ell_\pi(v,t) = \int_0^1 \pi\left(f(t,\cdot)\right) v$$

Note that u now stands for a discrete solution, while it was used to designate an analytic solution before.

Time discretization

We now discretize (4.1.11) in time using the backward Euler scheme: we choose a timestep $\Delta t > 0$ and consider an uniform subdivision of $[0;T]$: $\{t_k = k\Delta t\}_{k=0,\dots,\mathcal{T}}$ where $\mathcal{T} = \dfrac{T}{\Delta t}$.
Our fully discrete problem is: for $k = 0, \dots, \mathcal{T}$, find $u^k \in X$, approximation of $u(t_k,\cdot)$, satisfying:

$$u^0 = \pi(u_0) \tag{4.1.12a}$$

and:

$$\begin{aligned} &\left\langle \frac{u^k - u^{k-1}}{\Delta t}, v \right\rangle + c(u^k, u^k, v) + \nu a(u^k, v) + B(u^k, v) \\ &\qquad = \ell_\pi(v, t_k) + b_0(t_k)\beta_0(v) + b_1(t_k)\beta_1(v) \quad \forall v \in X \ \forall k = 1, \dots, \mathcal{T} \end{aligned} \tag{4.1.12b}$$

We sequentially compute $\{u^k\}_{k=0,\dots,\mathcal{T}}$ in the following way: u^0 comes straightforwardly from (4.1.12a), and for $k = 1, \dots, \mathcal{T}$, u^k depends on u^{k-1} through

(4.1.12b), which can be rewritten:

$$\frac{1}{\Delta t}\langle u^k, v\rangle + c(u^k, u^k, v) + \nu a(u^k, v) + B(u^k, v)$$
$$= \frac{1}{\Delta t}\langle u^{k-1}, v\rangle + \ell_\pi(v, t_k) + b_0(t_k)\beta_0(v) + b_1(t_k)\beta_1(v) \quad \forall v \in X \quad (4.1.13)$$

We can now expand our unknown $u^k \in X$ on the $\{\phi_j\}_j$ basis: $u^k = \sum_{j=1}^{\mathcal{N}} u_j^k \phi_j$, and the vector $\left(u_j^k\right)_j$ becomes our new unknown.
Moreover, it is sufficient for (linear-in-v) relation (4.1.13) to be satisfied for all v in a basis of X, namely for $v = \phi_i$, $\forall i = 0, \ldots, \mathcal{N}$.
So (4.1.13) can be rewritten as a nonlinear (due to the nonlinearity in $c(u^k, u^k, v)$) system of $\mathcal{N}+1$ equations (one for each instantation $v = \phi_i$) involving $\left(u_j^k\right)_{j=0,\ldots,\mathcal{N}}$. This nonlinear system is solved using Newton iterations: starting with an initial guess $\overline{u^k}$, one looks for $\delta = \sum_{j=1}^{\mathcal{N}} \delta_j \phi_j$ so that $u^k = \overline{u^k} + \delta$ satisfies the linearization near $\delta = 0$ of (4.1.13) for $v = \phi_i$, $i = 0, \ldots, \mathcal{N}$, that is to say:

$$\frac{1}{\Delta t}\langle \overline{u^k} + \delta, \phi_i\rangle + c(\overline{u^k}, \overline{u^k}, \phi_i) + 2c(\overline{u^k}, \delta, \phi_i) + \nu a(\overline{u^k} + \delta, \phi_i) + B(\overline{u^k} + \delta, \phi_i)$$
$$= \frac{1}{\Delta t}\langle u^{k-1}, \phi_i\rangle + \ell_\pi(\phi_i, t_k) + b_0(t_k)\beta_0(\phi_i) + b_1(t_k)\beta_1(\phi_i) \quad \forall i = 0, \ldots, \mathcal{N} \quad (4.1.14)$$

System (4.1.14) is a $(\mathcal{N}+1) \times (\mathcal{N}+1)$ linear system involving $(\delta_j)_{j=0,\mathcal{N}}$. Once solved for δ, one can test for convergence of the Newton iteration: if the norm of δ is smaller than a prescribed precision, then we stop here, produce u^k and get to the next time step $k+1$; otherwise we do one more Newton step, this time using u^k as initial guess $\overline{u^k}$.
We can note that the linear system to be solved at each Newton step is not symmetric but is sparse. Due to this sparsity, its solution (using an iterative method such as BICGSTAB or GMRES) takes about $O\left(\mathcal{N}^2\right)$ operations in the worst case. One can take advantage of the tridiagonal structure of the matrix (which is present in the one-dimensional case, since ϕ_i and ϕ_j have no common support if $|j-i| > 1$) and use Thomas' algorithm [101] to solve the system with $O(\mathcal{N})$ operations.

4.2 Reduction procedure

In this section, we show the offline/online procedure announced in the introduction to produce reduced basis solutions of the problem formed by (4.1.1), (4.1.2) and (4.1.3), based on the "full basis" numerical method presented above. We begin by a description of our parameters in Subsection 4.2.1. Our offline/online reduction procedure is described subsequently, in Subsection 4.2.2.

4.2.1 Parameters

We parametrize u_0, b_0, b_1 and f as:

$$b_0(t) = b_{0m} + \sum_{l=1}^{n(b_0)} A_l^{b_0} \Phi_l^{b_0}(t) \qquad b_1(t) = b_{1m} + \sum_{l=1}^{n(b_1)} A_l^{b_1} \Phi_l^{b_1}(t)$$

$$f(t,x) = f_m + \sum_{l=1}^{n_T(f)} \sum_{p=1}^{n_S(f)} A_{lp}^f \Phi_l^{fT}(t) \Phi_p^{fS}(x) \qquad u_0(x) = u_{0m} + \sum_{l=1}^{n(u_0)} A_l^{u_0} \Phi_l^{u_0}(x)$$

The functions involved in these linear combinations Φ^{b_0}, $\Phi_l^{b_1}$, Φ_l^{fT}, Φ_p^{fS} and $\Phi_l^{u_0}$ as well as the number of terms in the decompositions $n(b_0)$, $n(b_1)$, $n_T(f)$, $n_S(f)$ and $n(u_0)$ are fixed, while our parameters, namely: viscosity ν, coefficients b_{0m}, b_{1m}, f_m and u_{0m}, and $\left(A_l^{b_0}\right)_{l=1,\dots,n(b_0)}$, $\left(A_l^{b_1}\right)_{l=1,\dots,n(b_1)}$, $\left(A_{lp}^f\right)_{l=1,\dots,n_T(f);p=1,\dots,n_S(f)}$ and $(A_l^{u_0})_{l=1,\dots,n(u_0)}$ live in some Cartesian product of intervals $\mathcal{P}'$, subset of $\mathbb{R}^{1+4+n(b_0)+n(b_1)+n_T(f)n_S(f)+n(u_0)}$. Note that m stands for *mean* and is not a "numerical" index.
The compatibility condition (4.1.4) constraints b_{0m} and b_{1m} as functions of the other parameters:

$$b_{0m} = u_{0m} \quad \text{and} \quad b_{1m} = u_{0m} + \sum_{l=1}^{n(u_0)} A_l^{u_0} \Phi_l^{u_0}(x)$$

so that our "compliant" parameters actually belong to $\mathcal{P}$ defined by:

$$\begin{aligned}\mathcal{P} = \Big\{ \mu = (\nu, b_{0m}, A_1^{b_0}, \dots, A_{n(b_0)}^{b_0}, b_{1m}, A_1^{b_1}, \dots, A_{n(b_1)}^{b_0}, \\ f_m, A_{11}^f, A_{12}^f, \dots, A_{1,n_S(f)}^f, A_{2,1}^f, \dots, A_{2,n_S(f)}^f, \dots, A_{n_T(f),n_S(f)}^f, \\ u_{0m}, A_1^{u_0}, \dots, A_{n(u_0)}^{u_0}) \in \mathcal{P}' \text{ satisfying (4.1.4)} \Big\} \end{aligned} \tag{4.2.1}$$

4.2.2 Offline/online procedure

The key heuristic [72] for RB approximation of the (linear) parametrized variational problem is the following: given $\mu \in \mathcal{P}$,

$$\text{find } u(\mu) \in X \text{ so that } A(u(\mu), v; \mu) = L(v; \mu),\ \forall v \in X$$

is to choose a parameter-independent family $\mathcal{R}$ of linearly independent functions in X — with $\#\mathcal{R} \ll \dim X$, to achieve computational economy — and then, given an instance of the parameter, to search for the reduced solution:

$$\widetilde{u}(\mu) \in \text{Span}\mathcal{R},\ \text{so that } A(\widetilde{u}(\mu), v; \mu) = L(v; \mu)\ \forall v \in \text{Span}\mathcal{R}$$

Let us apply this idea on problem (4.1.12). We rely on the procedure described in [71], modified to allow parametrization of initial condition, boundary data and source term. To simplify notations, we do not explicitly write dependence of u and $\widetilde{u}$ on μ.

We suppose that a reduced basis $\mathcal{R} = \{\zeta_1, \dots, \zeta_N\}$ has been chosen (see Subsection 4.2.3 for one way to do so); we define $\widetilde{X} = \text{Span}\mathcal{R}$ and we look for $\left\{\widetilde{u}^k\right\}_{k=0,\dots,\mathcal{T}} \subset \widetilde{X}$ satisfying:

$$\widetilde{u}^0 = \widetilde{\pi}\left(\pi(u_0)\right) \tag{4.2.2a}$$

and:

$$\begin{aligned}\left\langle \frac{\widetilde{u}^k - \widetilde{u}^{k-1}}{\Delta t}, v \right\rangle + c(\widetilde{u}^k, \widetilde{u}^k, v) + \nu a(\widetilde{u}^k, v) + B(\widetilde{u}^k, v) \\ = \ell_\pi(v, t_k) + b_0(t_k)\beta_0(v) + b_1(t_k)\beta_1(v)\ \ \forall v \in \widetilde{X}\ \forall k = 1, \dots, \mathcal{T}\end{aligned} \tag{4.2.2b}$$

where $\widetilde{\pi}$ is the orthogonal projection from X onto $\widetilde{X}$, with respect to the standard L^2 inner product on X: $\langle w, v\rangle = \displaystyle\int_0^1 wv$.

The offline/online procedure for computation of $\widetilde{u}^0$ will come easily from our parametrization of u_0 in Subsection 4.2.1: since the constant function $\mathbf{1} = \displaystyle\sum_{j=0}^{\mathcal{N}} \phi_j$ belongs to X, we have:

$$\widetilde{u}^0 = u_{0m}\widetilde{\pi}(\mathbf{1}) + \sum_{l=1}^{n(u_0)} A_l^{u_0}\, \widetilde{\pi}\left(\pi\left(\Phi_l^{u_0}(\cdot)\right)\right) \tag{4.2.3}$$

We now discuss computation of $\widetilde{u}^k$ from $\widetilde{u}^{k-1}$ for $k = 1, \dots, \mathcal{T}$. We are willing to proceed with Newton steps as for the solution of (4.1.12b). We denote, for $k = 1, \dots, \mathcal{T}$, by

$$\widetilde{u}^{k-1} = \sum_{j=1}^{N} u_j^{k-1} \zeta_j \quad ; \quad \overline{\widetilde{u}^k} = \sum_{j=1}^{N} \overline{u_j^k} \zeta_j$$

respectively, the reduced solution at time t_{k-1}, and previous guess for the reduced solution at time t_k.
Our procedure relies on the following proposition:

Proposition 1. *1. The Newton increment $\delta = \sum_{j=1}^{N} \delta_j \zeta_j$ satisfies the following equations:*

$$\begin{aligned} &\sum_{j=1}^{N} \delta_j \left\{ \frac{\langle \zeta_j, \zeta_i \rangle}{\Delta t} + 2 \sum_{j'=1}^{N} \overline{u_{j'}^k} c(\zeta_{j'}, \zeta_j, \zeta_i) + \nu a(\zeta_j, \zeta_i) + B(\zeta_j, \zeta_i) \right\} \\ &\qquad = \sum_{j=1}^{N} u_j^{k-1} \frac{\langle \zeta_j, \zeta_i \rangle}{\Delta t} + \ell_\pi(\zeta_i, t_k) + b_0(t)\beta_0(\zeta_i) + b_1(t)\beta_1(\zeta_i) \\ &- \sum_{j=1}^{N} \overline{u_j^k} \left(\frac{\langle \zeta_j, \zeta_i \rangle}{\Delta t} + \sum_{j'=1}^{N} \overline{u_{j'}^k} c(\zeta_{j'}, \zeta_j, \zeta_i) + \nu a(\zeta_j, \zeta_i) + B(\zeta_j, \zeta_i) \right) \quad \forall i = 1, \dots, N \end{aligned} \tag{4.2.4}$$

2. We have:

$$\ell_\pi(\zeta_i, t_k) = f_m \int_0^1 \zeta_i + \sum_{l=1}^{n_T(f)} \sum_{p=1}^{n_S(f)} A_{lp}^f \Phi_l^{fT}(t_k) \int_0^1 \pi \left(\Phi_p^{fS}(\cdot) \right) \zeta_i \tag{4.2.5}$$

for all $i = 1, \dots, N$ and $k = 1, \dots, \mathcal{T}$.

Proof. 1. Equation (4.2.2b) for $\widetilde{u}^k = \overline{\widetilde{u}^k} + \delta$ linearized near $\delta = 0$, for $v = \zeta_i$, $\forall i = 1, \dots, N$ is:

$$\begin{aligned} &\frac{1}{\Delta t} \langle \overline{\widetilde{u}^k} + \delta, \zeta_i \rangle + c(\overline{\widetilde{u}^k}, \overline{\widetilde{u}^k}, \zeta_i) + 2c(\overline{\widetilde{u}^k}, \delta, \zeta_i) + \nu a(\overline{\widetilde{u}^k} + \delta, \zeta_i) + B(\overline{\widetilde{u}^k} + \delta, \zeta_i) \\ &= \frac{1}{\Delta t} \langle \widetilde{u}^{k-1}, \zeta_i \rangle + \ell_\pi(\zeta_i, t_k) + b_0(t)\beta_0(\zeta_i) + b_1(t)\beta_1(\zeta_i) \quad \forall i = 1, \dots, N \end{aligned}$$

Rewriting this equation using expansions of $\overline{\widetilde{u}^k}$ and δ in $\mathcal{R}$ and linearity of $\langle \cdot, \cdot \rangle$, c, a and B with respect to their first argument, and putting all $(\delta_j)_j$-dependent terms on the left-hand side give the announced equation.

2. is a direct consequence of the parametrization of f given in Subsection 4.2.1.

□

The following Proposition 2 justifies the well-definedness of the Newton iteration, for an orthonormal reduced basis $\{\zeta_1, \ldots, \zeta_N\}$. In practice, the Gram-Schmidt process can always be used to ensure this condition.

Proposition 2. *Suppose that $\{\zeta_1, \ldots, \zeta_N\}$ is orthonormal with respect to $\langle \cdot, \cdot \rangle$.*
Then, for Δt small enough, i.e. $\Delta t < \Delta t^* (\nu, \{\zeta_i\}_{i=1,\ldots,N})$, and initial guess $\widetilde{u^k}$ sufficiently close to $\widetilde{u}^k$, the Newton iteration (4.2.4) is well defined and converges (quadratically) to $\widetilde{u}^k$.

Proof. Iteration (4.2.4) is a Newton iteration for solving $F(x) = \alpha$, for appropriate α and F given by:

$$F(x) = \sum_{i=1}^{N} \left(\frac{\langle x, \zeta_i \rangle}{\Delta t} + c(x, x, \zeta_i) + \nu a(x, \zeta_i) + B(x, \zeta_i) \right) \zeta_i$$

We apply the result stated pp. 353–355 of [80] and pp. 362–367 of [40]. To do so, we have to check that, for Δt small enough:

1. the differential of F at $\widetilde{u}^k$, denoted by $DF(\widetilde{u}^k)$, is invertible;
2. $DF(\cdot)$ is Lipschitz-continuous *i.e.* there exist $L > 0$ so that

$$\forall x, x' \in X, \quad \forall v \in X, \quad \left\| DF(x) \cdot v - DF(x') \cdot v \right\| \leq L \left\| x - x' \right\| \|v\|$$

It is easy to check that the matrix of $DF(\widetilde{u}^k)$ in the reduced basis $\{\zeta_1, \ldots, \zeta_N\}$ is diagonally dominant, hence invertible, for:

$$\Delta t < \Delta t^* := \min_{i=1,\ldots,N} \frac{1}{\sum_{j=1}^{N} |2c(\widetilde{u}^k, \zeta_j, \zeta_i) + \nu a(\zeta_j, \zeta_i) + B(\zeta_j, \zeta_i)|}$$

and (2) is a straightforward computation. □

We put our offline/online procedure in Algorithm 1.

– *offline:*

1. choose a parameter-, and time-independent reduced basis $\{\zeta_1, \dots, \zeta_N\}$ (see Section 4.2.3)
2. compute and store the following parameter-independent functions of $\widetilde{X}$ and scalars, for all $i, j, j' = 1, \dots, N$, $l = 1, \dots, n(u_0)$, $p = 1, \dots, n_S(f)$:

$$\begin{array}{cc} \widetilde{\pi}(\mathbf{1}) & \widetilde{\pi}\left(\pi\left(\Phi_l^{u_0}(\cdot)\right)\right) \\ \langle \zeta_j, \zeta_i \rangle & a(\zeta_j, \zeta_i) \\ c(\zeta_{j'}, \zeta_j, \zeta_i) & B(\zeta_j, \zeta_i) \\ \beta_0(\zeta_i) & \beta_1(\zeta_i) \\ \displaystyle\int_0^1 \zeta_i & \displaystyle\int_0^1 \pi\left(\Phi_p^{fS}(\cdot)\right)\zeta_i \end{array}$$

– *online:*

1. assemble $\widetilde{u}^0$ as the linear combination (4.2.3) ;
2. for $k = 1, \dots, \mathcal{T}$:
   (a) set up an initial guess $\overline{\widetilde{u}^k} = \sum_{j=1}^{N} \overline{u_j^k}\zeta_j$;
   (b) compute and store $\ell_\pi(\zeta_i, t_k)$ by using (4.2.5) ;
   (c) look for $\delta = \sum_{j=1}^{N} \delta_j \zeta_j$ by solving the linear system (4.2.4) ;
   (d) set $\widetilde{u}^k \leftarrow \overline{\widetilde{u}^k} + \delta$;
   (e) if $\|\delta\|$ is small enough:
      i. output $\widetilde{u}^k$
      ii. set $k = k + 1$
   (f) else:
      i. update the guess : $\overline{\widetilde{u}^k} \leftarrow \widetilde{u}^k$, i.e. $\widetilde{u}_j^k \leftarrow u_j^k \ \forall j = 1, \dots, N$
      ii. go back to (c)

Algorithm 1: offline/online procedure

Let us make some remarks about the complexity of the above online algorithm, in contrast with the "full basis" one described in Section 4.1.2:

Remark 1. *1. The most computationally demanding step is (2) (c), since it involves resolution of a (nonsymmetric, dense) $N \times N$ linear system;*

the "full basis" counterpart solves $(\mathcal{N}+1)\times(\mathcal{N}+1)$ *nonsymmetric tridiagonal system. Thus significant computational savings are expected for* $N \ll \mathcal{N}$.

2. *Thanks to our parametrization of* $f(t,\cdot)$, *all integrals over* $[0;1]$ *in equation* (4.2.4) *can be precomputed during the offline phase, yielding a* $\mathcal{N}$-independent *online phase. This means that one can in principle choose* arbitrary *high precision on the full model* without *impact on the marginal cost of evaluation of an online solution. This "*$\mathcal{N}$*-independence" property shall be required of every complexity of any online procedure. One should also note that our online procedure does not produce "nodal" values* $\widetilde{u}^k(x_i)$, $i = 0,\dots,\mathcal{N}$ *(as this would clearly violate the* $\mathcal{N}$*-independence), but rather the components of* $\widetilde{u}^k$ *in the reduced basis.*

3. *Taken independently, the number of parameters* $n(u_0)$, $n(b_0)$, $n(b_1)$, $n_S(f)$ *and* $n_T(f)$, *as well as the number of timesteps* $\mathcal{T}$ *have a linear impact on the online complexity. Moreover, due to the double sum in* (4.2.5), *the online complexity is proportional to* $n_S(f)n_T(f)$. *We note that an advantage of treating the Dirichlet boundary condition weakly, as we do in this paper, is that* $n_T(f)$ *does not get increased by functions of* $n(b_0)$ *or* $n(b_1)$. *This is a clear advantage of our method: when boundary conditions are treated by returning to an homogeneous Dirichlet problem, as in [59],* $n(b_0)+n(b_1)$ *terms are added in the parametrization of* f.

4.2.3 Choice of the reduced basis

In this subsection, we describe different ways of choosing a pertinent reduced basis $\{\zeta_1,\dots,\zeta_N\}$. These lead to three different bases fitting into the certified (that is to say, the three admit the same procedure for online error bound we describe in Section 4.3) reduced basis framework.

The first is based on proper orthogonal decomposition (POD), the second is based on a "greedy" selection algorithm. The third is an hybridation of POD and greedy. These methods are standard in the literature.

Notation

The two procedures described below involve computation of the reference solution for different instances of the parameter, so we should use special notations, local to this section, to emphasize the dependence of the reference solution on the parameters. We define a parametrized solution u by:

$$u : \begin{cases} \{1, \dots, \mathcal{T}\} \times \mathcal{P} \to X \\ \qquad (k, \mu) \mapsto u(k, \mu) = \widetilde{u}^k \text{ satisfying (4.1.12b) for } \mu \text{ as parameter} \end{cases}$$

POD-driven procedure

We denote by $\{u_j^k(\mu)\}_j$ the coordinates of $u(k, \mu)$ in the basis $\{\phi_0, \dots, \phi_{\mathcal{N}}\}$:

$$u(k, \mu) = \sum_{j=1}^{\mathcal{N}} u_j^k(\mu)\phi_j$$

In the POD-based procedure (see [18]) of the reduced basis choice, we choose a finite-sized parameter sample $\Xi \subset \mathcal{P}$, compute the reference solutions $u(k, \mu)$ for all $k = 1, \dots, \mathcal{T}$ and all $\mu \in \Xi$, and form the *snapshots matrix* containing the components of these solutions in our basis $\{\phi_0, \dots, \phi_{\mathcal{N}}\}$:

$$M = \begin{pmatrix} u_0^0(\mu_1) & u_0^1(\mu_1) & \cdots & u_0^{\mathcal{T}}(\mu_1) & u_0^0(\mu_2) & \cdots & \cdots & u_0^{\mathcal{T}}(\mu_S) \\ u_1^0(\mu_1) & u_1^1(\mu_1) & \cdots & u_1^{\mathcal{T}}(\mu_1) & u_1^0(\mu_2) & \cdots & \cdots & u_1^{\mathcal{T}}(\mu_S) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_{\mathcal{N}}^0(\mu_1) & u_{\mathcal{N}}^1(\mu_1) & \cdots & u_{\mathcal{N}}^{\mathcal{T}}(\mu_1) & u_{\mathcal{N}}^0(\mu_2) & \cdots & \cdots & u_{\mathcal{N}}^{\mathcal{T}}(\mu_S) \end{pmatrix}$$

where $\Xi = \{\mu_1, \dots, \mu_S\}$.

One can check that M has $\mathcal{N} + 1$ rows and $S(\mathcal{T} + 1)$ columns.

To finish, we choose the size $N < S(\mathcal{T} + 1)$ of the desired reduced basis, we form the $S(\mathcal{T}+1) \times S(\mathcal{T}+1)$ non-negative symmetric matrix $M^T \Omega M$, where Ω is the matrix of our inner product $\langle \cdot, \cdot \rangle$, to find $z_1, \dots, z_N$ the N leading nonzero eigenvectors of this matrix (that is, the ones associated with the N largest eigenvalues, counting repeatedly possible nonsimple eigenvalues), and, for $i = 1, \dots, N$, the coordinates of ζ_i with respect to $\{\phi_0, \dots, \phi_{\mathcal{N}}\}$ are given by:

$$\frac{1}{\|Mz_i\|} Mz_i \tag{4.2.6}$$

"Local" greedy selection procedure

The *greedy* basis selection algorithm (cf. [45, 43]) is the following:

Parameter: N, the desired size of the reduced basis.

1. Choose a finite-sized, random, large sample of parameters $\Xi \subset \mathcal{P}$.
2. Choose $\mu_1 \in \mathcal{P}$ and $k_1 \in \{1, \dots, \mathcal{T}\}$ at random, and set
$$\zeta_1 = \frac{u(k_1, \mu_1)}{\|u(k_1, \mu_1)\|}$$
3. Repeat, for n from 2 to N:
   (a) Find:
   $$(k_n, \mu_n) = \underset{(k,\mu)\in 0,\dots,\mathcal{T}\}\times\Xi}{\operatorname{argmax}} \varepsilon^*(\mu, k)$$
   where $\varepsilon^*(k, \mu)$ is a (fastly evaluated) estimator of the RB error $\|u(k, \mu) - \widetilde{u}(k, \mu)\|$, where $\widetilde{u}(k, \mu)$ stands for the RB approximation to $u(k, \mu)$ using $\{\zeta_1, \dots, \zeta_{n-1}\}$ as reduced basis (see below).
   (b) Compute $\zeta_n^* = u(k_n, \mu_n)$.
   (c) Using one step of the (stabilized) Gram-Schmidt algorithm, find $\zeta_n \in \operatorname{Span}\{\zeta_1, \dots, \zeta_{n-1}, \zeta_n^*\}$ so that $\{\zeta_1, \dots, \zeta_{n-1}, \zeta_n\}$ is an orthonormal family of $(X, \langle\cdot,\cdot\rangle)$.

Algorithm 2: Greedy basis selection

The "greedy" name for this algorithm comes from the fact that the algorithm chooses, at each step of the repeat loop, the "best possible" time and parameter tuple to the reduced basis, that is the one for which the RB approximation error is estimated to be the worst.

Let's now discuss the choice for the error indicator ε^*. A natural candidate would be the online error bound ε described in Section 4.3. However, as we shall see in the next section, the bound for timestep t_k is a compound of the "propagation" of the error made in the previous timesteps $t_0, t_1, \dots, t_{k-1}$ (the ε_{k-1} term) and the "local error" just introduced at the k-th time discretization. Hence, this error estimator ε_k has a natural tendency to grow (exponentially) with k. Thus using it as error indicator ε^* will favor times k_n near final time $\mathcal{T}$ to be chosen at step 3.(a) of the algorithm below. Such choices are suboptimal, because including them in the reduced basis will not fix this exponential growth problem which is inherent to our approximation.

Instead we use, as in [45], a purely *local*-at-t_k error indicator, that is: the computable error bound described in Section 4.3 when taking $\varepsilon_{k-1} = 0$.
It has been noted in the literature that the greedy procedure can "stall" (i.e. select vectors for addition in the basis which have no effect in decreasing the error bounds). The author in [43] provides a "back-up procedure" in such a case. In our experiments, however, we have not noticed any "stalling" of the greedy procedure, maybe because of our sharper error bound.

POD-Greedy procedure

We can also make use of the POD-Greedy procedure [46] in Algorithm 3. This procedure aims to combine the advantages of the greedy and the POD based procedure; it has also been proposed so as to overcome the "stalling" of the greedy procedure.

Parameter: P_1.

1. Choose a finite-sized, random, large sample of parameters $\Xi \subset \mathcal{P}$.
2. Choose $\mu_1 \in \mathcal{P}$ at random, and choose as the current reduced basis $\mathcal{B}$ an orthonormalized basis of $\mathrm{Span}\{u(\mu_1, t_0), u(\mu_1, t_1), \ldots, u(\mu_1, \mathcal{T})\}$.
3. Repeat, until the desired number of items in the basis is reached:
   (a) Find:
   $$\mu^* = \operatorname*{argmax}_{\mu \in \Xi} \varepsilon^*(\mu)$$
   where $\varepsilon^*(\mu)$ is a (fastly evaluated) estimator of the RB error $\|u(\mu) - \widetilde{u}(\mu)\|$, where $\widetilde{u}(\mu)$ stands for the RB approximation to $u(\mu)$ using $\mathcal{B}$ as reduced basis.
   (b) Append to $\mathcal{B}$ the P_1 leading POD modes of $\{u^{\mathrm{proj}}(\mu^*, t_0), u^{\mathrm{proj}}(\mu^*, t_1), \ldots, u^{\mathrm{proj}}(\mu^*, \mathcal{T})\}$, where the proj superscript denotes projection on the orthogonal complement of $\mathrm{Span}(\mathcal{B})$.
4. Output $\mathcal{B} = \{\zeta_1, \ldots, \zeta_{\#\mathcal{B}}\}$ as reduced basis.

Algorithm 3: POD-Greedy basis selection

Again, an error indicator ε^* has to be chosen. We do as in [62], where the authors take the online error bound at final time.

Expansion of the basis by initial data modes

In case they did not get chosen by the POD or greedy algorithm, a classical strategy is to initialize the basis selection algorithms by taking in the reduced basis the constant function $\mathbf{1}$, and the functions $\Phi_l^{u_0}(\cdot)$ for $l = 1, \ldots, n(u_0)$. This may increase the size of the reduced basis (and thus online computation times) but ensures that $\widetilde{u}^0 = u_0$ (i.e. initial error is zero). Such an enrichment can possibly be a good move, as the error gets accumulated and amplified throughout the time iterations, zero initial error will certainly reduce the (estimated, as well as actual) RB approximation error.

4.3 Error bound

In this section, we derive a parameter and time dependent online error bound ε^k (for $k = 1, \ldots, \mathcal{T}$) satisfying: $\left\| u_e^k - \widetilde{u}^k \right\| \leq \varepsilon^k$, where $\|v\| = \left(\int_0^1 v^2 \right)^{1/2}$, and u_e^k is our reference "truth" solution in X satisfying the fully discretized PDE with strong Dirichlet enforcement, that is:

$$\begin{cases} \dfrac{\langle u_e^k - u_e^{k-1}, v \rangle}{\Delta t} + c(u_e^k, u_e^k, v) + \nu a(u_e^k, v) = \ell_\pi(v, t_k) \quad \forall v \in X_0 \\ u_e^k(0) = b_0(t_k) \\ u_e^k(1) = b_1(t_k) \end{cases} \tag{4.3.1}$$

where X_0 is the "homogeneous" subspace of X:

$$X_0 = \{ v \in X \text{ st. } v(0) = v(1) = 0 \}.$$

Our error bound should be precise enough (i.e., not overestimating the actual error $\left\| u_e^k - \widetilde{u}^k \right\|$ too much, and approaching zero as N increases) and online-efficient (that is, admit an offline/online computation procedure with an $\mathcal{N}$-independent online complexity).

We notice that our error bound measures the error between the reduced and the reference solution; it does not reflect the discretization error made when replacing the actual analytical solution of the Burgers equation with its numerical approximation (4.3.1). This is consistent with the fact that RB methods relies strongly on the existence of a high-fidelity numerical approximation of the analytical solution by a discrete one, hence regarded as "truth".

Subsection 4.3.1 deals with the derivation of the error bound; this error involves quantities whose computation is detailed in Subsections 4.3.2 and 4.3.3.

4.3.1 Error bound

The sketch of the derivation of our error bound is the following: we first give a "theoretical" error bound in Theorem 4; we then replace the uncomputable quantities appearing in this bound by their computable surrogates in paragraph 4.3.1.

Theoretical error bound

Notation. We suppose that the convergence tests appearing in the Newton iterations performed in Section 4.1.2 and Section 4.2.2 are sufficiently demanding so as to neglect the errors due to the iterative solution of the nonlinear systems (4.1.13) and (4.2.2b).
We now set up some notations : first the error at time t_k:

$$e_k = \begin{cases} u_e^k - \widetilde{u}^k & \text{if } k > 0 \\ \pi(u_0) - \widetilde{\pi}\left(\pi(u_0)\right) & \text{if } k = 0 \end{cases}$$

the residual form r_k, for $v \in X_0$:

$$r_k(v) = \ell_\pi(v, t_k) - \frac{1}{\Delta t}\langle \widetilde{u}^k - \widetilde{u}^{k-1}, v\rangle - c(\widetilde{u}^k, \widetilde{u}^k, v) - \nu a(\widetilde{u}^k, v) \qquad (4.3.2)$$

and the "X_0-norm" of the residual:

$$\|r_k\|_0 = \sup_{v\in X_0, \|v\|=1} r_k(v)$$

We introduce :

$$\psi_k(v, w) = 2c(\widetilde{u}^k, v, w) + \nu a(v, w)$$

and the so-called *stability constants*:

$$C_k = \inf_{v\in X_0, \|v\|=1} \psi_k(v, v) \qquad (4.3.3)$$

To finish, we define:

$$\eta_k = |e_k(0)|\,\|\phi_0\| + |e_k(1)|\,\|\phi_{\mathcal{N}}\| \quad ; \quad \sigma_k = 2\,|C_k|\,\eta_k \quad ; \quad \mathcal{E} = \sup_{v\in X_0, \|v\|=1} v\left(\frac{1}{\mathcal{N}}\right)$$

($\mathcal{E}$ is finite because X_0 is finite dimensional), and, finally:

$$f_k = \mathcal{E}\left(|e_k(0)|\,|\psi_k(\phi_0,\phi_1)| + |e_k(1)|\,|\psi_k(\phi_{\mathcal{N}},\phi_{\mathcal{N}-1})| \right)$$

$$\xi_k^{\mathcal{A}} = \frac{\mathcal{E}^2(|e_k(0)| + |e_k(1)|)}{3} \quad ; \quad \xi_k^{\mathcal{B}} = \frac{5}{3}\mathcal{E}\left(e_k(0)^2 + e_k(1)^2\right)$$

$$\xi_k^{\gamma} = \frac{|e_k(0)|^3 + |e_k(1)|^3}{3}$$

We will also make use of the standard notation:

$$[x]_- = \max(-x, 0) \quad \forall x \in \mathbb{R}$$

The theoretical foundation for our error bound is the following theorem. As the technique developed in [71] did not fit our problem (because of the weak Dirichlet treatment, whose advantage has been shown in Remark 1 point (3)), we developed a new strategy for obtaining this error bound.

Theorem 4. *If:*

$$\frac{1}{\Delta t} + C_k - \xi_k^{\mathcal{A}} > 0 \;\forall k = 1, \ldots, \mathcal{T} \tag{4.3.4}$$

then the norm of the error $\|e_k\|$ *satisfies:*

$$\|e_k\| \leq \begin{cases} \dfrac{\mathcal{B}_k + \sqrt{\mathcal{D}_k}}{2\mathcal{A}_k} & \text{if } \mathcal{D}_k \geq 0 \\ \dfrac{\mathcal{B}_k}{\mathcal{A}_k} & \text{if } \mathcal{D}_k < 0 \end{cases}$$

with:

$$\mathcal{A}_k = \frac{1}{\Delta t} + C_k - \xi_k^{\mathcal{A}}$$

$$\mathcal{B}_k = \frac{2\eta_k + \|e_{k-1}\| + \mathcal{E}\langle\phi_0,\phi_1\rangle(|e_k(0)| + |e_k(1)|)}{\Delta t} + \sigma_k + f_k + \|r_k\|_0 + \xi_k^{\mathcal{B}}$$

$$\gamma_k = \frac{\eta_k\,\|e_{k-1}\| + \mathcal{E}\eta_k\langle\phi_0,\phi_1\rangle(|e_k(0)| + |e_k(1)|)}{\Delta t} + \eta_k f_k + [C_k]_-\,\eta_k^2 + \frac{1}{6}\left|e_k(1)^3 - e_k(0)^3\right| + \xi_k^{\gamma} + \|r_k\|_0\,\eta_k$$

and:

$$\mathcal{D}_k = (\mathcal{B}_k)^2 + 4\mathcal{A}_k\gamma_k.$$

The proof of Theorem 4 is presented in Section 4.5.

Computable error bound.

We now find an efficiently computable (that is, with an offline/online decomposition, with a complexity of the online part independent of $\mathcal{N}$) error bound ε_k derived from the one described above; to do so we discuss each of the ingredients appearing in its expression.

- Computation of the norm of the initial error $\|e_0\|$ is addressed in the next Section 4.3.2; the one of $\|r_k\|_0$ is in Section 4.3.3.
- We have, for $w \in \{0,1\}$:

$$e_k(w) = u_e^k(w) - \widetilde{u}^k(w) = b_w(t_k) - \widetilde{u}^k(w)$$

  so that $e_k(0)$ and $e_k(1)$ can be computed during the online phase.
- Similarly, the scalars $\|\phi_0\|$, $\psi_k(\phi_0, \phi_1)$ and $\psi_k(\phi_{\mathcal{N}}, \phi_{\mathcal{N}-1})$ can straightforwardly be computed online.
- The "continuity constant" $\mathcal{E}$ can be computed offline and stored by solving the optimization problem defining it. Thus η_k, f_k, $\xi_k^{\mathcal{A}}$, $\xi_k^{\mathcal{B}}$ and ξ_k^{γ} can be computed online.
- The exact value of C_k could be found by solving a generalized eigenvalue problem on X: C_k is the smallest $\lambda \in \mathbb{R}$ so that there exists $z \in X_0$, $\|z\| = 1$ satisfying:

$$\psi_k^{Sym}(z, v) = \lambda \langle z, v \rangle \quad \forall v \in X_0$$

  with ψ_k^{Sym} the symmetric bilinear form defined by:

$$\psi_k^{Sym}(w, v) = \nu a(w, v) + c(\widetilde{u}^k, w, v) + c(\widetilde{u}^k, v, w) \quad \forall w, v \in X_0$$

  The cost of doing so is prohibitive as it is an increasing function of $\dim X = \mathcal{N} + 1$. Instead, we will see in Section 4.3.4 how to compute lower and upper bounds C_k^{inf} and C_k^{sup}: $C_k^{inf} \leq C_k \leq C_k^{sup}$.
- We can then compute the following lower and upper bounds for $\mathcal{A}_k$:

$$\mathcal{A}_k^{inf} = \frac{1}{\Delta t} + C_k^{inf} - \xi_k^{\mathcal{A}} \qquad ; \qquad \mathcal{A}_k^{sup} = \frac{1}{\Delta t} + C_k^{sup} - \xi_k^{\mathcal{A}}$$

  and the hypothesis (4.3.4) is ensured by checking that $\mathcal{A}_k^{inf} > 0$.
- We can also compute an upper bound of σ_k:

$$\sigma_k^{sup} = 2\eta_k \max\left(\left| C_k^{sup} \right|, \left| C_k^{inf} \right| \right)$$

– To compute an upper bound of $\mathcal{B}_k$, we need to replace the preceding error norm $\|e_{k-1}\|$ which is (except for $k=1$) not exactly computable, with the online upper bound $\varepsilon_{k-1} \geq \|e_{k-1}\|$ at the preceding time step:

$$\mathcal{B}_k^{sup} = \frac{2\eta_k + \varepsilon_{k-1} + \mathcal{E}\langle\phi_0,\phi_1\rangle(|e_k(0)| + |e_k(1)|)}{\Delta t} + \sigma_k^{sup} + f_k + \|r_k\|_0 + \xi_k^{\mathcal{B}}$$

– And γ_k gets replaced by its upper bound γ_k^{sup}:

$$\gamma_k^{sup} = \frac{\eta_k\varepsilon_{k-1} + \mathcal{E}\eta_k\langle\phi_0,\phi_1\rangle(|e_k(0)| + |e_k(1)|)}{\Delta t} + \eta_k f_k + \left[C_k^{inf}\right]_- \eta_k^2 + \frac{1}{6}\left|e_k(1)^3 - e_k(0)^3\right| + \xi_k^{\gamma} + \|r_k\|_0\,\eta_k$$

– We finally compute an upper bound for $\mathcal{D}_k$:

$$\mathcal{D}_k^{sup} = \begin{cases} \left(\mathcal{B}_k^{sup}\right)^2 + 4\mathcal{A}_k^{sup}\gamma_k^{sup} & \text{if } \gamma_k^{sup} \geq 0 \\ \left(\mathcal{B}_k^{sup}\right)^2 + 4\mathcal{A}_k^{inf}\gamma_k^{sup} & \text{if } \gamma_k^{sup} < 0. \end{cases}$$

and our "computable" error bound is then:

$$\begin{cases} \dfrac{\mathcal{B}_k^{sup} + \sqrt{\mathcal{D}_k^{sup}}}{2\mathcal{A}_k^{inf}} & \text{if } \mathcal{D}_k^{sup} \geq 0 \\ \dfrac{\mathcal{B}_k^{sup}}{\mathcal{A}_k^{inf}} & \text{if } \mathcal{D}_k^{sup} < 0. \end{cases}$$

The remainder of the section consists in the description of computation of the four left-out quantities $\|e_0\|$, $\|r_k\|_0$, C_k^{sup} and C_k^{inf}.

4.3.2 Initial error

The present subsection deals with efficient computation of the $\|e_0\|$ term in the computable error bound described in paragraph 4.3.1. We denote by $\mathbf{H}$ the Gram matrix of the family:

$$\left\{\mathbf{1} - \widetilde{\pi}(\mathbf{1}), \pi\left(\Phi_1^{u_0}(\cdot)\right) - \widetilde{\pi}\left(\pi\left(\Phi_1^{u_0}(\cdot)\right)\right), \ldots, \pi\left(\Phi_{n(u_0)}^{u_0}(\cdot)\right) - \widetilde{\pi}\left(\pi\left(\Phi_{n(u_0)}^{u_0}(\cdot)\right)\right)\right\}$$

that is, $\mathbf{H}$ is the $(1+n(u_0)) \times (1+n(u_0))$ symmetric matrix of all the inner products between two any of the above vectors and by $\mathbf{e_0}$ the vector containing the components of e_0 with respect to the family above, *i.e.*:

$$\mathbf{e_0} = \left(u_{0m}, A_1^{u_0}, \ldots, A_{n(u_0)}^{u_0}\right)^T$$

We have:

Lemma 6. *The norm of the initial error* $\|e_0\|$ *is given by:*

$$||e_0|| = \sqrt{\mathbf{e_0}^T \mathbf{H} \mathbf{e_0}} \tag{4.3.5}$$

Proof. Parametrization 4.2.1 gives:

$$e_0 = \pi(u_0) - \widetilde{\pi}\left(\pi(u_0)\right) = u_{0m}\left(\mathbf{1} - \widetilde{\pi}(\mathbf{1})\right) + \sum_{l=1}^{n(u_0)} A_l^{u_0}\left(\pi\left(\Phi_l^{u_0}(\cdot)\right) - \widetilde{\pi}\left(\pi\left(\Phi_l^{u_0}(\cdot)\right)\right)\right)$$

and the result follows from the expansion of $\|e_0\|^2$ when e_0 is replaced by the expression above. □

This formula allows us to compute the time and parameter-independent Gram matrix $\mathbf{H}$ during the offline phase, and, during the online phase, to assemble the $(1 + n(u_0))$-vector $\mathbf{e_0}$ and to perform (4.3.5) to get $||e_0||$ with an online cost dependent only of $n(u_0)$.

4.3.3 Norm of the residual

We now present the computation of the $\|r_k\|_0$ term in the computable error bound of Section 4.3.1.

– Let the expansions of the reduced solutions with respect to the reduced basis be:

$$\widetilde{u}^p = \sum_{j=1}^{N} u_j^p \zeta_j \quad \text{for } p \in \{k-1, k\}$$

– Let $\mathbf{G}$ be the $(1+n_S(f)+2N+N^2)\times(1+n_S(f)+2N+N^2)$-sized Gram matrix of

$$\{\Gamma^{int}, \Gamma_1^{fS}, \ldots, \Gamma_{n_S(f)}^{fS}, \Gamma_1^{\langle\rangle}, \ldots, \Gamma_N^{\langle\rangle}, \Gamma_{1,1}^c, \Gamma_{1,2}^c, \ldots, \Gamma_{1,N}^c, \Gamma_{2,1}^c, \ldots, \Gamma_{2,N}^c, \ldots, \Gamma_{N,N}^c, \Gamma_1^a, \ldots, \Gamma_N^a\}$$

where $\Gamma^{int}, \Gamma_p^{fS}, \Gamma_j^{\langle\rangle}, \Gamma_{j,j'}^c, \Gamma_j^a \in X_0$ $(p = 1, \ldots, n_S(f); j, j' = 1, \ldots, N)$ satisfy:

$$\begin{aligned}
\langle \Gamma^{int}, v\rangle &= \int_0^1 v && \forall v \in X_0\\
\langle \Gamma_p^{fS}, v\rangle &= \int_0^1 \pi\left(\Phi_p^{fS}(\cdot)\right) v && \forall v \in X_0\\
\langle \Gamma_j^{\langle\rangle}, v\rangle &= \langle \zeta_j, v\rangle && \forall v \in X_0\\
\langle \Gamma_{j,j'}^c, v\rangle &= c(\zeta_j, \zeta_{j'}, v) && \forall v \in X_0\\
\langle \Gamma_j^a, v\rangle &= a(\zeta_j, v) && \forall v \in X_0
\end{aligned}$$

(Those Γ's exist by virtue of the Riesz representation theorem).

– Let $\boldsymbol{\rho}_k$ be the following vector:

$$\boldsymbol{\rho}_k = \Big(f_m, \sum_{l=1}^{n_T(f)} A^f_{l,j}\Phi^{fT}_l(t_k)\ (j=1,\dots,n_S(f)), -\frac{1}{\Delta t}(u^k_j - u^{k-1}_j)\ (j=1,\dots,N),$$
$$u^k_j u^k_l\ (j,l=1,\dots,N), \nu u^k_j\ (j=1,\dots,N)\Big)^T$$

Lemma 7. *We have:*

$$\|r_k\|_0 = \sqrt{\boldsymbol{\rho}_k^T \mathbf{G}\boldsymbol{\rho}_k} \tag{4.3.6}$$

Proof. From the Riesz representation theorem, there exists a unique $\rho_k \in X_0$ so that

$$\langle \rho_k, v\rangle = r_k(v)\ \ \forall v \in X_0 \tag{4.3.7}$$

and we have: $\|r_k\|_0 = \|\rho_k\|$.
From (4.3.7) and the definition of r_k (4.3.2), we have that ρ_k is defined uniquely by:

$$\langle \rho_k, v\rangle = \ell_\pi(v,t_k) - \frac{1}{\Delta t}\langle \widetilde{u}^k - \widetilde{u}^{k-1}, v\rangle - c(\widetilde{u}^k, \widetilde{u}^k, v) - \nu a(\widetilde{u}^k, v) \forall v \in X_0$$

because $\beta_0(v) = \beta_1(v) = B(\cdot, v) = 0$ for all $v \in X_0$.
Using parametrization (4.2.5) of $\ell_\pi(\cdot, t_k)$, we get that (4.3.7) is equivalent to:

$$\langle \rho_k, v\rangle = f_m \int_0^1 v + \sum_{l=1}^{n_T(f)}\sum_{p=1}^{n_S(f)} A^f_{lp}\Phi^{fT}_l(t_k)\int_0^1 \pi\left(\Phi^{fS}_p(\cdot)\right) v - \frac{1}{\Delta t}\sum_{j=1}^{N}(u^k_j - u^{k-1}_j)\langle \zeta_j, v\rangle$$
$$- \sum_{j=1}^{N} u^k_j \left(\sum_{j'=1}^{N} u^k_{j'} c(\zeta_j, \zeta_{j'}, v) + \nu a(\zeta_j, v)\right)\ \ \forall v \in X_0 \tag{4.3.8}$$

By the superposition principle, ρ_k can be written as the linear combination:

$$\rho_k = f_m \Gamma^{int} + \sum_{p=1}^{n_S(f)}\left(\sum_{l=1}^{n_T(f)} A^f_{lp}\Phi^{fT}_l(t_k)\right)\Gamma^{fS}_p - \frac{1}{\Delta t}\sum_{j=1}^{N}(u^k_j - u^{k-1}_j)\Gamma^{\langle\rangle}_j$$
$$- \sum_{j=1}^{N}\sum_{j'=1}^{N} u^k_j u^k_{j'}\Gamma^c_{j,j'} - \sum_{j=1}^{N}\nu u^k_j \Gamma^a_j$$

Thus $\boldsymbol{\rho}_k$ contains the components of ρ_k with respect to the family whose $\mathbf{G}$ is the Gram matrix, and so:

$$\|r_k\|_0 = \|\rho_k\| = \sqrt{\boldsymbol{\rho}_k^T \mathbf{G}\boldsymbol{\rho}_k}. \tag{4.3.9}$$

□

The offline/online decomposition for computation of $\|r_k\|_0$ is as follows: in the offline phase, we compute the Γ's vectors, and compute and store their Gram matrix $\mathbf{G}$. In the online phase, we compute $\boldsymbol{\rho}_k$ and compute $\|r_k\|_0$ using (4.3.6). Note that one can reduce offline and online computational burden, as well as storage requirement, by noticing that $\Gamma^c_{j,j'} = \Gamma^c_{j',j}$ for all j, j'.

The cost of computation (and storage) of ρ_k and $\|\rho_k\|$ asymptotically dominates the cost of the online phase. This cost is in $O\left(\left(n_S(f)n_T(f) + N^2\right)^2\right)$. Again, we see that our weak Dirichlet treatment, as $n_T(f)$ remains independent of $n(b_0)$ and $n(b_1)$, allows for a better complexity of the online phase.

4.3.4 Lower and upper bounds on stability constant

To find lower and upper bounds on C_k efficiently in order to use them in our computable error bound of Section 4.3.1, we turn to the successive constraints method (SCM) [51, 71]. Here we present the application to our case for the sake of self-containedness. Our difference is the use of the metric (4.3.10) during the constraint-selection phase.

Notation. As in Section 4.2.3, we will need to handle reduced solutions of several values of the parameter tuple $\mu \in \mathcal{P}$ (see (4.2.1)), for different timesteps $k = 1, \ldots, \mathcal{T}$. We thus define an application that is the "reduced" counterpart of u defined in Section 4.2.3:

$$\widetilde{u} : \begin{cases} \{1, \ldots, \mathcal{T}\} \times \mathcal{P} \to \widetilde{X}_0 \\ \qquad (k, \mu) \mapsto \widetilde{u}(k, \mu) = \widetilde{u}^k \text{ satisfying (4.2.2b) for } \mu \text{ as parameter} \end{cases}$$

We make C_k depend explicitly on μ by defining:

$$C_k(\mu) = \inf_{v \in X_0, \|v\|=1} \left[2c(\widetilde{u}(k, \mu), v, v) + \nu(\mu)a(v, v)\right]$$

SCM lower bound. We now proceed to the derivation of the SCM lower bound of $C_k(\mu)$. We use the RB expansion: $\widetilde{u}(k, \mu) = \sum_{j=1}^{N} u_j^k(\mu)\zeta_j$ to rewrite

$C_k(\mu)$ as:

$$\begin{aligned} C_k(\mu) &= \inf_{v\in X_0,\|v\|=1}\left[\sum_{j=1}^{N} 2u_j^k(\mu)c(\zeta_j,v,v)+\nu a(v,v)\right] \\ &= \inf_{y=(y_1,\ldots,y_{N+1})\in\mathcal{Y}}\left[\sum_{j=1}^{N} 2u_j^k(\mu)y_j+\nu y_{N+1}\right] \\ &= \inf_{y\in\mathcal{Y}}\mathcal{J}(\mu,k,y) \end{aligned}$$

where:

$$\mathcal{Y}=\{y=(y_1,\ldots,y_{N+1})\in\mathbb{R}^{N+1}|\exists v\in X_0,\|v\|=1 \text{ s.t.} \\ y_j=c(\zeta_j,v,v)\forall j=1,\ldots,N, y_{N+1}=a(v,v)\}$$

and:

$$\mathcal{J}(\mu,k,y)=2\sum_{j=1}^{N}u_j^k(\mu)y_j+\nu y_{N+1}$$

We define, for a given "constraint subset" $\mathcal{C}\subset\{1,\ldots,\mathcal{T}\}\times P$ that will be chosen later:

– $\widetilde{\mathcal{Y}}=\left\{(y_1,\ldots,y_{N+1})\in\prod_{i=1}^{N+1}\left[\sigma_i^{min};\sigma_i^{max}\right]|\mathcal{J}(\mu',k',y)\geq C_{k'}(\mu'),\ \forall(\mu',k')\in\mathcal{S}(\mu,k)\right\}$
with $\mathcal{S}(\mu,k)$ standing for the set of the M points in $\mathcal{C}$ that are closest to (μ,k) with respect to this metric:

$$d((\mu,k),(\mu',k'))=\sum_{i=1}^{\dim\mathcal{P}}\left(\frac{\mu^i-\mu'^i}{\mu_{min}^i-\mu_{max}^i}\right)^2+\left(\frac{k-k'}{\mathcal{T}}\right)^2 \qquad (4.3.10)$$

where $\left(\mu^1,\ldots,\mu^{\dim\mathcal{P}}\right)$ are the coordinates of $\mu\in\mathcal{P}$, and:

$$\mu_{min}^i=\min_{\mu\in\mathcal{P}}\mu^i,\qquad \mu_{max}^i=\max_{\mu\in\mathcal{P}}\mu^i$$

for $i=1,\ldots,\dim\mathcal{P}$ (here $\dim\mathcal{P}=1+2+n(b_0)+n(b_1)+n_T(f)n_S(f)+n(u_0)$ is the number of parameters).
The metric defined in (4.3.10) quantifies proximity of two parameter-time tuples, with appropriate weighting so as to account for scaling differences between the parameters.

– We further define:

$$\sigma_i^{min}=\inf_{v\in X_0,\|v\|=1}c(\zeta_i,v,v),\quad\forall i=1,\ldots,N\quad\sigma_{N+1}^{min}=\inf_{v\in X_0,\|v\|=1}a(v,v)$$

$$\sigma_i^{max}=\sup_{v\in X_0,\|v\|=1}c(\zeta_i,v,v),\quad\forall i=1,\ldots,N\quad\sigma_{N+1}^{max}=\sup_{v\in X_0,\|v\|=1}a(v,v)$$

The SCM lower bound is then given by the following lemma:

Lemma 8 (Proposition 1 in [51]). *For every $\mathcal{C} \subset \{1, \dots, \mathcal{T}\} \times P$ and $M \in \mathbb{N}$, and every $k = 1, \dots, \mathcal{T}$, a lower bound for $C_k(\mu)$ is given by:*

$$C_k^{inf}(\mu) = \inf_{y \in \widetilde{\mathcal{Y}}} [\mathcal{J}(\mu, k, y)] \tag{4.3.11}$$

An algorithm for choosing a constraint subset $\mathcal{C}$ will be given after the description of the SCM upper bound and the SCM offline/online procedure.

SCM upper bound. We define:

$$\widetilde{\mathcal{Y}}^{up} = \{y^*(k_i, \mu_i) \ ; \ i = 1, \dots, I \ ; \ (k_i, \mu_i) \in \mathcal{C}\}$$

where:

$$\mathcal{C} = \{(k_1, \mu_1), (k_2, \mu_2), \dots, (k_I, \mu_I)\}$$

and:

$$y^*(k_i, \mu_i) = \operatorname*{arginf}_{y \in \mathcal{Y}} [\mathcal{J}(\mu_i, k_i, y)] \quad (i = 1, \dots, I)$$

Lemma 9 (Proposition 1 in [51]). *For every $k = 1, \dots, \mathcal{T}$, an upper bound for C_k is given by:*

$$C_k^{sup}(\mu) = \inf_{y \in \widetilde{\mathcal{Y}}^{up}} \mathcal{J}(\mu, k, y) \tag{4.3.12}$$

SCM offline/online procedure. Relying on Lemmas 8 and 9, our offline/online procedure for computing C_k^{inf} and C_k^{sup} reads:

– offline:

1. choose M and constraint set $\mathcal{C}$ (see next paragraph) ;
2. compute and store σ_i^{min} and σ_i^{max} $(i = 1, \ldots, N+1)$ by solving a generalized eigenproblem on X_0 ;
3. for each $(k', \mu') \in \mathcal{C}$:
   (a) solve a generalized eigenproblem on X_0 to find $C_{k'}(\mu')$ (and store it);
   (b) let $w \in X_0$ be a unit eigenvector of the above eigenproblem; compute and store (in $\widetilde{\mathcal{Y}}^{up}$) $y^*(k', \mu')$ using:

$$y^*(k', \mu')_j = c(\zeta_j, w, w) \quad (j = 1, \ldots, N)$$
$$y^*(k', \mu')_{N+1} = a(w, w)$$

– online:
  – for the lower bound:
    1. assemble and solve optimization problem (4.3.11) ;
  – for the upper bound: test one-by-one each element of $\widetilde{\mathcal{Y}}^{up}$ to solve (4.3.12).

Algorithm 4: SCM offline/online

In the lower bound online phase, the optimization problem required to solve is a *linear programming* problem (LP) with $N+1$ variables and $N+1+M$ constraints (M one-sided inequalities and $N+1$ two-sided). There are algorithms, such as the simplex algorithm (see [73] for instance), which solve such optimization problems under (on average) polynomial complexity with respect to the number of variables and number of constraints, even if they can be exponential in the worst cases. What matters here is this complexity is independent of $\mathcal{N}$. The upper bound online phase has a complexity depending linearily on the cardinality of the reasonably-sized $\mathcal{C}$ and on N.

"Greedy" constraint set selection. To choose $\mathcal{C}$ in Algorithm 4, step 1, we can use the greedy constraint set selection Algorithm 5.

1. choose $M \in \mathbb{N}$;
2. initialize $\mathcal{C} = \{(k_1, \mu_1)\}$ with arbitrary $k_1 \in \{1, \dots, \mathcal{T}\}$ and $\mu_1 \in \mathcal{P}$;
3. choose a rather large, finite-sized sample $\Xi \subset \{1, \dots, \mathcal{T}\} \times \mathcal{P}$;
4. repeat:
   - using the "current" $\mathcal{C}$ to compute C^{sup} and C^{inf}, append:

$$(k^*, \mu^*) = \underset{(k,\mu)\in\Xi}{\operatorname{argmax}} \frac{\exp\left(C_k^{sup}(\mu)\right) - \exp\left(C_k^{inf}(\mu)\right)}{\exp\left(C_k^{sup}(\mu)\right)} \qquad \text{to } \mathcal{C}.$$

Algorithm 5: Greedy constraint set selection

The repeat loop in this algorithm can be stopped either when $\#\mathcal{C}$ has reached a maximal value, or when the "relative exponential sharpness" indicator:

$$\max_{(k,\mu)\in\Xi} \frac{\exp\left(C_k^{sup}(\mu)\right) - \exp\left(C_k^{inf}(\mu)\right)}{\exp\left(C_k^{sup}(\mu)\right)}$$

gets less than a desired precision. We use a measure of the difference between the exponentials so as to account for the "exponential" effect of the stability constants on the error bounds [71].

As in the greedy algorithm for basis selection described at Section 4.2.3, this algorithm makes, at each step, the "best possible" choice, that is the value of the parameter and time for which the bounds computed using the current constraint set are the less sharp.

A last remark we can give on the algorithm is about the trade-off in the choice of M: whatever M is, we always get a certified bound on $C_k(\mu)$, but increasing M will improve sharpness of this bound, at the expense of an increase in online computation time.

4.4 Numerical results

We now present some numerical results obtained with the methodology described above. We implemented it in a software package written in C++, using GNU OpenMP [42] as threading library, ARPACK [3] for eigenvalues computation and GLPK [41] as linear programming problems solver.

For all the experiments above, the convergence test for Newton iterations when solving (4.1.13) and (4.2.2b) was the following: $\|\delta\|^2 \leq 3 \times 10^{-16}$. The penalization constant used was $P = 10^7$.

We also took $M = \#\mathcal{C} = 10$ as parameters for the SCM procedure.
The Φ functions appearing in the parametrizations of u_0, b_0, b_1 and f are chosen to be sine functions with fixed known angular velocity. More specifically, we suppose that:

$$b_0(t) = b_{0m} + \sum_{l=1}^{n(b_0)} A_l^{b_0} \sin(\omega_l^{b_0} t) \qquad b_1(t) = b_{1m} + \sum_{l=1}^{n(b_1)} A_l^{b_1} \sin(\omega_l^{b_1} t)$$

$$f(t,x) = f_m + \sum_{l=1}^{n_T(f)} \sum_{p=1}^{n_S(f)} A_{lp}^{f} \sin(\omega_l^{fT} t) \sin(\omega_p^{fS} x) \qquad u_0(x) = u_{0m} + \sum_{l=1}^{n(u_0)} A_l^{u_0} \sin(\omega_l^{u_0} x)$$

4.4.1 Reference solutions

Figure 4.1 shows an example of the reference solution of (4.1.1), (4.1.2) and (4.1.7) every 10 timesteps. The parameters were:

$$\begin{aligned} \mathcal{N} &= 40 & \Delta t &= .02 \\ T &= 2 & \nu &= 1 \\ b_0(t) &= 1 & b_1(t) &= 1.28224 \\ f(t,x) &= 1 & u_0(x) &= 1 + 2\sin(3x) \end{aligned} \tag{4.4.1}$$

Figure 4.2 does the same with a lower viscosity. The parameters were:

$$\begin{aligned} \mathcal{N} &= 40 & \Delta t &= .002 \\ T &= 2 & \nu &= .1 \\ b_0(t) &= 1 & b_1(t) &= 1.28224 \\ f(t,x) &= 1 & u_0(x) &= 1 + 2\sin(3x) \end{aligned} \tag{4.4.2}$$

Figures 4.1 and 4.2 show the solution u of the viscous Burgers' equation plotted as functions of space x, for various times t, respectively for the parameter set (4.4.1) and (4.4.2).
We have checked that the *a posteriori* indicator of boundary error (4.1.8) gets no higher than 6×10^{-7} in both cases.

4.4.2 Reduced solutions

Computational economy

To show the substantial time savings provided by the reduced basis approximation, we compute the reduced solution for the parameters set given by

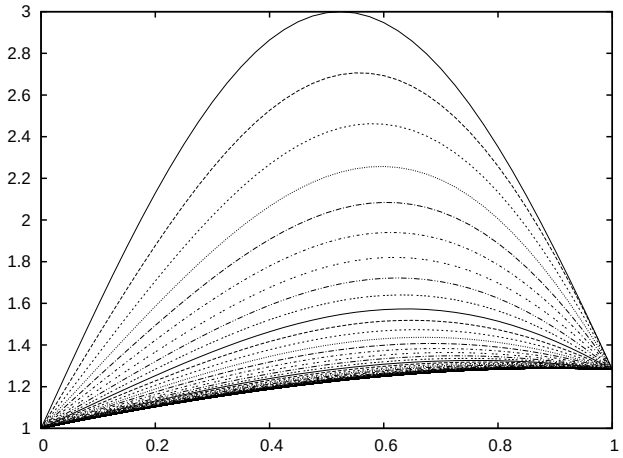


Figure 4.1: Full solution with high viscosity $\nu = 1$. Plots of the solution u of equation (4.1.1), (4.1.2), (4.1.7) as a function of space x, for various times t ranging from $t = 0$ to $t = 2$ (the bottom lines correspond to high times). We use the parameters defined in (4.4.1).

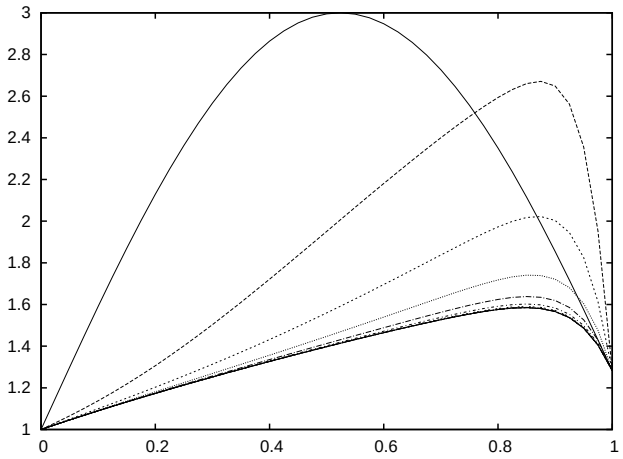


Figure 4.2: Full solution with low viscosity $\nu = .1$. Plots of the solution u of (4.1.1), (4.1.2), (4.1.7) as a function of space x, for various times t ranging from $t = 0$ to $t = 2$ (the bottom lines correspond to high times). We use the parameters defined in (4.4.2).

(4.4.1), with $\mathcal{N} = 60$. The full solution (with $\mathcal{N} = 60$) takes 0.26s CPU time to be produced (when using Thomas' algorithm for tridiagonal matrices inversion).
We use the POD-driven basis selection procedure to select the $N = 7$ leading POD modes (using $S = 30$ snapshots). The resulting basis (with the functions sorted by increasing magnitude of eigenvalues) is shown in Figure 4.3. We did not make use of the "expansion" procedure described in Section 4.2.3. The overall CPU time for the offline phase was 6.36s.
We used fixed parameters $n(b_0) = n(b_1) = n_S(f) = n_T(f) = n(u_0) = 1$ and parameter ranges as shown in Table 4.1.

Parameter	Min.	Max.	Parameter	Min.	Max.
ν	.8	1.2	$A_1^{u_0}$	1.1	3
$A_1^{b_0}$	.9	1.2	$\omega_1^{b_0}$	1	1
$A_1^{b_1}$	.9	1.2	$\omega_1^{b_1}$	1	1
f_m	0	2	ω_1^{fT}	2	2
$A_{1,1}^{f}$	0.7	1.3	ω_1^{fS}	2	2
u_{0m}	0	1	$\omega_1^{u_0}$	3	3

Table 4.1: Ranges of the different parameters in the benchmark of Section 4.4.2.

We then used this basis to compute the reduced solution for a particular (randomly chosen) instance of the parameters. The reduced solution was computed in 0.04s, *including* the time necessary for the online error bound computation, shown in Figure 4.4 (solid line). Our procedure reduces the marginal cost to 15% of the original cost, yet providing a certified L^2 relative error of less than 1‰.

Error bound estimation

Still using the preceding POD basis and instance of the parameters, we compared the online error bound with the actual error, for the same parameter set as above. The result is shown in Figure 4.4. We see that our error bound is quite sharp, especially when it follows the decrease in the actual error near $t = 1.3$. We also checked for the quality of the SCM procedure, by comparing the actual stability constant with the lower bound provided by SCM (Figure 4.5).

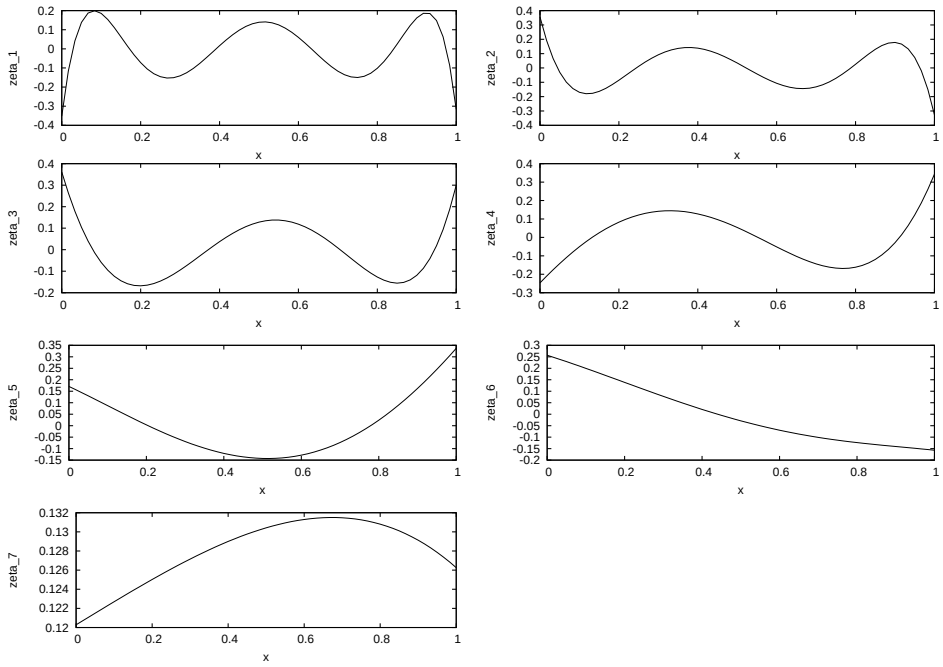


Figure 4.3: POD reduced basis: plots, as functions of space, of the 7 leading POD modes, i.e. the ζ_i ($i = 1, \ldots, 7$) defined by (4.2.6), with z_i ($i = 1, \ldots, 7$) the leading eigenvectors of $M^T \Omega M$. The modes are sorted (from top to bottom, and from left to right) by increasing magnitude of eigenvalues. Parameter ranges for snapshot sampling are those in Table 4.1.

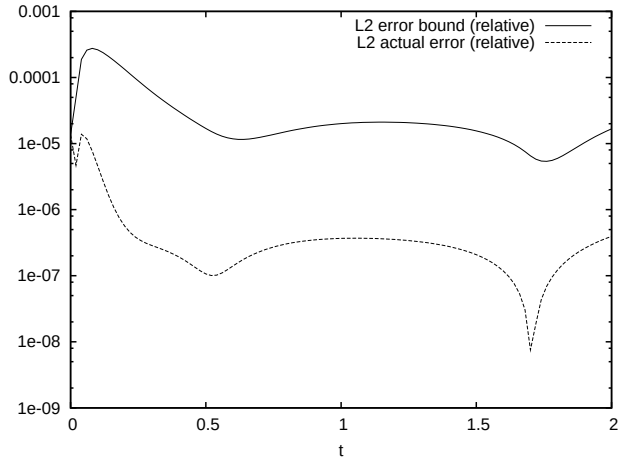


Figure 4.4: Relative L^2 online error bound and actual error. We plot in solid line $\frac{\varepsilon_k}{\|\widetilde{u}^k\|}$, and in dashed line $\frac{\left\|u^k - \widetilde{u}^k\right\|}{\|\widetilde{u}^k\|}$ as functions of $t = k\Delta t$ for $k = 1, \ldots, \mathcal{T}$.

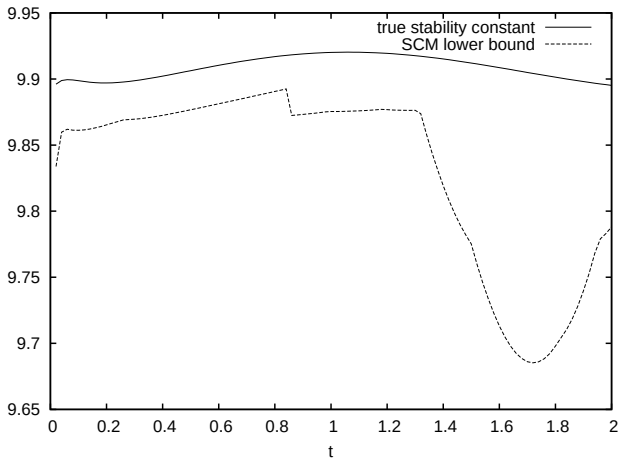


Figure 4.5: True stability constant C_k defined by (4.3.3) and SCM lower bound C_k^{inf} defined by (4.3.11) as functions of time. We use $M = \#\mathcal{C} = 10$ as SCM parameters.

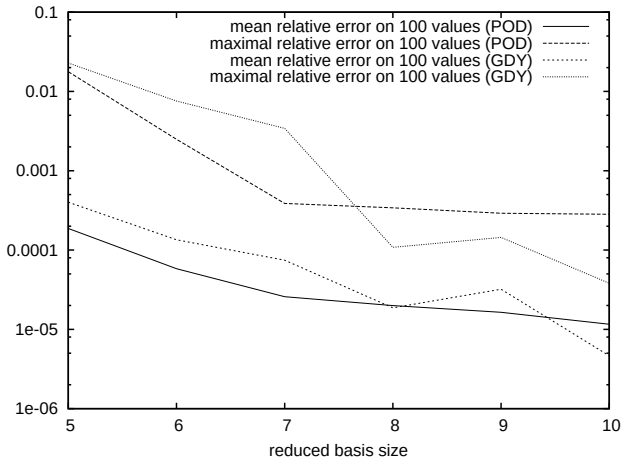


Figure 4.6: Convergence benchmark 1. We plot (on a logarithmic scale) maximal and mean relative online error bounds over a (uniform) random sample of 100 initial values $u_{0m} \in [0, 1]$, as functions of the reduced basis size N, when the reduced basis is chosen using POD-based procedure (POD) or greedy procedure (GDY). Fixed parameters are $\nu = 1$ and $f_m = 1$.

4.4.3 Convergence benchmarks

In order to compare our three basis selection procedures (POD, greedy and POD-Greedy), we have made "convergence benchmarks", i.e. representations of the maximal and mean (estimated) error over all timesteps, and over a sample of 100 parameters as functions of the size of the reduced basis. The same sample of benchmark parameters is used throughout all the procedure.

Comparison of greedy (with $\#\Xi = 100$) and POD (with $S = 60$) procedures for $\mathcal{N} = 40$, $T = 2$, $\Delta t = .02$, $n(b_0) = n(b_1) = n_S(f) = n_T(f) = n(u_0) = 0$, with parameters f_m and ν fixed to unity, and varying $u_{0m} \in [0, 1]$ (and thus initial boundary values b_{0m} and b_{1m}, moving accordingly to compatibility conditions (4.1.4)) is shown in Figure 4.6.

The benchmarking process for greedy took 19.5s of CPU time, the one for POD took 14.98s. The online cost, depending only on the size of the reduced basis, is the same regardless of how the reduced basis has been chosen. We also see the fast (exponential) convergence of error bound towards zero as N increases.

Another benchmark was then made, with the same data, except that $\nu = .1$ and $\Delta t = .002$. The result is visualized in Figure 4.7. The POD benchmarking process took 349 s of CPU time, the POD-Greedy, 282 s, and the Greedy, 470 s. We notice that a smaller viscosity leads to degraded precision of our RB approximation. The resulting final basis selected by the POD-Greedy is displayed in Figure 4.8.

The POD-Greedy algorithm is run using $P_1 = 2$ and initialized using the full time-discrete trajectory (hence, 200 vectors) for one parameter value. Using the initial data as initialization for the POD-Greedy algorithm and using $P_1 = 1$ may give better overall performance of POD-Greedy (at the expense of an increased offline computation time), as the sample of error bounds is updated at every step of the algorithm (instead of every other step for $P_1 = 2$).

4.4.4 Effect of mesh refinement and penalization constant

To check for the robustness of our bound, we studied the influence of $\mathcal{N}$ (the number of "full" spatial discretization points) and P on the magnitude and the sharpness of our error bound. We ran the same benchmark as in Section 4.4.2, but with different $\mathcal{N}$ or P.

We did the test with refined meshes ($\mathcal{N} = 200$, $\mathcal{N} = 800$) and we obtained a similar error bound profile; we conclude that the sharpness of our error bound is quite insensitive to mesh refinement.

In Figure 4.9, we visualize the actual error and the error bound for various values of P (with $\mathcal{N} = 60$). We see that the error bound is tighter and sharper for high values of the penalization constant P. This can easily be explained by the fact that, as $P \to +\infty$, the errors at the boundary $e_k(0)$ and $e_k(1)$ vanish, hence modifying all the terms in the error bound (i.e., decreasing η_k, σ_k, f_k, $\xi_k^{\mathcal{A}}$, $\xi_k^{\mathcal{B}}$, ξ_k^{γ}, $\mathcal{B}_k$, γ_k and increasing $\mathcal{A}_k$) where these errors appear.

4.4.5 Comparison with existing bound

We compared our error bound with the bound described in [71], for $\mathcal{N} = 60$, $\Delta t = .02$, $T = 2$, fixed $u_0 = b_0 = b_1 = 0$, $f = 1$ and variable $\nu \in [0.1; 1]$. The reduced basis is found by POD with $S = 90$. Figure 4.10 shows that our bound is clearly better than the existing reference bound.

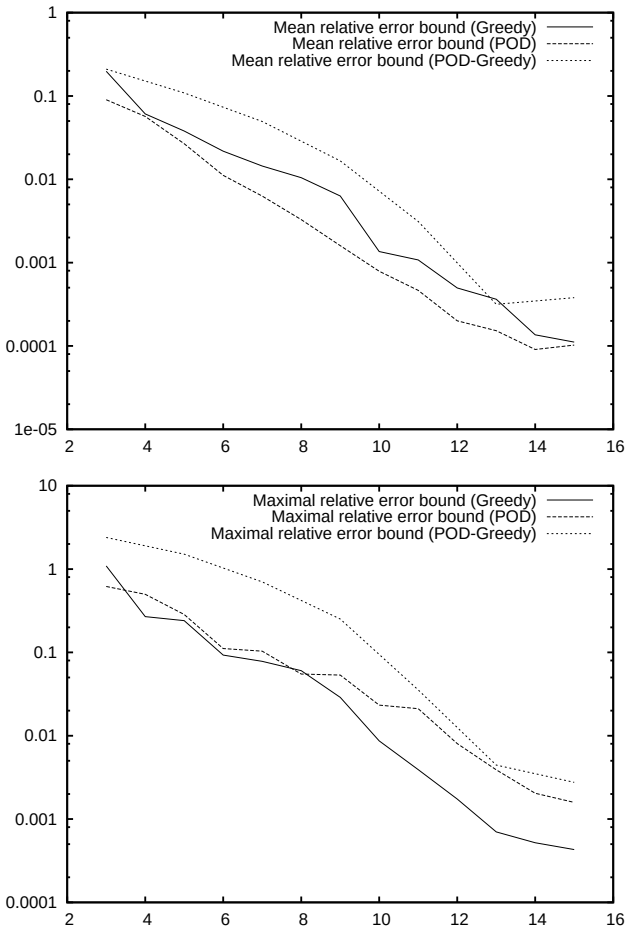


Figure 4.7: Convergence benchmark 2. We plot (on a logarithmic scale) maximal (bottom) and mean (top) online error bounds over a (uniform) random sample of 100 initial values $u_{0m} \in [0, 1]$, as functions of the reduced basis size N, when reduced basis is chosen using POD-based procedure with $S = 90$, Greedy, or POD-Greedy procedure with $P_1 = 2$. Fixed parameters are $\nu = 0.1$ and $f_m = 1$.

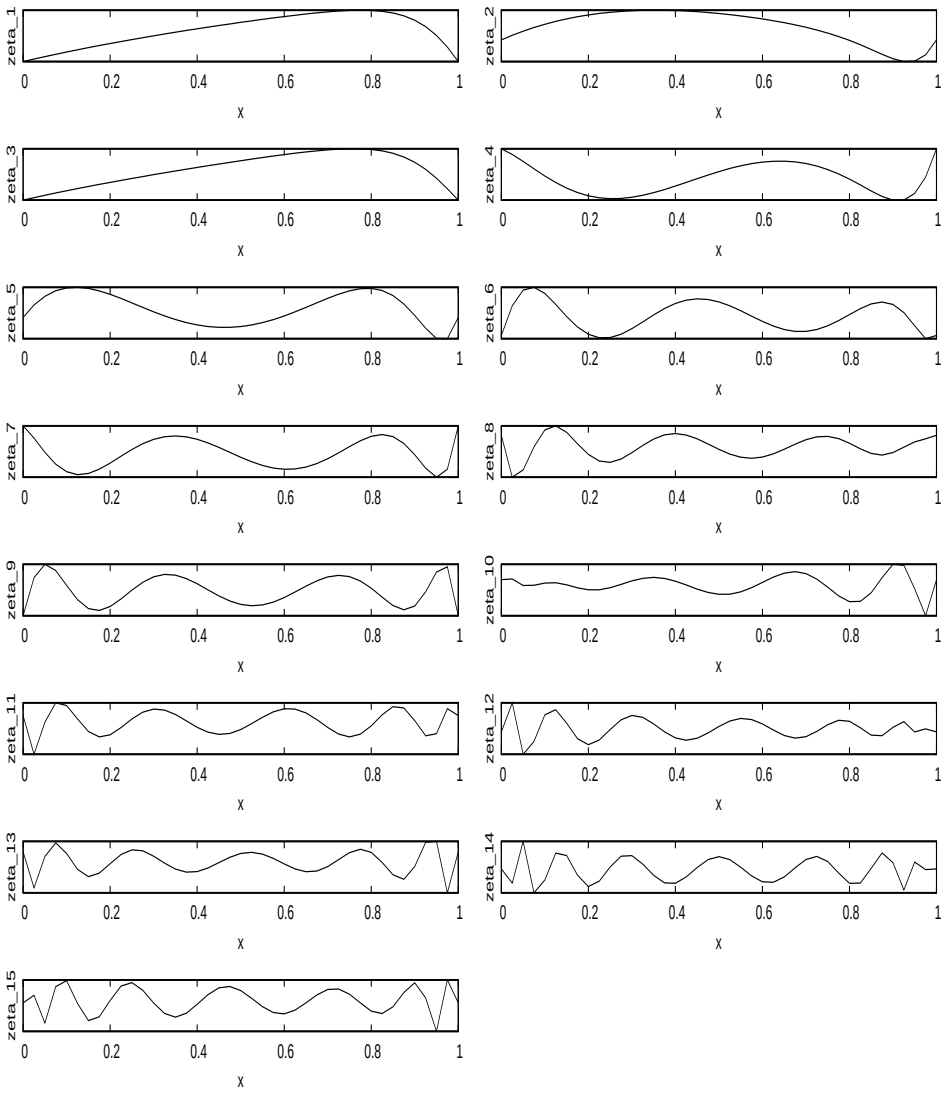


Figure 4.8: Reduced basis selected at the final step of the POD-Greedy procedure carried out for the previous benchmark (Figure 4.7). Basis elements are plotted as functions of space, and are sorted (from top to bottom, left to right) in order of selection in the greedy procedure.

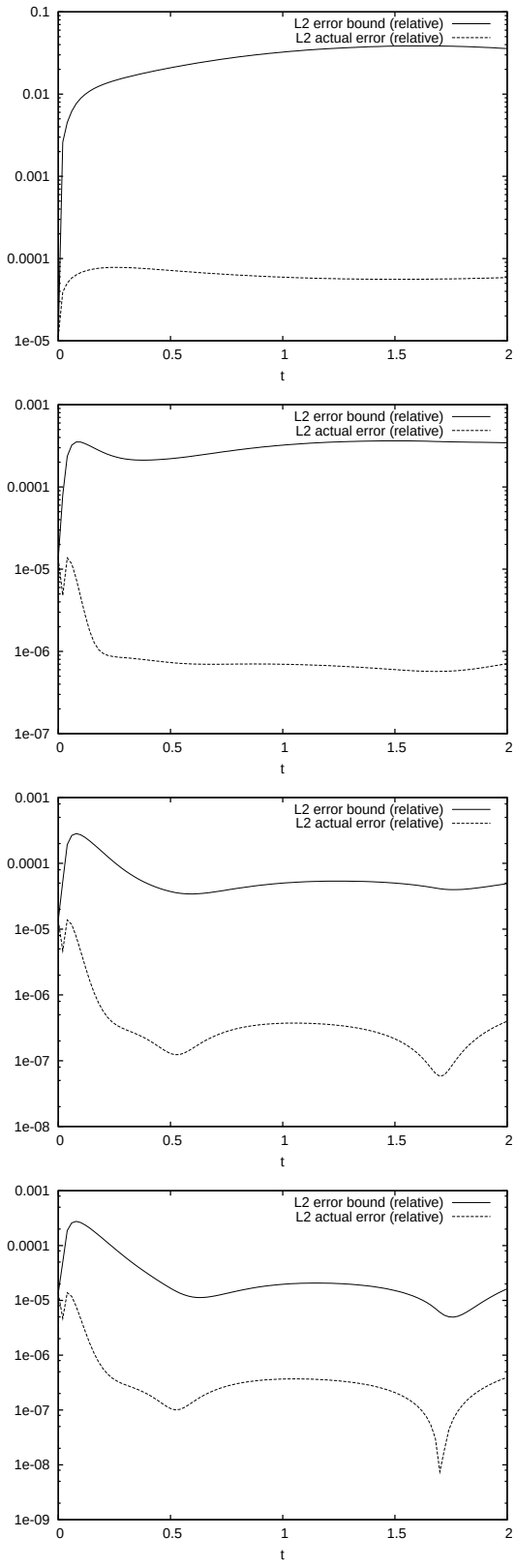


Figure 4.9: Relative L^2 online error bound and actual error. We plot in solid line $\dfrac{\varepsilon_k}{\|\widetilde{u}^k\|}$, and in dashed line $\dfrac{\left\|u^k-\widetilde{u}^k\right\|}{\|\widetilde{u}^k\|}$ as functions of $t=k\Delta t$ for $k=1,\ldots,\mathcal{T}$, and $\mathcal{N}=60$ and, from left to right and top to bottom, $P=10^2$, $P=10^4$, $P=10^5$ and $P=10^{12}$.

Besides, we have performed a benchmark over a fixed sample of 100 random values of ν uniformly chosen in $[0.1; 1]$. For our bound, the mean error bound is 0.00076, the maximum error bound is 0.02, while for the bound of [71], we obtain 0.0041 for the mean bound and 0.25 for the maximum.
It is easy to see that, when the boundary conditions are fixed to zero, the recurrence formula for the error bound reduces to:

$$\|e_k\| \leq \frac{\|e_{k-1}\| + \Delta t \, \|r_k\|_0}{1 + C_k \Delta t}$$

while the error bound described in [71] has the following expression:

$$\|e_k\| \leq \sqrt{\frac{\|e_{k-1}\|^2 + \frac{\Delta t}{\nu} \|r_k\|_0^2}{1 + \widetilde{C_k} \Delta t}}$$

where the modified stability constant $\widetilde{C_k}$ reads:

$$\widetilde{C_k} = \inf_{v \in X_0} \frac{4c(\widetilde{u}^k, v, v) + \nu a(v, v)}{\|v\|^2}$$

We recall that our stability constant C_k is given by:

$$C_k = \inf_{v \in X_0, \|v\|=1} \left[2c(\widetilde{u}^k, v, v) + \nu a(v, v) \right]$$

The ν-dependence of the bound can explain why our bound is better, especially for small values of ν. Besides, the derivation of our error bound makes lesser use of inequalities (for instance, we do not make use of Young's inequality at the beginning of the proof, each inequality used is a potential source of optimality loss) and keeps treating more terms.

Conclusion

We have presented a certified procedure for low marginal cost approximate resolution of the viscous Burgers equation with parametrized viscosity, as well as initial and boundary value data. This procedure makes use of a reduced basis offline/online procedure for a penalized weak formulation, an efficiently computed error bound in natural L^2 norm (made possible by the successive constraints method (SCM)), and three procedures at hand for choosing a basis to expand reduced solutions in.
Our procedure becomes less useful when the ratio time/viscosity increases, as this degradates the stability constant C_k. Another limitation of our

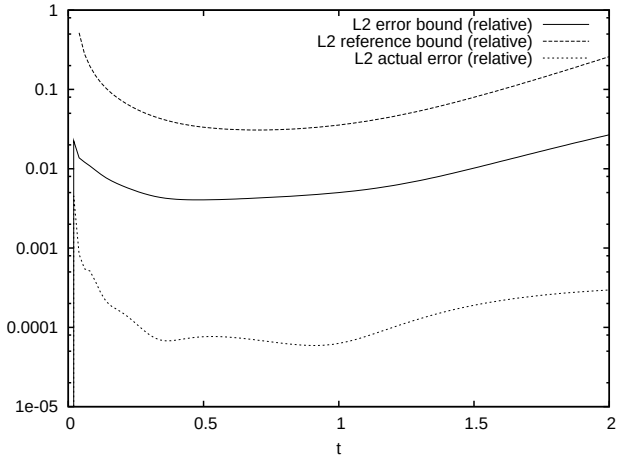


Figure 4.10: Comparison with the existing error bound for reduced basis Burgers equation (reference bound). We plot the actual error, the existing (reference) error bound and our error bound as functions of time. We took $\nu = 0.1$.

method, for one willing to use it for large times, is that the online procedure complexity still depends on the temporal discretization step. However, our numerical experiments show a substantial decrease in marginal cost when using reduced basis approximation, as well as efficiency (both in terms of sharpness and computation time) of the provided error bound for moderate viscosities. This decrease in the cost is made possible by the fact that online procedure has a complexity that is independent from the number of spatial discretization points.

4.5 Appendix – Proof of Theorem 4

Proof. Subtracting left-hand side of (4.2.2b) from both sides of relation (4.3.1) yields that for $k = 1, \ldots, \mathcal{T}$, the error at time t_k: $e_k = u_e^k - \widetilde{u}^k$ satisfies, for every $v \in X_0$:

$$\frac{1}{\Delta t}(\langle e_k, v\rangle - \langle e_{k-1}, v\rangle) + c(u_e^k, u_e^k, v) - c(\widetilde{u}^k, \widetilde{u}^k, v) + \nu a(e_k, v) = r_k(v) \quad (4.5.1)$$

We write:

$$e_k = e_k(0)\phi_0 + e_k(1)\phi_{\mathcal{N}} + e_k^z \quad (4.5.2)$$

with $e_k^z \in X_0 = \{v \in X \text{ st. } v(0) = v(1) = 0\}$. And, by applying (4.5.1) with $v = e_k^z$:

$$\frac{1}{\Delta t}(\langle e_k, e_k^z\rangle - \langle e_{k-1}, e_k^z\rangle) + c(u_e^k, u_e^k, e_k^z) - c(\widetilde{u}^k, \widetilde{u}^k, e_k^z) + \nu a(e_k, e_k^z) = r_k(e_k^z) \tag{4.5.3}$$

Since:

$$\begin{aligned}\langle e_k, e_k^z\rangle &= e_k(0)\langle \phi_0, e_k^z\rangle + e_k(1)\langle \phi_{\mathcal{N}}, e_k^z\rangle + \|e_k^z\|^2 \\ &= \|e_k^z\|^2 + e_k(0)e_k^z\left(\frac{1}{\mathcal{N}}\right)\langle \phi_0, \phi_1\rangle + e_k(1)e_k^z\left(1 - \frac{1}{\mathcal{N}}\right)\langle \phi_{\mathcal{N}-1}, \phi_{\mathcal{N}}\rangle\end{aligned}$$

and that, for every $v \in X$:

$$\begin{aligned}c(u_e^k, u_e^k, v) - c(\widetilde{u}^k, \widetilde{u}^k, v) &= -\frac{1}{2}\int_0^1 \left((u_e^k)^2 - (\widetilde{u}^k)^2\right)\frac{\partial v}{\partial x} \\ &= -\frac{1}{2}\int_0^1 \left(u_e^k - \widetilde{u}^k\right)\left(u_e^k + \widetilde{u}^k\right)\frac{\partial v}{\partial x} \\ &= -\frac{1}{2}\int_0^1 e_k\left(\widetilde{u}^k + \widetilde{u}^k + e_k\right)\frac{\partial v}{\partial x} \\ &= 2c(\widetilde{u}^k, e_k, v) + c(e_k, e_k, v)\end{aligned}$$

We have that (4.5.3) implies:

$$\begin{aligned}\frac{1}{\Delta t}(\|e_k^z\|^2 - \langle e_{k-1}, e_k^z\rangle) + \psi_k(e_k, e_k^z) + \frac{1}{\Delta t}\left(e_k(0)e_k^z\left(\frac{1}{\mathcal{N}}\right)\langle \phi_0, \phi_1\rangle + e_k(1)e_k^z\left(1 - \frac{1}{\mathcal{N}}\right)\langle \phi_{\mathcal{N}}, \phi_{\mathcal{N}-1}\rangle\right) \\ = r_k(e_k^z) - c(e_k, e_k, e_k^z) \quad (4.5.4)\end{aligned}$$

We are now willing to find a lower bound for the left-hand side of (4.5.4) and an upper bound for its right-hand side.

Lower bound for LHS. From Cauchy-Schwarz inequality:

$$-\langle e_{k-1}, e_k^z\rangle \quad \geq \quad -\|e_{k-1}\|\,\|e_k^z\|$$

and, by the triangle inequality:

$$\|e_k\| - \eta_k \leq \|e_k^z\| \leq \|e_k\| + \eta_k \tag{4.5.5}$$

because $\eta_k = |e_k(0)|\,\|\phi_0\| + |e_k(1)|\,\|\phi_{\mathcal{N}}\|$.

So:

$$-\langle e_{k-1}, e_k^z\rangle \quad \geq \quad -\|e_{k-1}\|\,(\|e_k\| + \eta_k) \tag{4.5.6}$$

We also have:

$$\|e_k^z\|^2 \geq \|e_k\|^2 + \|e_k - e_k^z\|^2 - 2\,\|e_k\|\,\|e_k - e_k^z\| \geq \|e_k\|^2 - 2\,\|e_k\|\,\eta_k \tag{4.5.7}$$

because of Cauchy-Schwarz and triangle inequalities.
Besides,

$$0 \leq \|e_k^z\| \leq \|e_k^z - e_k\| + \|e_k\| \leq \eta_k + \|e_k\|$$

so that:

$$\|e_k^z\|^2 \leq (\|e_k\| + \eta_k)^2 \tag{4.5.8}$$

Using the bilinearity of ψ_k, we have:

$$\psi_k(e_k, e_k^z) = \psi_k(e_k^z, e_k^z) + e_k(0)\psi_k(\phi_0, \phi_1)e_k^z\left(\frac{1}{\mathcal{N}}\right) + e_k(1)\psi_k(\phi_{\mathcal{N}}, \phi_{\mathcal{N}-1})e_k^z\left(1 - \frac{1}{\mathcal{N}}\right) \tag{4.5.9}$$

because $\psi_k(\phi_0, \phi_{\mathcal{N}}) = \psi_k(\phi_{\mathcal{N}}, \phi_0) = 0$, since ϕ_0 and $\phi_{\mathcal{N}}$ have no common support, $\psi_k(\phi_0, \phi_j) = 0$ for $j > 1$, and $\psi_k(\phi_{\mathcal{N}}, \phi_j) = 0$ for $j < \mathcal{N} - 1$.
From the definition of the stability constant C_k:

$$\psi_k(e_k^z, e_k^z) \geq C_k \|e_k^z\|^2$$

So that, thanks to (4.5.7) and (4.5.8),

$$\psi_k(e_k^z, e_k^z) \geq \begin{cases} C_k \|e_k\|^2 - 2\eta_k C_k \|e_k\| & \text{if } C_k \geq 0 \\ C_k \|e_k\|^2 + 2\eta_k C_k \|e_k\| + C_k \eta_k^2 & \text{if } C_k \leq 0 \end{cases}$$

That is

$$\psi_k(e_k^z, e_k^z) \geq C_k \|e_k\|^2 - \sigma_k \|e_k\| - [C_k]_- \eta_k^2 \tag{4.5.10}$$

from the definition of σ_k.
We have:

$$\left|e_k^z\left(\frac{1}{\mathcal{N}}\right)\right| \leq \mathcal{E}\|e_k^z\| \leq \mathcal{E}\|e_k\| + \mathcal{E}\eta_k \tag{4.5.11}$$

and by symmetry:

$$\left|e_k^z\left(1 - \frac{1}{\mathcal{N}}\right)\right| \leq \mathcal{E}\|e_k^z\| \leq \mathcal{E}\|e_k\| + \mathcal{E}\eta_k \tag{4.5.12}$$

so that, combining (4.5.11), (4.5.12) and introducing f_k:

$$\left|e_k(0)\psi_k(\phi_0, \phi_1)e_k^z\left(\frac{1}{\mathcal{N}}\right) + e_k(1)\psi_k(\phi_{\mathcal{N}}, \phi_{\mathcal{N}-1})e_k^z\left(1 - \frac{1}{\mathcal{N}}\right)\right| \leq \|e_k\| f_k + \eta_k f_k$$

Now we can say that:

$$e_k(0)\psi_k(\phi_0, \phi_1)e_k^z\left(\frac{1}{\mathcal{N}}\right) + e_k(1)\psi_k(\phi_{\mathcal{N}}, \phi_{\mathcal{N}-1})e_k^z\left(1 - \frac{1}{\mathcal{N}}\right) \geq -\|e_k\| f_k - \eta_k f_k \tag{4.5.13}$$

And we also have:

$$\begin{aligned}
&\left|e_k(0)e_k^z\left(\frac{1}{\mathcal{N}}\right)\langle\phi_0,\phi_1\rangle + e_k(1)e_k^z\left(1-\frac{1}{\mathcal{N}}\right)\langle\phi_{\mathcal{N}-1},\phi_{\mathcal{N}}\rangle\right| \\
&\leq |e_k(0)|\left(\mathcal{E}\|e_k\|+\mathcal{E}\eta_k\right)\langle\phi_0,\phi_1\rangle + |e_k(1)|\left(\mathcal{E}\|e_k\|+\mathcal{E}\eta_k\right)\langle\phi_{\mathcal{N}},\phi_{\mathcal{N}-1}\rangle \\
&= \mathcal{E}(|e_k(0)|\langle\phi_0,\phi_1\rangle + |e_k(1)|\langle\phi_{\mathcal{N}},\phi_{\mathcal{N}-1}\rangle)\|e_k\| + \mathcal{E}\eta_k(|e_k(0)|\langle\phi_0,\phi_1\rangle + |e_k(1)|\langle\phi_{\mathcal{N}},\phi_{\mathcal{N}-1}\rangle) \\
&= \mathcal{E}\langle\phi_0,\phi_1\rangle(|e_k(0)|+|e_k(1)|)\|e_k\| + \mathcal{E}\eta_k\langle\phi_0,\phi_1\rangle(|e_k(0)|+|e_k(1)|) \quad (4.5.14)
\end{aligned}$$

since $\langle\phi_0,\phi_1\rangle = \langle\phi_{\mathcal{N}},\phi_{\mathcal{N}-1}\rangle$ by symmetry.
Thus, thanks to (4.5.7), (4.5.6), (4.5.9), (4.5.10), (4.5.13) and (4.5.14), the left-hand side of (4.5.4) is greater than:

$$\begin{aligned}
&\left(\frac{1}{\Delta t}+C_k\right)\|e_k\|^2 - \left(\frac{2\eta_k+\|e_{k-1}\|+\mathcal{E}\langle\phi_0,\phi_1\rangle(|e_k(0)|+|e_k(1)|)}{\Delta t}+\sigma_k+f_k\right)\|e_k\| \\
&- \frac{\eta_k\|e_{k-1}\|+\mathcal{E}\eta_k\langle\phi_0,\phi_1\rangle(|e_k(0)|+|e_k(1)|)}{\Delta t} - [C_k]_-\,\eta_k^2 - \eta_k f_k \quad (4.5.15)
\end{aligned}$$

Upper bound for RHS. We have:

$$c(e_k,e_k,e_k^z) = c(e_k,e_k,e_k) - e_k(0)c(e_k,e_k,\phi_0) - e_k(1)c(e_k,e_k,\phi_{\mathcal{N}})$$

but:

$$\begin{aligned}
c(e_k,e_k,e_k) &= -\frac{1}{2}\int_0^1 e_k^2\frac{\partial e_k}{\partial x} \\
&= -\frac{1}{6}\int_0^1 \frac{\partial\left[(e_k)^3\right]}{\partial x} \\
&= -\frac{1}{6}\left((e_k(1))^3-(e_k(0))^3\right)
\end{aligned}$$

So:

$$\begin{aligned}
c(e_k,e_k,e_k^z) &= -\frac{1}{6}\left((e_k(1))^3-(e_k(0))^3\right) - \left(e_k(0)\int_0^1 e_k^2\frac{\partial\phi_0}{\partial x} + e_k(1)\int_0^1 e_k^2\frac{\partial\phi_{\mathcal{N}}}{\partial x}\right) \\
&= -\frac{1}{6}\left((e_k(1))^3-(e_k(0))^3\right) + \mathcal{N}\left(e_k(0)\int_0^{1/\mathcal{N}} e_k^2 - e_k(1)\int_{1-1/\mathcal{N}}^1 e_k^2\right) \quad (4.5.16)
\end{aligned}$$

Since, for all $x\in\left[0;\frac{1}{\mathcal{N}}\right]$,

$$e_k(x) = e_k(0) + \mathcal{N}\left(e_k\left(\frac{1}{\mathcal{N}}\right)-e_k(0)\right)x$$

we have, thanks to $\left|e_k\left(\frac{1}{\mathcal{N}}\right)\right| \leq \mathcal{E}\,\|e_k\|$:

$$\begin{aligned}\left|\mathcal{N}\times e_k(0)\int_0^{1/\mathcal{N}} e_k^2\right| &\leq \mathcal{N}\frac{|e_k(0)|}{\mathcal{N}}\\ &\quad\times\left(\left|e_k\left(\frac{1}{\mathcal{N}}\right)\right||e_k(0)| + \frac{e_k\left(\frac{1}{\mathcal{N}}\right)^2 + e_k(0)^2}{3} + \frac{2\left|e_k\left(\frac{1}{\mathcal{N}}\right)e_k(0)\right|}{3}\right)\\ &\leq |e_k(0)|\left(\mathcal{E}\,\|e_k\|\,|e_k(0)| + \frac{\mathcal{E}^2\,\|e_k\|^2}{3} + \frac{e_k(0)^2}{3} + \frac{2\,|e_k(0)|}{3}\mathcal{E}\,\|e_k\|\right)\\ &\leq \frac{\mathcal{E}^2\,|e_k(0)|}{3}\,\|e_k\|^2 + \frac{5}{3}\,|e_k(0)|^2\,\mathcal{E}\,\|e_k\| + \frac{|e_k(0)|^3}{3}\end{aligned}$$

As a similar computation can be worked out for $\left|\mathcal{N}\times e_k(1)\int_{1-1/\mathcal{N}}^1 e_k^2\right|$, we have:

$$-\left(e_k(0)\int_0^1 e_k^2\frac{\partial\phi_0}{\partial x} + e_k(1)\int_0^1 e_k^2\frac{\partial\phi_{\mathcal{N}}}{\partial x}\right) \leq \xi_k^{\mathcal{A}}\,\|e_k\|^2 + \xi_k^{\mathcal{B}}\,\|e_k\| + \xi_k^{\gamma}$$

We also have, thanks to (4.5.5):

$$|r_k(e_k^z)| \leq \|r_k\|_0\,\|e_k^z\| \leq \|r_k\|_0\,\|e_k\| + \|r_k\|_0\,\eta_k$$

where:

$$\|r_k\|_0 = \sup_{v\in X_0,\|v\|=1} r_k(v)$$

Hence, the right-hand side of (4.5.4) is less than:

$$\xi_k^{\mathcal{A}}\,\|e_k\|^2 + (\|r_k\|_0 + \xi_k^{\mathcal{B}})\,\|e_k\| + \frac{1}{6}\left|e_k(1)^3 - e_k(0)^3\right| + \xi_k^{\gamma} + \|r_k\|_0\,\eta_k$$

Conclusion. Now (4.5.4) implies, thanks to (4.5.15), and (4.5.16):

$$\mathcal{A}_k\,\|e_k\|^2 - \mathcal{B}_k\,\|e_k\| - \gamma_k \leq 0 \tag{4.5.17}$$

Viewing left-hand side of (4.5.17) as a (convex, thanks to our hypothesis (4.3.4)) quadratic function Q of $\|e_k\|$, whose discriminant is $\mathcal{D}_k$, equation (4.5.17) implies that, if $\mathcal{D}_k \geq 0$, $\|e_k\|$ is smaller than the greatest real root of Q, that is:

$$\|e_k\| \leq \frac{\mathcal{B}_k + \sqrt{\mathcal{D}_k}}{2\mathcal{A}_k}$$

If $\mathcal{D}_k < 0$, then necessarily $\gamma_k < 0$ (as $\mathcal{A}_k$ is positive). Hence (4.5.17) implies:

$$\mathcal{A}_k\,\|e_k\|^2 - \mathcal{B}_k\,\|e_k\| \leq 0$$

and so:

$$\|e_k\| \leq \frac{\mathcal{B}_k}{\mathcal{A}_k}. \qquad \square$$

Acknowledgements: We would like to thank the anonymous referees, whose careful reading and comments have helped to greatly improve this paper. This work has been partially supported by the French National Research Agency (ANR) through COSINUS program (project COSTA-BRAVA n° ANR-09-COSI-015).

Chapter 5

Goal-oriented error estimation for reduced basis method, with application to certified sensitivity analysis

Résumé: La méthode base réduite est une puissante technique de réduction de modèle, conçue pour accélérer le calcul d'un grand nombre de solutions numériques à des équations aux dérivées partielles (EDP) paramétrées. Nous considérons le calcul d'une quantité d'intérêt, qui est une fonctionnelle linéaire de la solution de l'EDP paramétrée. Comparée à la quantité d'intérêt originale, la quantité d'intérêt calculée en utilisant le modèle réduit est entâchée d'une erreur de réduction. Nous présentons une nouvelle borne, efficace et explicitement calculable, pour cette erreur. Sur différents exemples, nous montrons que cette borne d'erreur est plus précise que celles déjà existantes. Nous présentons également une application de ce travail aux études certifiées d'analyse de sensibilité.

Abstract: The reduced basis method is a powerful model reduction technique designed to speed up the computation of multiple numerical solutions of parameterized partial differential equations (PDEs). We consider a quantity of interest, which is a linear functional of the parameterized PDE solution. Compared to the original quantity of interest, the quantity of interest computed using the reduced model is tainted by a reduction error. We

present a new, efficiently and explicitly computable bound for this error, and we show on different examples that this error bound is more precise than the existing ones. We also present an application of our work to certified sensitivity analysis studies.

Introduction

A large number of mathematical models are based on partial differential equations (PDEs). These models require input data (e.g., the physical features of the considered system, the geometry of the domain, the external forces...) which enter in the PDE as *parameters*. In many applications (for instance, design optimization, data assimilation, or uncertainty quantification), one has to numerically compute the solution of a parametrized partial differential equation for a large number of values of the parameters. In such a case, it is generally interesting, in terms of computation time, to perform all possible parameter-independent computations in an *offline* phase, which is done only once, and to call an *online* phase for each required value of the parameter, during which the information gathered in the offline phase can be used to speed-up the computation of an approximate solution of the PDE, and, hence, to reduce the marginal (ie., per parameter) computation cost.

The reduced basis method ([72]) is a way of specifying such offline and online phases, which has been successfully applied to various well-known PDEs ([45, 61, 110]). One should note that, in the reduced basis (RB) method, the online phase does not compute a solution which is strictly identical to the numerical PDE solution, but an approximation of it, obtained by projecting the original discretized equations onto a well-chosen basis. In the application cases given above, however, one is not interested in the solution by itself, but rather in a *quantity of interest*, or model *output*, which is a functional of this solution. Taking this functional into account when performing the model reduction leads to a so-called *goal-oriented* method. For instance, goal-oriented basis choice procedures have been tried with success in the context of dynamical systems in [115, 53], where the basis is chosen so as to contain the modes that are relevant to accurately represent the output of interest, and in a general context in [13], where the basis is chosen so as to minimize the overall output error. All those papers showed that using an adapted basis could lead to a great improvement of reduction error. This paper is about goal-oriented error estimation, that is, the description of a rigorous and computable *error bound* between the model output and the reduced one. In [72], the output error bounds are computed by using an adjoint-based method, which involves the application of the RB method to an auxiliary

(dual) problem in order to correct the output and to compute a goal-oriented error bound. This method, however, has the drawback of doubling offline *and online* computational times. In this paper, we propose a new goal-oriented error bound which does not require a doubled online computation time, and we show, in numerical examples, that this method, for a fixed computational budget, outperforms the adjoint-based error bound.
This paper is organized as follows: in the first part, we describe our output error bound and give its method of computation; in the second part, we see how to apply our error bound to certified sensitivity analysis studies; finally, the third and fourth parts present the numerical applications of our method.

5.1 Methodology

5.1.1 Preliminaries

We begin by setting up the context of the reduced basis method for affine-parameterized linear partial differential equations presented in [72]. Our reference problem is the following: given a parameter tuple $\mu \in \mathcal{P} \subset \mathbb{R}^p$, find $u(\mu)$, the solution (in a discretized functional space X of finite dimension) of:

$$A(\mu)u(\mu) = f(\mu), \tag{5.1.1}$$

where $A(\mu)$ is a μ-dependent invertible square matrix of dimension $\dim X$, and $f \in X$; then compute the *output*:

$$s(\mu) = s(u(\mu)) \tag{5.1.2}$$

where $s : X \to \mathbb{R}$ is a linear form on X.
Problems such as (5.1.1) usually appear as discretizations of μ-parametrized linear partial differential equations (PDE); the X space is typically a finite element subspace (for instance, Lagrange P^1 finite elements), and $A(\mu)$ and f are given by Galerkin projection of the weak form of the PDE onto a suitable basis of X. The boundary conditions of the PDE are usually either encoded in X or in $A(\mu)$.
The dimension of the finite element subspace $\dim X$ is generally fairly large, so that the numerical computation of $u(\mu)$ from the inversion of $A(\mu)$ is expensive. The reduced basis aims at speeding up "many queries", that is, the computation of $u(\mu)$ for all parameters $\mu \in \mathcal{P}_0$ where $\mathcal{P}_0$ is a finite but

"large" subset of the parameter set $\mathcal{P}$. We suppose that $A(\mu)$ and $f(\mu)$ admit the following so-called affine decomposition [72]:

$$\forall \mu \in \mathcal{P}, \quad A(\mu) = \sum_{q=1}^{Q} \Theta_q(\mu) A_q, \quad f(\mu) = \sum_{q'=1}^{Q'} \gamma_{q'}(\mu) f'_q \tag{5.1.3}$$

where $Q, Q' \in \mathbb{N}^*$, $\Theta_q : \mathcal{P} \to \mathbb{R}$ and $\gamma_{q'} : \mathcal{P} \to \mathbb{R}$ (for $q = 1, \ldots, Q$, $q' = 1, \ldots, Q'$) are smooth functions, A_q are square matrices of dimension $\dim X$ and $f_{q'} \in X$.

We suppose that X is endowed with the standard Euclidean inner product: $\langle u, v \rangle = u^t v$, with associated norm $\|u\| = \sqrt{\langle u, u \rangle}$, and consider a subspace $\widetilde{X}$ of X, and a matrix Z whose columns are the components of a basis of $\widetilde{X}$ in a basis of X. This basis of $\widetilde{X}$ is called the *reduced basis* in the sequel. We denote by $\widetilde{u}(\mu)$ the components, in the reduced basis, of the solution of the Galerkin projection of (5.1.1) onto $\widetilde{X}$, that is, the solution of:

$$Z^t A Z \widetilde{u}(\mu) = Z^t f(\mu) \tag{5.1.4}$$

(where, for any matrix M, M^t is the transpose of M).

The many-query computation can then be split into two parts: the first part (usually called the "offline phase"), which is done only once, begins by finding a reduced subspace and its basis (this gives the Z matrix), then the Q parameter-independent matrices:

$$\widetilde{A}_q = Z^t A_q Z, \quad q = 1, \ldots, Q$$

and the Q' vectors:

$$\widetilde{f}_{q'} = Z^t f_{q'}, \quad q' = 1, \ldots, Q'$$

are computed and stored. In the second part (the "online phase"), we compute, for each value of the parameter μ:

$$\widetilde{A}(\mu) = \sum_{q=1}^{Q} \Theta_q(\mu) \widetilde{A}_q, \quad \widetilde{f}(\mu) = \sum_{q'=1}^{Q'} \gamma_q(\mu) \widetilde{f}_{q'} \tag{5.1.5}$$

and solve for $\widetilde{u}(\mu)$ satisfying:

$$\widetilde{A}(\mu) \widetilde{u}(\mu) = \widetilde{f}(\mu). \tag{5.1.6}$$

The key point is that the operations in (5.1.5) and (5.1.6) are performed on vectors and matrices of size $\dim \widetilde{X}$, and that the complexity of these operations is totally independent from the dimension of the underlying "truth"

subspace X. In many cases, the smoothness of the map $\mu \mapsto u(\mu)$ allows to find (in a constructive way, ie., compute) $\widetilde{X}$ so that $\dim \widetilde{X} \ll \dim X$ while keeping $\|u(\mu) - Z\widetilde{u}(\mu)\|$ small, hence enabling significant computational savings.

The output $s(\mu)$ can also be approximated from $\widetilde{u}(\mu)$ using an efficient offline-online procedure: let $l \in X$ be so that:

$$s(u) = \langle l, u \rangle \quad \forall u \in X;$$

in the offline phase we compute and store:

$$\widetilde{l} = Z^t l$$

and in the online phase we take:

$$\widetilde{s}(\mu) = \left\langle \widetilde{l}, \widetilde{u}(\mu) \right\rangle$$

as an approximation for $s(\mu)$.

Under additional coercivity hypothesis on A, the reduced basis method [72] also provides an efficient offline-online procedure for computing $\varepsilon^u(\mu)$ so that the approximation can be *certified*:

$$\forall \mu \in \mathcal{P} \quad \|u(\mu) - Z\widetilde{u}(\mu)\| \leq \varepsilon^u(\mu).$$

The online procedure for the computation of $\varepsilon(\mu)$ is also of complexity independent of $\dim X$.

This online error bound can in turn be used to provide a certification on the output:

$$\forall \mu \in \mathcal{P} \quad |s(\mu) - \widetilde{s}(\mu)| \leq \underbrace{\|l\|\, \varepsilon^u(\mu)}_{=:\varepsilon^L(\mu)} \tag{5.1.7}$$

and this bound is clearly "optimal" amongst those depending on μ through $\varepsilon^u(\mu)$ only. We call this bound the "Lipschitz" bound, and denote it by $\varepsilon^L(\mu)$.

We here notice that [72] uses a different approximation of $s(\mu)$, which also depends on the solution of the adjoint equation of (5.1.1) projected on a suitably selected dual reduced basis. While this approximation has a better rate of convergence, its computation roughly requires doubling of both offline and online complexities as well as storage requirements. The aim of our work is to bound $|s(\mu) - \widetilde{s}(\mu)|$ by a quantity which is smaller than the right-hand side of (5.1.7) and can be computed using an efficient offline-online procedure which does not require computation of $\varepsilon^u(\mu)$ nor any adjoint problem solution during the online phase.

5.1.2 Theoretical error bound

In this section, we give the expression of our output error bound. We begin by some notations: let's denote the residual by $r(\mu)$:

$$r(\mu) = A(\mu)Z\widetilde{u}(\mu) - f(\mu) \in X,$$

and the adjoint problem solution (which will naturally appear in the proof of Theorem 5) by $w(\mu)$:

$$w(\mu) = A(\mu)^{-t}l.$$

Let, for any orthonormal basis $\Phi = \{\phi_1, \ldots, \phi_{\mathcal{N}}\}$ of X, any $N \in \mathbb{N}^*$, and $i = 1, \ldots, N$,

$$D_i(\mu, \Phi) = \langle w(\mu), \phi_i \rangle .$$

We take a partition $\{\mathcal{P}_1, \ldots, \mathcal{P}_K\}$ of the parameter space $\mathcal{P}$, that is:

$$\mathcal{P} = \cup_{k=1}^{K} \mathcal{P}_k \quad \text{and} \quad k \neq k' \rightarrow \mathcal{P}_k \cap \mathcal{P}_{k'} = \emptyset.$$

We set, for $i = 1, \ldots, N$ and $k = 1, \ldots, K$:

$$\beta_{i,k}^{min}(\Phi) = \min_{\mu \in \mathcal{P}_k} D_i(\mu), \quad \beta_{i,k}^{max}(\Phi) = \max_{\mu \in \mathcal{P}_k} D_i(\mu),$$

and:

$$\beta_i^{up}(\mu, \Phi) = \begin{cases} \beta_{i,k(\mu)}^{max}(\Phi) \text{ if } \langle r(\mu), \phi_i \rangle > 0 \\ \beta_{i,k(\mu)}^{min}(\Phi) \text{ else,} \end{cases}$$

$$\beta_i^{low}(\mu, \Phi) = \begin{cases} \beta_{i,k(\mu)}^{min}(\Phi) \text{ if } \langle r(\mu), \phi_i \rangle > 0 \\ \beta_{i,k(\mu)}^{max}(\Phi) \text{ else,} \end{cases}$$

where $k(\mu)$ is the only k in $\{1, \ldots, K\}$ so that $\mu \in \mathcal{P}_k$. We also set:

$$T_1^{low}(\mu, N, \Phi) = \sum_{i=1}^{N} \langle r(\mu), \phi_i \rangle \, \beta_i^{low}(\mu, \Phi), \quad T_1^{up}(\mu, N, \Phi) = \sum_{i=1}^{N} \langle r(\mu), \phi_i \rangle \, \beta_i^{up}(\mu, \Phi),$$

$$T_1(\mu, N, \Phi) = \max \left(\left| T_1^{low}(\mu, N, \Phi) \right|, |T_1^{up}(\mu, N, \Phi)| \right).$$

Finally, we suppose that μ is a random variable on $\mathcal{P}$ and set:

$$T_2(N, \Phi) = \mathbf{E}_\mu \left(\left| \sum_{i>N} \langle w(\mu), \phi_i \rangle \langle r(\mu), \phi_i \rangle \right| \right),$$

where we take, for convenience, $\phi_i = 0$ for all $i > \mathcal{N}$ (so that the sum above is in fact between N and $\mathcal{N}$).
We have the following theorem:

Theorem 5. *For any $\alpha \in]0;1[$ and for any $N \in \mathbb{N}^*$, we have:*

$$P\left(|s(\mu) - \widetilde{s}(\mu)| > T_1(\mu, N, \Phi) + \frac{T_2(N, \Phi)}{\alpha}\right) \leq \alpha.$$

Proof. We begin by noticing that:

$$A(\mu)^{-1} r(\mu) = Z\widetilde{u}(\mu) - u(\mu)$$

so that:

$$\widetilde{s}(\mu) - s(\mu) = \langle l, Z\widetilde{u}(\mu) - u(\mu)\rangle = \left\langle l, A(\mu)^{-1} r(\mu)\right\rangle = \langle w(\mu), r(\mu)\rangle$$

We expand the residual in the Φ basis:

$$r(\mu) = \sum_{i\geq 1} \langle r(\mu), \phi_i\rangle \, \phi_i.$$

Hence:

$$\widetilde{s}(\mu) - s(\mu) = \sum_{i\geq 1} \left\langle l, A(\mu)^{-1}\phi_i\right\rangle \langle r(\mu), \phi_i\rangle = \sum_{i\geq 1} \langle w(\mu), \phi_i\rangle \langle r(\mu), \phi_i\rangle . \tag{5.1.8}$$

We clearly have that for any $N \in \mathbb{N}^*$,:

$$\sum_{i=1}^{N} \langle r(\mu), \phi_i\rangle \, \beta_i^{low}(\mu, \Phi) \leq \sum_{i=1}^{N} \langle r(\mu), \phi_i\rangle \langle w(\mu), \phi_i\rangle \leq \sum_{i=1}^{N} \langle r(\mu), \phi_i\rangle \, \beta_i^{up}(\mu, \Phi)$$

and this implies:

$$\left|\sum_{i=1}^{N} \langle r(\mu), \phi_i\rangle \langle w(\mu), \phi_i\rangle\right| \leq T_1(\mu, N, \Phi). \tag{5.1.9}$$

So we have:

$$\begin{aligned}
& P\left(|s(\mu) - \widetilde{s}(\mu)| > T_1(\mu, N, \Phi) + \frac{T_2(N, \Phi)}{\alpha}\right) \\
\leq\ & P\left(|s(\mu) - \widetilde{s}(\mu)| > \left|\sum_{i=1}^{N} \langle r(\mu), \phi_i\rangle \left\langle A(\mu)^{-t} l, \phi_i\right\rangle\right| + \frac{T_2(N, \Phi)}{\alpha}\right) \text{ by (5.1.9)} \\
=\ & P\left(|s(\mu) - \widetilde{s}(\mu)| - \left|\sum_{i=1}^{N} \langle r(\mu), \phi_i\rangle \left\langle A(\mu)^{-t} l, \phi_i\right\rangle\right| > \frac{T_2(N, \Phi)}{\alpha}\right) \\
\leq\ & P\left(\left|\sum_{i>N} \langle r(\mu), \phi_i\rangle \left\langle A(\mu)^{-t} l, \phi_i\right\rangle\right| > \frac{T_2(N, \Phi)}{\alpha}\right) \text{ by (5.1.8)} \\
\leq\ & \alpha \text{ thanks to Markov's inequality.} \quad \square
\end{aligned}$$

Choice of Φ. The error bound given in Theorem 5 above is valid for any orthonormal basis Φ. For efficiency reasons, we would like to choose Φ so that the parameter-independent part $T_2(N, \Phi)$ is the smallest possible, for a fixed truncation index $N \in \mathbb{N}^*$.

To our knowledge, minimizing $T_2(N, \Phi)$ over orthonormal bases of X is an optimization problem for which no efficient algorithm exists. However, we can minimize an upper bound of $T_2(N, \Phi)$.

We define an auto-adjoint, positive operator $G : X \to X$ by:

$$\forall \phi \in X, \quad G\phi = \frac{1}{2}\mathbf{E}_\mu \left(\langle r(\mu), \phi \rangle \, r(\mu) + \langle w(\mu), \phi \rangle \, w(\mu) \right). \tag{5.1.10}$$

Let $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_{\mathcal{N}} \geq 0$ be the eigenvalues of G. Let, for $i \in \{1, 2 \ldots, \mathcal{N}\}$, ϕ_i^G be an unit eigenvector of G associated with the i^{th} eigenvalue, and $\Phi^G = \{\phi_1^G, \ldots, \phi_{\mathcal{N}}^G\}$.

We can state that:

Theorem 6.

$$T_2(N, \Phi^G) \leq \sum_{i>N} \lambda_i^2.$$

Proof. We have:

$$T_2(N, \Phi) \leq \frac{1}{2}\mathbf{E}_\mu \left(\sum_{i>N} \langle u(\mu), \phi_i \rangle^2 + \sum_{i>N} \langle r(\mu), \phi_i \rangle^2 \right) =: T_2^{sup}(N, \Phi) = \sum_{i>N} \langle G\phi_i, \phi_i \rangle$$

Using Theorem 1.1 of [112], we get that the minimum of $T_2^{sup}(N, \Phi)$ is attained for $\Phi = \Phi^G$, and that minimum is $\sum_{i>N} \lambda_i^2$. □

This theorem suggests to use $\Phi = \Phi^G$, so as to control $T_2(N, \Phi)$.

5.1.3 Computable error bound

In this Subsection, we present an implementable offline/online procedure for the estimation of the upper bound for $|\widetilde{s}(\mu) - s(\mu)|$ presented in Theorem 5.

Estimation of ϕ_i^G. We fix a truncation index $N \in \mathbb{N}^*$, and we estimate $\{\phi_i^G\}_{i=1,\ldots,N}$ by using a modification of the method of snapshots used in Proper Orthogonal Decomposition [96]. This estimation is performed during the offline phase. We take a finite (large), subset of parameters $\Xi \subset \mathcal{P}$,

randomly sampled from the distribution of the parameter, and we approximate the G operator by:

$$\widehat{G}\phi = \frac{1}{2\#\Xi}\sum_{\mu\in\Xi}\left(\langle r(\mu),\phi\rangle\, r(\mu) + \langle w(\mu),\phi\rangle\, w(\mu)\right)$$

In other words, $\widehat{G}$ is a Monte-Carlo estimator of G. We take $\{\widehat{\phi}_i^G\}_{i=1,\dots,N}$ as the unit eigenvectors associated with the N largest eigenvalues of $\widehat{G}$. The operator $\widehat{G}$ admits the following matrix representation:

$$\widehat{G} = \frac{1}{2\#\Xi}\left(WW^t + RR^t\right),$$

where W (resp. R) is the matrix whose columns are the components of $w(\mu)$ (resp. $r(\mu)$) in a basis of X, for $\mu\in\Xi$. These two matrices have $\#\Xi$ columns and $\dim X$ lines, which means that the matrix above is $\dim X \times \dim X$. In general, we take $\#\Xi \ll \dim X$, and so it is computationally advantageous to notice that if ϕ is an eigenvector of $\widehat{G}$ associated with a nonzero eigenvalue λ, then:

$$\frac{1}{\lambda}\frac{1}{2\#\Xi}\left((WW^t\phi + RR^t\phi)\right) = \phi,$$

so that $\phi \in \text{Im } W + \text{Im } R =: \mathcal{V}$. Hence, if V is the matrix of an orthonormal basis of $\mathcal{V}$, then there exists ψ so that $\phi = V\psi$ and we have:

$$\frac{1}{2\#\Xi}WW^t\phi + RR^t\phi = \lambda\phi \implies \left[V^t\frac{1}{2\#\Xi}\left((WW^t + RR^t)\right)V\right]\psi = \lambda\psi.$$

As a consequence, it is sufficient to find the dominant eigenvectors $\widehat{\psi}_1^G,\dots,\widehat{\psi}_N^G$ of the matrix $\Sigma = \dfrac{1}{2\#\Xi}V^t(WW^t + RR^t)V$ (of size 2Ξ), and to deduce $\widehat{\phi}_i^G$ from $\widehat{\psi}_i^G$ by the relation $\widehat{\phi}_i^G = V\widehat{\psi}_i^G$. Besides, by writing Σ as:

$$\Sigma = \frac{1}{2\#\Xi}\left((V^tW)(W^tV) + (V^tR)(R^tV)\right),$$

it is possible to compute and store Σ without storing nor computing any dense $\dim X \times \dim X$ matrix.

Computation of $T_1(\mu, N, \Phi)$. For $i = 1,\dots,N$ and $k = 1,\dots,K$, the reals $\beta_{i,k}^{min}(\Phi)$ and $\beta_{i,k}^{max}(\Phi)$ can be computed during the offline phase, as they are parameter-independent. They can be approximated by a simple discrete minimization:

$$\widetilde{\beta}_{i,k}^{min}(\Phi) = \min_{\mu\in\Xi\cap\mathcal{P}_k} D_i(\mu,\Phi), \quad \widetilde{\beta}_{i,k}^{max}(\Phi) = \max_{\mu\in\Xi\cap\mathcal{P}_k} D_i(\mu,\Phi),$$

or, thanks to the availability of the gradient of $D_i(\mu)$ with respect to μ, by a more elaborate quasi-Newton optimization such as L-BFGS [118].
We also compute the following parameter-independent, offline-computable quantities:

$$\left\langle f_{q'}, \widehat{\phi}_i^G \right\rangle, \left\langle A_q \zeta_j, \widehat{\phi}_i^G \right\rangle \ (i = 1, \dots, N,\ j = 1, \dots, n,\ q = 1, \dots, Q,\ q' = 1, \dots, Q')$$

where $\{\zeta_1, \dots, \zeta_n\}$ is a basis of the reduced space $\widetilde{X}$.
Now, let a parameter $\mu \in \mathcal{P}$ be given, and let $\widetilde{u}_1(\mu), \dots, \widetilde{u}_n(\mu)$ be the components of the reduced solution $\widetilde{u}(\mu)$ in the reduced basis $\{\zeta_1, \dots, \zeta_n\}$.
By using the relation:

$$\left\langle r(\mu), \widehat{\phi}_i^G \right\rangle = \sum_{q=1}^{Q} \Theta_q(\mu) \sum_{j=1}^{n} \widetilde{u}_j(\mu) \left\langle A_q \zeta_j, \widehat{\phi}_i^G \right\rangle - \sum_{q'=1}^{Q'} \gamma_{q'}(\mu) \left\langle f_{q'}, \phi_i^G \right\rangle,$$

the dot products between the residual and $\widehat{\phi}_i^G$ can be computed in the online phase, with a complexity of $O(nQ+Q')$ arithmetic operations, $O(Q)$ evaluations of Θ functions and $O(Q')$ evaluations γ functions, which is independent of $\dim X$. Then, β_i^{low} and β_i^{up} can be straightforwardly deduced.
Estimation of $T_2(N, \Phi)$, final error bound. We approximate $T_2(N, \Phi)$ by computing the following Monte-Carlo estimator:

$$\widehat{T}_2(N, \Phi) = \frac{1}{2\#\Xi} \sum_{\mu \in \Xi} \left| \widetilde{s}(\mu) - s(\mu) - \sum_{i=1}^{N} \langle w(\mu), \phi_i \rangle \langle r(\mu), \phi_i \rangle \right|.$$

As this quantity is μ-independent, it can be computed once and for all during the offline phase.
By using Theorem 5, we get that for $\varepsilon(\mu, \alpha, N, \Phi) = T_1(\mu, N, \Phi) + T_2(N, \Phi)/\alpha$, we have:

$$P\left(|s(\mu) - \widetilde{s}(\mu)| \geq \varepsilon(\mu, \alpha, N, \Phi)\right) \leq \alpha,$$

so we may take, as estimated error bound with risk α,

$$\widehat{\varepsilon}(\mu, \alpha, N, \Phi) = T_1(\mu, N, \Phi) + \frac{\widehat{T}_2(N, \Phi)}{\alpha}. \tag{5.1.11}$$

5.2 Application to sensitivity analysis

Our error estimation method is applied in sensitivity analysis, so as to quantify the error caused by the replacement of the original model output by the

reduced basis output during the Monte-Carlo estimation of the Sobol indices. For the sake of self-completeness, we briefly present the aim and the computation of these indices, and we refer to [88], [86] and [58] for details.

5.2.1 Definition of the Sobol indices

For $i = 1, \dots, p$, the i^{th} Sobol index of a function of p variables $s(\mu_1, \dots, \mu_p)$ is defined by:

$$S_i = \frac{\text{Var}\left(\mathbb{E}(s(\mu_1, \dots, \mu_p)|\mu_i)\right)}{\text{Var}\left(s(\mu_1, \dots, \mu_p)\right)}, \tag{5.2.1}$$

the variances and conditional expectation being taken with respect to a postulated distribution of the $(\mu_1, \dots, \mu_p)$ input vector accounting for the uncertainty on the inputs' value. These indices are well defined as soon as $s \in L^2(\mathcal{P})$ and that $\text{Var}\left(s(\mu_1, \dots, \mu_p)\right) \neq 0$. When $\mu_1, \dots, \mu_p$ are (stochastically) independent, the i^{th} Sobol index can be interpreted as the fraction of the variance of the output that is caused by the uncertainty on the i^{th} parameter μ_i. All the Sobol indices lie in $[0; 1]$; the closer to zero (resp., one) S_i is, the less (resp., the more) importance μ_i's uncertainty has on s's uncertainty.

5.2.2 Estimation of the Sobol indices

The conditional expectation and variances appearing in (5.2.1) are generally not amenable to analytic computations. In those cases, one can estimate S_i by using a Monte-Carlo method which works as follows: from two M-sized random, independent samples of inputs' distribution, we compute $2M$ appropriate evaluations $\{s_j\}$ and $\{s'_j\}$ of s, and estimate S_i by:

$$\widehat{S}_i = \frac{\frac{1}{M}\sum_{j=1}^{M} s_j s'_j - \left(\frac{1}{M}\sum_{j=1}^{M} s_j\right)\left(\frac{1}{M}\sum_{j=1}^{M} s'_j\right)}{\frac{1}{M}\sum_{j=1}^{M} s_j^2 - \left(\frac{1}{M}\sum_{j=1}^{M} s_j\right)^2}. \tag{5.2.2}$$

When M and/or the required time for the evaluation of the model output are large, it is computationally advantageous to replace s by its surrogate model $\widetilde{s}$. By using (5.2.2) on $\widetilde{s}$ (hence with reduced model outputs $\{\widetilde{s}_j\}$ and $\{\widetilde{s}'_j\}$), one estimates the Sobol indices *of the surrogate model* rather than those of the true model. We presented in [58], Sections 3.1 and 3.2, a method to quantify the error made in the Sobol index estimation when replacing the original model by the surrogate one, we presented in [58], Sections 3.1 and

3.2, two estimators $\widehat{S}^m_{i,\alpha_{as}/2}$ and $\widehat{S}^M_{i,1-\alpha_{as}/2}$ relying on output error bound samples $\{\varepsilon_j\}$ and $\{\varepsilon'_j\}$ so that we have:

Theorem 7. *If:*

$$\forall j = 1, \ldots, M, \quad |s_j - \widetilde{s}_j| \leq \varepsilon_j \ \text{ and } \ \left|s'_j - \widetilde{s}'_j\right| \leq \varepsilon'_j,$$

then we have (under certain hypotheses on the bootstrap method in use):

$$P\left(S_i \in [\widehat{S}^m_{i,\alpha_{as}/2}; \widehat{S}^M_{i,1-\alpha_{as}/2}]\right) \geq 1 - \alpha_{as}.$$

In our case, the output error bound $\varepsilon(\mu)$ of Theorem 5 does not satisfy the above hypothesis, but satisfies a weaker "probabilistic" statement. This is the object of the following Corollary:

Corollary 1. *If:*

$$\forall j = 1, \ldots, M, \quad P\left(|s_j - \widetilde{s}_j| \geq \varepsilon_j\right) \leq \alpha \ \text{ and } \ \forall j = 1, \ldots, M, \quad P\left(\left|s'_j - \widetilde{s}'_j\right| \geq \varepsilon'_j\right) \leq \alpha,$$

then we have:

$$P\left(S_i \in [\widehat{S}^m_{i,\alpha_{as}/2}; \widehat{S}^M_{i,1-\alpha_{as}/2}]\right) \geq (1 - \alpha_{as}) \times (1 - \alpha)^{2M}.$$

Proof. We easily have that:

$$\begin{aligned} P\left(S_i \in [\widehat{S}^m_{i,\alpha_{as}/2}; \widehat{S}^M_{i,1-\alpha_{as}/2}]\right) &\geq P\left(S_i \in [\widehat{S}^m_{i,\alpha_{as}/2}; \widehat{S}^M_{i,1-\alpha_{as}/2}] \,|\, \forall j, |s_j - \widetilde{s}_j| < \varepsilon(\mu)\right) \\ &\quad \times P\left(\forall j, |s_j - \widetilde{s}_j| < \varepsilon(\mu)\right) \\ &\geq (1 - \alpha_{as}) \times (1 - \alpha)^{2M}. \end{aligned}$$

□

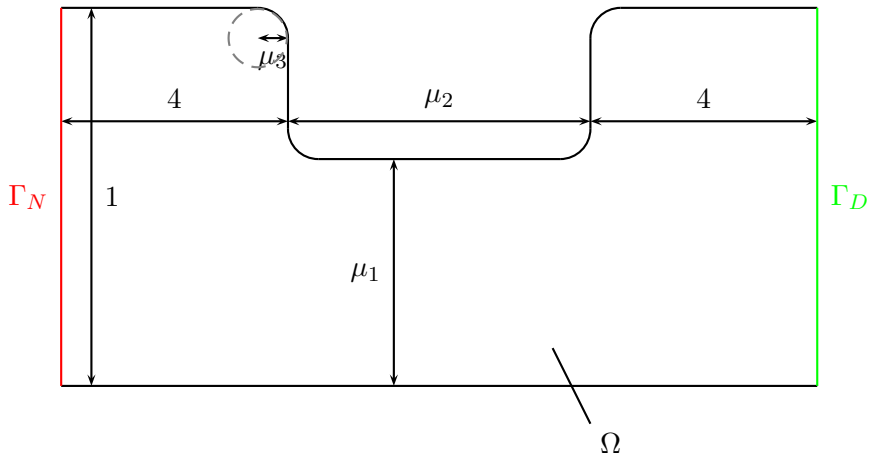
5.3 Numerical results I: Diffusion equation

5.3.1 Benchmark problem

Our benchmark problem [85] is the following: given a parameter vector

$$\mu = (\mu_1, \mu_2, \mu_3) \in \mathcal{P} = [0.25, 0.5] \times [2, 4] \times [0.1, 0.2],$$

we consider the domain $\Omega = \Omega(\mu)$ below:



Our continuous field variable $u_e = u_e(\mu) \in X_e$ satisfies:

$$\begin{cases} \Delta u_e = 0 \text{ in } \Omega \\ u_e = 0 \text{ on } \Gamma_D \\ \dfrac{\partial u_e}{\partial n} = -1 \text{ on } \Gamma_N \\ \dfrac{\partial u_e}{\partial n} = 0 \text{ on } \partial\Omega \setminus (\Gamma_N \cup \Gamma_D) \end{cases} \tag{5.3.1}$$

where

$$X_e = \{v \in H^1(\Omega) \text{ s.t. } v|_{\Gamma_D} = 0\},$$

Δ denotes the Laplace operator, and $\dfrac{\partial}{\partial n}$ is the normal derivative with respect to $\partial\Omega$.

This continuous variable denotes the potential of a steady, incompressible flow moving in a tube whose profile is given by Ω, with open ends on Γ_N and Γ_D. The Neumann boundary condition on Γ_N states that the fluid enters by Γ_N with unit speed, the condition on $\partial\Omega \setminus (\Gamma_N \cup \Gamma_D)$ states that the velocity field is tangential to the boundary of the tube; finally the Dirichlet condition on Γ_D guarantees well-posedness, as the potential field is determinated up to a constant.

The problem (5.3.1) is equivalent to the following variational formulation: find $u_e = u_e(\mu) \in X_e$ so that:

$$\int_\Omega \nabla u_e \cdot \nabla v = -\int_{\Gamma_N} v \quad \forall v \in X_e.$$

This variational problem is well-posed, as the bilinear form $(u, v) \mapsto \int_\Omega \nabla u \cdot \nabla v$ is coercive on X_e (see, for instance, [107], lemma A.14).

The above variational problem is discretized using a finite triangulation $\mathcal{T}$ of Ω and the associated $P^1(\mathcal{T})$ (see [20] or [80]) finite element subspace: find $u \in X$ so that

$$\int_\Omega \nabla u \cdot \nabla v = -\int_{\Gamma_N} v \quad \forall v \in X,$$

where $X = \{v \in \mathbf{P}^1(\mathcal{T}) \text{ s.t. } v|_{\Gamma_D} = 0\}$.
In our experiments, $\dim X = 525$.
The affine decomposition of the matrix of the bilinear form in the left-hand side of the above equation is obtained by using a piecewise affine mapping from $\Omega(\mu)$ to a reference domain $\bar{\Omega}$ as explained in [79], page 11. The number of terms in the obtained affine decompositions are $Q = 9$ and $Q' = 1$.
Our scalar output of interest is taken to be:

$$s(\mu) = \int_{\Gamma_N} u(\mu),$$

and $\mathcal{P}$ is endowed with the uniform distribution.

5.3.2 Results

We now present the numerical results obtained using our error estimation procedure on the output of the model described above. We give a comparison with the natural "Lipschitz" bound $\varepsilon^L(\mu)$ given in (5.1.7) and with the dual-based error estimation method [72]. The bound $\varepsilon^u(\mu)$ on $\|Z\tilde{u}(\mu) - u(\mu)\|$ is computed using the procedure described in [72]. Note that we estimated the "lower bound for the inf-sup parameter" "by inspection", as explained in [72], Section 3.3.2.
We also compare our goal-oriented error bound with the dual-based error bound method described in [72] (Section 3.2). This method involves the computation, during the online phase, of the reduced-basis solution of a linear, output-dependent, adjoint problem. The solution of this adjoint problem enables correction and estimation of the reduced output, at the expense of a doubled online computation cost.
For the comparisons to be fair, one should compare the error bounds of same online cost. It is widely assumed that there exists a constant C so that this cost is $C \times 2(\dim \widetilde{X})^3$ for the dual-based method, and $C(\dim \widetilde{X})^3$ for our method, since dual-based method involves online inversion of two linear systems of size $\dim \widetilde{X}$, and one system of the same size for our method. Hence, the reported reduced basis sizes for the dual method are multiplied by a factor $\sqrt[3]{2}$.

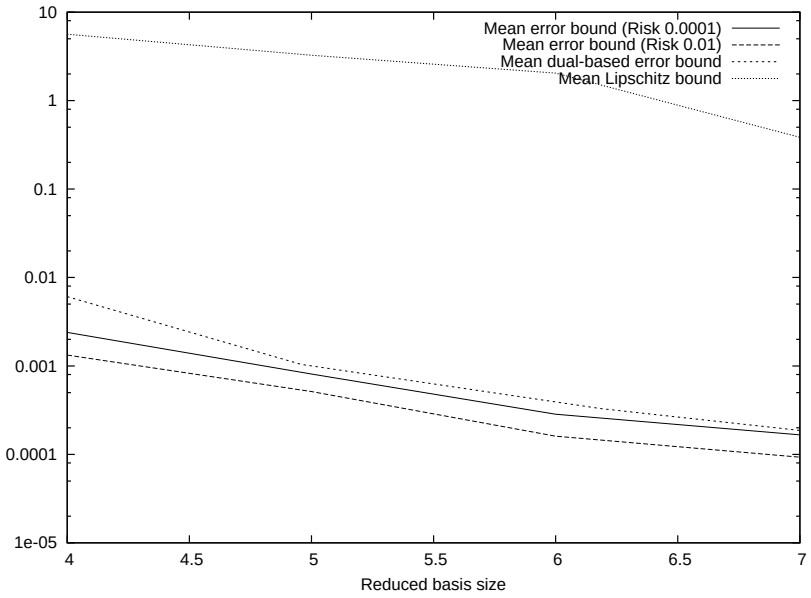


Figure 5.1: Comparison of the mean error bounds (for risk $\alpha \in \{0.01, 0.0001\}$), the mean Lipschitz bounds and the mean dual-based error bounds.

In all cases, the reduced bases are computed using POD with snapshot size 80. To compute $\widehat{G}$, we use a snapshot of size 200.

The partition used to split the $\mathcal{P}$ set is compound of $K = 8$ cubes whose edges are parallel to the axes, and are chosen to contain (approximatively) the same number of points of an input sample.

In Figure 5.1, we give, for various reduced bases and various target risk levels α, the mean error bound on the output:

$$\bar{\varepsilon} = \frac{1}{\#S} \sum_{\mu \in S} \widehat{\varepsilon}(\mu, \alpha, N, \Phi)$$

where S is a random subset of $\mathcal{P}$ with size 200, $\widehat{\varepsilon}(\mu, \alpha, N, \Phi)$ is defined at (5.1.11), and the truncation index N is taken equal to 15.

For all the graphs, an unique test sample S is kept so as to guarantee fair comparisons.

Note that, due to limited numerical precision, the error bounds have been truncated upwards to 9×10^{-5}.

We observe that our new, goal-oriented, output error bound largely outperforms the non-goal-oriented Lipschitz bound and is better than the goal-oriented, dual-based error bound. Finally, we observe that the high con-

Input parameter	$\left[\widehat{S}_i^m; \widehat{S}_i^M\right]$	$\left[\widehat{S}^m_{i,\alpha_{as}/2}; \widehat{S}^M_{i,1-\alpha_{as}/2}\right]$
μ_1	[0.530352;0.530933]	[0.48132; 0.5791]
μ_2	[0.451537;0.452099]	[0.397962;0.51139]
μ_3	[0.00300247;0.0036825]	[-0.0575764;0.0729923]

Table 5.1: Results of the application of Section 5.2 to the estimation of the Sobol indices of the output of our benchmark model.

centration of $\widehat{T}_2$ around zero allows us to choose a very low target risk and remain very competitive.

5.3.3 Application to sensitivity analysis

We estimate confidence intervals for the sensitivity indices of $s(\mu)$ by using the method described in [58], together with the remarks in Section 2.2.

We take $M = 1000$ as sample size, $B = 500$ as number of bootstrap replications, $\dim \tilde{X} = 10$ as reduced basis size, $\alpha = 0.00001$ as output error bound risk, and $\alpha_{as} = 0.05$ as Monte-Carlo risk. The level of the combined confidence interval $\left[\widehat{S}^m_{i,\alpha_{as}/2}; \widehat{S}^M_{i,1-\alpha_{as}/2}\right]$ is then $(1-\alpha_{as})(1-\alpha)^M > 0.93$.

The results are gathered in Table 5.1. The spread between $\widehat{S}_i^m$ and $\widehat{S}_i^M$ accounts for the *metamodel-induced* error in the estimation of the Sobol indices. The remaining spread between $\widehat{S}^m_{i,\alpha_{as}/2}$ and $\widehat{S}^M_{i,1-\alpha_{as}/2}$ is the impact of the sampling error (due to the replacement of the variances in the definition of the Sobol indices by their empirical estimators). We see that, in this case, the metamodel-induced error (certified by the use of our goal-oriented error bound) is very small with regard to the sampling error.

5.4 Numerical results II: transport equation

We now apply our error bound on a non-homogeneous linear transport equation. Compared to the previous example, the considered PDE is of a different kind (hyperbolic rather than elliptic).

5.4.1 Benchmark problem

In this problem, the continuous field $u_e = u_e(x,t)$ is the solution of the linear transport equation:

$$\frac{\partial u_e}{\partial t}(x,t) + \mu \frac{\partial u_e}{\partial x}(x,t) = \sin(x)\exp(-x)$$

for all $(x,t) \in]0,1[\times]0,1[$, satisfying the initial condition:

$$u_e(x, t=0) = x(1-x) \quad \forall x \in [0,1],$$

and boundary condition:

$$u_e(x=0,t) = 0 \quad \forall t \in [0,1].$$

The parameter μ is chosen in $\mathcal{P} = [0.5, 1]$ and $\mathcal{P}$ is endowed with the uniform measure.

We now choose a spatial discretization step $\Delta x > 0$ and a time discretization step $\Delta t > 0$, and we introduce our discrete unknown $u = (u_i^n)_{i=0,\dots,N_x;n=0,\dots,N_t}$ where

$$N_x = \frac{1}{\Delta x}, \quad \text{and} \quad N_t = \frac{1}{\Delta t}.$$

We note here that the considered PDE is hyperbolic and time-dependent, and that we perform the reduction on the space-time unknown u, of dimension $(N_x+1)\cdot(N_t+1)$. This is different from reducing the space-discretized equation at each time step.

The u vector satisfies the discretized initial-boundary conditions:

$$\forall i, \quad u_i^0 = (i\Delta x)(1 - i\Delta x) \tag{5.4.1}$$

$$\forall n, \quad u_0^n = 0 \tag{5.4.2}$$

and the first-order upwind scheme implicit relation:

$$\forall i, n \quad \frac{u_i^{n+1} - u_i^n}{\Delta t} + \mu \frac{u_{i+1}^{n+1} - u_i^{n+1}}{\Delta x} = \sin(i\Delta x)\exp(-i\Delta x). \tag{5.4.3}$$

Let's denote by $B = B(\mu)$ (resp. y) the matrix (resp. the vector) so that (5.4.1),(5.4.2) and (5.4.3) are equivalent to:

$$Bu = y \tag{5.4.4}$$

that is:

$$B^T B u = B^T y, \tag{5.4.5}$$

so that equation (5.4.5) is (5.1.1) with $A(\mu) = B^T B$ and $f = B^T y$.

The output of interest is: $s(\mu) = u_{N_x}^{N_t}$. In the following, we take $\Delta t = 0.02$ and $\Delta x = 0.05$.

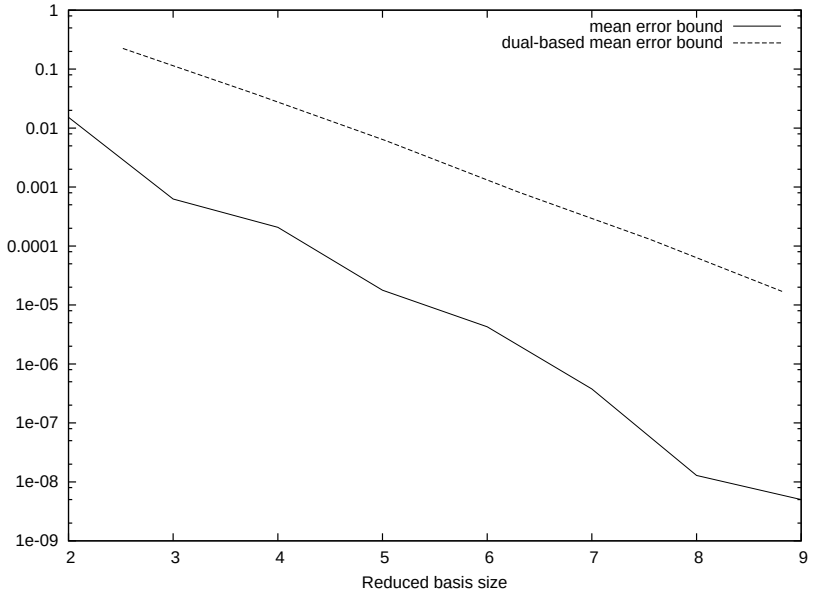


Figure 5.2: Comparison between the mean error bound (for risk $\alpha = 0.01$) and the mean dual-based error bound, for different reduced basis sizes.

5.4.2 Results

We took a very low risk level $\alpha = 0.01$, a snapshot size of 70, $N = 10$ retained $\widehat{\phi}_i^G$ vectors and $K = 1$. The results (Figure 5.2) show that, once again, the error bound we propose in this paper outperforms the dual-based error bound.

Conclusion

We have presented a new explicitly computable output error bound for the reduced-basis method. We have shown, on two different practical examples, that this bound is clearly better than the naive Lipschitz bound and that, at the expense of a slight, controllable risk, the performances of this new bound are better than the existing dual-based output error bound.

Conclusion et perspectives

Ce travail de thèse, dont la particularité est de se situer à la frontière entre le stochastique (chapitres 2, 3, et, en partie, le chapitre 5) et le déterministe (chapitres 4 et 5), a permis de montrer qu'en intégrant une analyse de l'erreur de métamodèle dans l'estimation des indices de sensibilité de Sobol, il était possible de profiter du gain en temps de calcul apporté par l'utilisation d'un métamodèle, tout en bénéficiant d'une certification rigoureusement justifiée de l'estimation fournie. Cette certification peut se baser sur celle d'un métamodèle construit par la méthode base réduite, que nous avons étendue et améliorée sur deux aspects : par l'augmentation de ses performances dans le cas de l'équation de Burgers, qui devrait ouvrir la voie à son utilisation sur le modèle de Saint-Venant, ainsi que par la proposition d'une nouvelle méthode de borne d'erreur sortie-dépendante, dont la précision est cruciale en analyse de sensibilité. Nous pensons que les contributions présentées dans cette thèse permettront l'analyse de sensibilité globale certifiée des modèles numériques utilisés en océanographie.

Présentons maintenant quelques perspectives induites par ce travail.

Perspectives

Les trois premières perspectives que nous présentons sont des prolongements des questions 1 à 4 que nous avons formulées en section 1.1. La dernière est reliée à l'utilisation simultanée du modèle et du métamodèle pour l'analyse de sensibilité. Ces quatre perspectives, toutes reliées à l'estimation certifiée des indices de Sobol en un minimum de temps de calcul, peuvent être traitées indépendamment l'une de l'autre.

Estimation et efficacité pour l'estimation conjointe d'indices

Cette perspective prolonge notre réponse aux questions 1 (traitement de l'erreur Monte-Carlo dans l'estimation des indices de Sobol) et 2 (identification d'un estimateur de variance optimale).
Au chapitre 3, nous avons donné des résultats de normalité asymptotique univariée dans le cas de l'estimation d'un seul indice de Sobol du premier ordre.
Il s'agit maintenant de considérer l'estimation conjointe de plusieurs indices de Sobol, relativement à différentes variables d'entrée, et éventuellement pour des ordres supérieurs à un. Ceci est motivé par le fait que les utilisateurs de l'analyse de sensibilité estiment souvent tous les indices d'ordre un et tous les indices totaux : $(S_1, \ldots, S_p, S_1^{\mathrm{T}}, \ldots, S_p^{\mathrm{T}})$.
L'obtention d'une propriété de normalité asymptotique pour le vecteur d'estimateurs permettra la construction de régions de confiance asymptotiques fines, ouvrant également la voie à des tests, asymptotiquement justifiés, de non-significativité conjointe des indices de sensibilité.
De plus, l'estimateur asymptotiquement efficace dans ce contexte peut être différent de celui identifié dans le cas univarié, car la nature des observations (les évaluations du modèle) change. Par exemple, dans un modèle à $p = 3$ variables, l'estimation de (S_1, S_2) sur un plan Monte-Carlo *pick-freeze* conduit à utiliser un échantillon de la loi de (Y, Y', Y''), où :

$$Y = f(X_1, X_2, X_3),\ Y' = f(X_1, X_2', X_3'),\ Y'' = f(X_1, X_2, X_3'),$$

et où (X_1', X_2', X_3') est une copie indépendante de (X_1, X_2, X_3).

Estimation d'erreur pour l'analyse de sensibilité basée sur les polynômes de chaos

Nous souhaitons élargir la question 3 (quantification de l'impact de l'utilisation d'un métamodèle sur l'estimation des indices de Sobol par Monte-Carlo) en remplaçant l'estimateur de Monte-Carlo par une estimation basée sur la décomposition en polynômes de chaos de la sortie.
L'utilisation de la décomposition en polynômes de chaos pour estimer les indices de Sobol est motivée par le fait que cette méthode peut tirer partie de la régularité de la sortie en fonction des entrées, afin d'atteindre une vitesse de convergence supérieure à celle obtenue par Monte-Carlo.

Réduction certifiée pour les équations de Saint-Venant

Cette perspective s'inscrit dans le cadre de la question 4 : développement de métamodèles certifiés pour des modèles de géophysique.
Dans le chapitre 4, nous avons présenté une méthode efficace applicable à l'équation de Burgers basée sur l'approximation base réduite, qui offre un gain substantiel en temps de calcul tout en certifiant les approximations faites pour permettre ces gains en temps de calcul. L'équation de Burgers étant un « prototype » des équations de Saint-Venant (Shallow-Water) utilisées en modélisation fluviale et océanographique, au sens où elle présente les deux principales difficultés mathématiques de celui-ci (la non-linéarité quadratique, et la dépendance en temps), cette méthode devrait très probablement pouvoir s'adapter aux équations de Saint-Venant.
Nous avons déjà implémenté une version réduite, aux performances encourageantes, d'un code Saint-Venant basé sur un schéma explicite en temps et une méthode aux différences finies en espace. Cependant, les premiers résultats laissent penser que la borne d'erreur certifiant ce code réduit sera beaucoup trop conservative. L'utilisation d'un schéma implicite en temps (comme pour le travail de réduction de l'équation de Burgers) devrait permettre d'obtenir une borne exploitable.

Multifidélité pour l'estimation des indices de Sobol

Dans tout ce travail de thèse, nous avons effectué l'estimation des indices de sensibilité en utilisant uniquement des évaluations du métamodèle, les appels au « vrai » modèle étant effectués seulement lors de la phase de construction du métamodèle (phase *offline*).
Une piste de travail, inspirée des quasi-variables de contrôle [34] et des méthodes de Monte-Carlo multiniveau [39] serait d'utiliser également des appels au modèle, conjointement aux appels au métamodèle, durant la phase d'analyse de sensibilité. La proportion d'appels au métamodèle et au vrai modèle est à calibrer en fonction de leurs temps relatifs de calcul, et à la précision du métamodèle.
Cette piste, qui a donné des résultats intéressants sur des exemples « jouets », pourrait encore se prolonger en utilisant une famille de plusieurs codes de calcul, tous métamodèles d'un même métamodèle, mais ayant des précisions et des demandes en temps de calcul différentes, afin d'améliorer encore l'es-

timation des indices de Sobol.

Bibliographie

[1] GEB Archer, A. Saltelli, and IM Sobol. Sensitivity measures, ANOVA-like techniques and the use of bootstrap. *Journal of Statistical Computation and Simulation*, 58(2) :99–120, 1997.

[2] N. Aronszajn. *Theory of reproducing kernels.* Harvard University, 1951.

[3] ARPACK : Arnoldi Package. http://www.caam.rice.edu/software/ARPACK/.

[4] I. Babuska. The finite element method with penalty. *Math. Comp*, 27(122) :221–228, 1973.

[5] P. Barbillon. *Méthodes d'interpolation à noyaux pour l'approximation de fonctions type boîte noire coûteuses.* PhD thesis, Université Paris XI, 2010.

[6] J.W. Barrett and C.M. Elliott. Finite element approximation of the Dirichlet problem using the boundary penalty method. *Numerische Mathematik*, 49(4) :343–366, 1986.

[7] G. Blatman. *Chaos polynomial creux et adaptatif pour la propagation d'incertitudes et l'analyse de sensibilité.* PhD thesis, Université Blaise Pascal - Clermont-Ferrand II, 2009.

[8] E. Borgonovo. A new uncertainty importance measure. *Reliability Engineering & System Safety*, 92(6) :771–784, 2007.

[9] E. Borgonovo, W. Castaings, and S. Tarantola. Moment independent importance measures : New results and analytical test cases. *Risk Analysis*, 31(3) :404–428, 2011.

[10] G.E.P. Box and N.R. Draper. *Empirical model-building and response surfaces.* John Wiley & Sons, 1987.

[11] S. Boyaval, C. Le Bris, Y. Maday, N.C. Nguyen, and A.T. Patera. A reduced basis approach for variational problems with stochastic parameters : Application to heat conduction with variable robin coefficient. *Computer Methods in Applied Mechanics and Engineering*, 198(41-44) :3187–3206, 2009.

[12] A. Buffa, Y. Maday, A.T. Patera, C. Prud'homme, and G. Turinici. A priori convergence of the greedy algorithm for the parametrized reduced basis. *Mathematical Modelling and Numerical Analysis*, 2009.

[13] T. Bui-Thanh, K. Willcox, O. Ghattas, and B. van Bloemen Waanders. Goal-oriented, model-constrained optimization for reduction of large-scale systems. *Journal of Computational Physics*, 224(2) :880–896, 2007.

[14] G.T. Buzzard and D. Xiu. Variance-based global sensitivity analysis via sparse-grid interpolation and cubature. 2008.

[15] D.G. Cacuci. *Sensitivity and Uncertainty Analysis, Theory.* Chapman & Hall/CRC, 2003.

[16] RH Cameron and WT Martin. The orthogonal development of non-linear functionals in series of Fourier-Hermite functionals. *Annals of Mathematics*, 48(2) :385–392, 1947.

[17] Rob Carnell. *lhs : Latin Hypercube Samples*, 2009. R package version 0.5.

[18] A. Chatterjee. An introduction to the proper orthogonal decomposition. *Current Science*, 78(7) :808–817, 2000.

[19] Y. Chen, J.S. Hesthaven, Y. Maday, and J. Rodríguez. Improved successive constraint method based a posteriori error estimate for reduced basis approximation of 2d maxwell's problem. *ESAIM : Mathematical Modelling and Numerical Analysis*, 43(06) :1099–1116, 2009.

[20] P.G. Ciarlet. *The finite element method for elliptic problems.* Society for Industrial Mathematics, 2002.

[21] T. Crestaux et al. Polynomial chaos expansion for sensitivity analysis. *Reliability engineering & System Safety*, 94(7) :1161–1172, 2009.

[22] RI Cukier, CM Fortuin, K.E. Shuler, AG Petschek, and JH. Schaibly. Study of the sensitivity of coupled reaction systems to uncertainties in rate coefficients. i theory. *The Journal of Chemical Physics*, 59(8) :3873, 1973.

[23] RI Cukier, HB Levine, and KE Shuler. Nonlinear sensitivity analysis of multiparameter model systems. *Journal of computational physics*, 26(1) :1–42, 1978.

[24] RI Cukier, JH Schaibly, and K.E. Shuler. Study of the sensitivity of coupled reaction systems to uncertainties in rate coefficients. iii. analysis of the approximations. *The Journal of Chemical Physics*, 63 :1140, 1975.

[25] S. Da Veiga and F. Gamboa. Efficient estimation of nonlinear conditional functionals of a density. *Submitted*, 2008.

[26] S. Da Veiga and A. Marrel. Gaussian process modeling with inequality constraints, To appear in *Annales de la Faculté des Sciences de Toulouse*.

[27] Garrett M. Dancik. *mlegp : Maximum Likelihood Estimates of Gaussian Processes*, 2011. R package version 3.1.2.

[28] E. De Rocquigny, N. Devictor, and S. Tarantola. *Uncertainty in industrial practice*. Wiley Online Library, 2008.

[29] M.K. Deb, I.M. Babuka, and J.T. Oden. Solution of stochastic partial differential equations using Galerkin finite element techniques. *Computer Methods in Applied Mechanics and Engineering*, 190(48) :6359–6372, 2001.

[30] N. Durrande, D. Ginsbourger, O. Roustant, and L. Carraro. Anova kernels and rkhs of zero mean functions for model-based sensitivity analysis. *Journal of Multivariate Analysis*, 2011.

[31] B. Efron. Nonparametric standard errors and confidence intervals. *Canadian Journal of Statistics*, 9(2) :139–158, 1981.

[32] B. Efron and R. Tibshirani. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical science*, 1(1) :54–75, 1986.

[33] B. Efron and R.J. Tibshirani. *An introduction to the bootstrap*. Chapman & Hall/CRC, 1993.

[34] M. Emsermann and B. Simon. Improving simulation efficiency with quasi control variates. 2002.

[35] K. Fang, R. Li, and Agus Sudjianto. *Design and modeling for computer experiments*. Computer science and data analysis series. Chapman & Hall/CRC, 2006.

[36] J.C. Fort, T. Klein, A. Lagnoux, and B. Laurent. Estimation of the sobol indices in a linear functional multidimensional model. 2012.

[37] K. Frank and S. Heinrich. Computing Discrepancies of Smolyak Quadrature Rules. *Journal of Complexity*, 12(4) :287–314, 1996.

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

[39] M.B. Giles and B.J. Waterhouse. Multilevel quasi-monte carlo path simulation. *Advanced Financial Modelling, Radon Series on Computational and Applied Mathematics*, pages 165–181, 2009.

[40] V. Girault and P.A. Raviart. Finite element methods for Navier-Stokes equations, volume 5 of Springer Series in Computational Mathematics, 1986.

[41] GLPK : GNU Linear Programming Kit. http://www.gnu.org/software/glpk/.

[42] GOMP : An OpenMP implementation for GCC. http://gcc.gnu.org/projects/gomp/.

[43] M.A. Grepl. *Reduced-Basis Approximation and A Posteriori Error Estimation for Parabolic Partial Differential Equations*. PhD thesis, Massachusetts Institute of Technology, 2005.

[44] M.A. Grepl, Y. Maday, N.C. Nguyen, and A.T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *Mathematical Modelling and Numerical Analysis*, 41(3) :575–605, 2007.

[45] M.A. Grepl and A.T. Patera. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *Mathematical Modelling and Numerical Analysis*, 39(1) :157–181, 2005.

[46] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM : Mathematical Modelling and Numerical Analysis*, 42(2) :277–302, 2008.

[47] Tristen Hayfield and Jeffrey S. Racine. Nonparametric econometrics : The np package. *Journal of Statistical Software*, 27(5), 2008.

[48] J.C. Helton, J.D. Johnson, C.J. Sallaberry, and C.B. Storlie. Survey of sampling-based methods for uncertainty and sensitivity analysis. *Reliability Engineering & System Safety*, 91(10-11) :1175–1209, 2006.

[49] T. Homma and A. Saltelli. Importance measures in global sensitivity analysis of nonlinear models. *Reliability Engineering & System Safety*, 52(1) :1–17, 1996.

[50] E. Hopf. The partial differential equation $u_t + uu_x = \mu_{xx}$. *Communications on Pure and Applied Mathematics*, 3(3) :201–230, 1950.

[51] D.B.P. Huynh, G. Rozza, S. Sen, and A.T. Patera. A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants. *Comptes Rendus Mathematique*, 345(8) :473–478, 2007.

[52] I.A. Ibragimov and RZ Has' Minskii. *Statistical estimation–asymptotic theory*, volume 16. Springer, 1981.

[53] M. Ilak and C.W. Rowley. Modeling of transitional channel flow using balanced proper orthogonal decomposition. *Physics of Fluids*, 20 :034103, 2008.

[54] B. Iooss. Revue sur l'analyse de sensibilité globale de modèles numériques. *Journal de la Société Française de Statistique*, 152(1) :3–25, 2011.

[55] T. Ishigami and T. Homma. An importance quantification technique in uncertainty analysis for computer models. In *First International Symposium on Uncertainty Modeling and Analysis Proceedings, 1990.*, pages 398–403. IEEE, 1990.

[56] Breton J.-C. Processus gaussiens. 2006.

[57] A. Janon, M. Nodet, and C. Prieur. Certified reduced-basis solutions of viscous Burgers equations parametrized by initial and boundary values. Preprint available at http://hal.inria.fr/inria-00524727/en, 2010, *Accepted in Mathematical modelling and Numerical Analysis.*

[58] A. Janon, M. Nodet, and C. Prieur. Uncertainties assessment in global sensitivity indices estimation from metamodels. Preprint available at http://hal.inria.fr/inria-00567977, 2011, *Accepted in International Journal for Uncertainty Quantification.*

[59] N. Jung, B. Haasdonk, and D. Kroner. Reduced Basis Method for quadratically nonlinear transport equations. *International Journal of Computing Science and Mathematics*, 2(4) :334–353, 2009.

[60] J.P.C. Kleijnen. *Design and analysis of simulation experiments.* Springer Publishing Company, Incorporated, 2007.

[61] D.J. Knezevic, N.C. Nguyen, and A.T. Patera. Reduced basis approximation and a posteriori error estimation for the parametrized unsteady boussinesq equations. *Mathematical Models and Methods in Applied Sciences*, 2010.

[62] D.J. Knezevic and A.T. Patera. A certified reduced basis method for the Fokker-Planck equation of dilute polymeric fluids : FENE dumbbells in extensional flow. *SIAM Journal on Scientific Computing*, 32(2) :793–817, 2010.

[63] M. Lamboni, B. Iooss, A.L. Popelin, and F. Gamboa. Derivative-based global sensitivity measures : general links with sobol'indices and numerical tests. *Arxiv preprint arXiv :1202.0943*, 2012.

[64] B. Laurent. Efficient estimation of integral functionals of a density. *The Annals of Statistics*, 24(2) :659–681, 1996.

[65] WR Madych and SA Nelson. Bounds on multivariate polynomials and exponential error estimates for multiquadric interpolation. *Journal of Approximation Theory*, 70(1) :94–114, 1992.

[66] A. Marrel, B. Iooss, B. Laurent, and O. Roustant. Calculations of sobol indices for the gaussian process metamodel. *Reliability Engineering & System Safety*, 94(3) :742–751, 2009.

[67] G. Matheron. *Les variables régionalisées et leur estimation.* Paris, 1965.

[68] H. Monod, C. Naud, and D. Makowski. Uncertainty and sensitivity analysis for crop models. In D. Wallach, D. Makowski, and J. W. Jones, editors, *Working with Dynamic Crop Models : Evaluation, Analysis, Parameterization, and Applications*, chapter 4, pages 55–99. Elsevier, 2006.

[69] Vlad I. Morariu, Balaji Vasan Srinivasan, Vikas C. Raykar, Ramani Duraiswami, and Larry S. Davis. Automatic online tuning for fast gaussian summation. In *Advances in Neural Information Processing Systems (NIPS)*, 2008.

[70] Thomas Muehlenstaedt, Olivier Roustant, Laurent Carraro, and Sonja Kuhnt. Data-driven Kriging models based on FANOVA-decomposition, 2010. Preprint.

[71] N.C. Nguyen, G. Rozza, and A.T. Patera. Reduced basis approximation and a posteriori error estimation for the time-dependent viscous Burgers' equation. *Calcolo*, 46(3) :157–185, 2009.

[72] N.C. Nguyen, K. Veroy, and A.T. Patera. Certified real-time solution of parametrized partial differential equations. *Handbook of Materials Modeling*, pages 1523–1558, 2005.

[73] J. Nocedal and S.J. Wright. *Numerical optimization*. Springer Verlag, 1999.

[74] A. Nouy et al. Generalized spectral decomposition for stochastic nonlinear problems. *Journal of Computational Physics*, 228(1) :202–235, 2009.

[75] J.E. Oakley and A. O'Hagan. Probabilistic sensitivity analysis of complex models : a bayesian approach. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 66(3) :751–769, 2004.

[76] N.A. Pierce and M.B. Giles. Adjoint recovery of superconvergent functionals from pde approximations. *SIAM Review*, pages 247–264, 2000.

[77] Gilles Pujol, Bertrand Iooss, and Alexandre Janon. *sensitivity : Sensitivity Analysis*, 2012. R package version 1.5.

[78] A. Quarteroni, G. Rozza, and A. Manzoni. Certified reduced basis approximation for parametrized partial differential equations and applications. *Math. Industry*, 2011.

[79] A. Quarteroni, G. Rozza, and A. Manzoni. Certified reduced basis approximation for parametrized partial differential equations and applications. *Journal of Mathematics in Industry*, 1(1) :3, 2011.

[80] A.M. Quarteroni and A. Valli. *Numerical approximation of partial differential equations*. Springer, 2008.

[81] R Core Team. *R : A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. ISBN 3-900051-07-0.

[82] R Development Core Team. *R : A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2011. ISBN 3-900051-07-0.

[83] J. Racine. An efficient cross-validation algorithm for window width selection for nonparametric kernel regression. *Communications in Statistics Simulation and Computation*, 22 :1107–1107, 1993.

[84] D.V. Rovas, L. Machiels, and Y. Maday. Reduced-basis output bound methods for parabolic problems. *IMA journal of numerical analysis*, 26(3) :423, 2006.

[85] G. Rozza and Patera A.T. Venturi : Potential flow. *http://augustine.mit.edu/workedproblems/rbMIT/venturi/*, 2008.

[86] A. Saltelli. Making best use of model evaluations to compute sensitivity indices. *Computer Physics Communications*, 145(2) :280–297, 2002.

[87] A. Saltelli. *Sensitivity analysis : an introduction (tutorial)*, 2004.

[88] A. Saltelli, K. Chan, and E.M. Scott. *Sensitivity analysis*. Wiley : New York, 2000.

[89] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola. *Global sensitivity analysis : the primer*. Wiley Online Library, 2008.

[90] A. Saltelli, S. Tarantola, and K.P.S. Chan. A quantitative model-independent method for global sensitivity analysis of model output. *Technometrics*, pages 39–56, 1999.

[91] A. Saltelli, S. Tarantola, Campolongo F., and Ratto M. Sensitivity analysis in practice : a guide to assessing scientific models, 2004.

[92] T. J. Santner, B. Williams, and W. Notz. *The Design and Analysis of Computer Experiments*. Springer-Verlag, 2003.

[93] T.J. Santner, B.J. Williams, and W. Notz. *The design and analysis of computer experiments*. Springer Verlag, 2003.

[94] R. Schaback. Mathematical results concerning kernel techniques. In *Prep. 13th IFAC Symposium on System Identification, Rotterdam*, pages 1814–1819. Citeseer, 2003.

[95] M. Scheuerer, R. Schaback, and M. Schlather. Interpolation of spatial data – a stochastic or a deterministic problem ? *Preprint, Universität Göttingen. http://num.math.uni-goettingen.de/schaback/research/papers/IoSD.pdf*, 2011.

[96] L. Sirovich. Turbulence and the dynamics of coherent structures. part i-ii. *Quarterly of applied mathematics*, 45(3) :561–590, 1987.

[97] I. M. Sobol. Sensitivity estimates for nonlinear mathematical models. *Math. Modeling Comput. Experiment*, 1(4) :407–414 (1995), 1993.

[98] I.M. Sobol. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation*, 55(1-3) :271–280, 2001.

[99] IM Sobol and S. Kucherenko. Derivative based global sensitivity measures and their link with global sensitivity indices. *Mathematics and Computers in Simulation*, 79(10) :3009–3017, 2009.

[100] C.B. Storlie, L.P. Swiler, J.C. Helton, and C.J. Sallaberry. Implementation and evaluation of nonparametric regression procedures for sensitivity analysis of computationally demanding models. *Reliability Engineering & System Safety*, 94(11) :1735–1763, 2009.

[101] J.C. Strikwerda. *Finite difference schemes and partial differential equations*. Society for Industrial Mathematics, 2004.

[102] B. Sudret. Global sensitivity analysis using polynomial chaos expansions. *Reliability Engineering & System Safety*, 93(7) :964–979, 2008.

[103] S. Tarantola, D. Gatelli, and TA Mara. Random balance designs for the estimation of first order global sensitivity indices. *Reliability Engineering & System Safety*, 91(6) :717–727, 2006.

[104] J.Y. Tissot. *Sur la décomposition ANOVA et l'estimation des indices de Sobol'. Application à un modèle d'écosystème marin.* PhD thesis, Université de Grenoble, 2012.

[105] J.Y. Tissot and C. Prieur. A bias correction method for the estimation of sensitivity indices based on random balance designs. *Reliability engineering and systems safety*, 2010.

[106] J.Y. Tissot and C. Prieur. Variance-based sensitivity analysis using harmonic analysis. Submitted, 2012.

[107] A. Toselli and O.B. Widlund. *Domain decomposition methods–algorithms and theory*. Springer Verlag, 2005.

[108] K. Urban and A.T. Patera. A new error bound for reduced basis approximation of parabolic partial differential equations. *Comptes Rendus Mathematique*, 2012.

[109] A.W. Van der Vaart. *Asymptotic statistics*. Cambridge Univ Press, 2000.

[110] K. Veroy and A.T. Patera. Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations : Rigorous reduced-basis a posteriori error bounds. *International Journal for Numerical Methods in Fluids*, 47(8-9) :773–788, 2005.

[111] K. Veroy, C. Prud'homme, and A.T. Patera. Reduced-basis approximation of the viscous Burgers equation : rigorous a posteriori error bounds. *Comptes Rendus Mathematique*, 337(9) :619–624, 2003.

[112] S. Volkwein. Proper orthogonal decomposition and singular value decomposition. Tech. report Karl-Franzens-Univ. Graz & Techn. Univ. Graz, 1999.

[113] R. Von Mises. Mathematical theory of probability and statistics. *Mathematical Theory of Probability and Statistics, New York : Academic Press, 1964*, 1, 1964.

[114] H. Weyl. Mean motion. *American Journal of Mathematics*, 60(4) :889–896, 1938.

[115] K. Willcox and J. Peraire. Balanced model reduction via the proper orthogonal decomposition. *AIAA journal*, 40(11) :2323–2330, 2002.

[116] Curtis Storlie with contributions from Alexandre Janon. *CompModSA : Sensitivity Analysis for Complex Computer Models*, 2012. R package version 1.3.1-1.

[117] C. Xu and G.Z. Gertner. Reliability of global sensitivity indices. *Journal of Statistical Computation and Simulation*, 81(12) :1939–1969, 2011.

[118] C. Zhu, R.H. Bryd, and J. Nocedal. *L-BFGS-B : Algorithm 778 : L-BFGS-B, FORTRAN routines for large scale bound constrained optimization*, 1997.