Sensitivity analysis of a filtering algorithm for wind lidar measurements

Thomas A Rieutord

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Institut National Polytechnique de Toulouse (INP Toulouse)

Discipline ou spécialité :

Mathématiques Appliquées

Présentée et soutenue par :

M. THOMAS RIEUTORD

le lundi 13 novembre 2017

Titre :

Sensitivity analysis of a filtering algorithm for wind lidar measurements

---

Ecole doctorale :

Mathématiques, Informatique, Télécommunications de Toulouse (MITT)

Unité de recherche :

Groupe de Météorologie Expérimentale et Instrumentale (CNRM-GMEI)

Directeur(s) de Thèse :

M. FABRICE GAMBOA

M. ALAIN DABAS

Rapporteurs :

M. DAVID GINSBOURGER, UNIVERSITE DE BERNE

M. PETER VAN LEEUWEN, UNIVERSITY OF READING

Membre(s) du jury :

M. BERTRAND IOOSS, EDF CHATOU, Président

M. EWAN O'CONNOR, FINNISH METEOROLOGICAL INSTITUTE, Membre

Mme AMANDINE MARREL, CEA CADARACHE, Membre

Contents

Remerciements

Enfin les remerciements ! Vous qui lisez ces lignes, je vous remercie, vous donnez du sens à tout le cœur que j'ai mis à l'ouvrage. J'espère que ça vous plaira, bonne lecture !

J'adresse mes remerciements sincères et chaleureux à Fabrice Gamboa, qui a repris le flambeau avec sourire et efficacité. C'était toujours un plaisir de te voir, même quand j'écrivais des grosses bêtises et malgré la pile de travail avec laquelle je ressortais ! Également, un grand merci à Alain Dabas, qui m'a suivi tout au long de la thèse et même un peu avant... et un peu après. Je suis très enthousiaste à l'idée de continuer avec toi !

Si j'ai eu la chance de faire une thèse, c'est grâce à la préparation et l'implication de Christophe Baehr. Je le remercie sincèrement pour cela.

Je remercie Claire Doubremelle et Mihai Maris, de m'avoir écouté puis agi avec tact et efficacité lorsque la situation l'imposait.

Merci mille fois Lucie, a.k.a. "le pompier de GMEI". Ton soutien, ta solidarité et ton sens de l'humour m'ont été infiniment précieux. J'espère qu'on continuera à travailler ensemble quand tu seras eu GMAP.

Merci aussi à Cécile et Antoine, mes compagnons de thèse en première année. J'ai vu sur eux les ravages de la rédaction... mais aussi le soulagement providentiel des pots de thèse !

Je remercie les autres chercheuses de l'équipe, Pauline, Guylaine et Cyrielle, pour leurs relectures, leurs avis sur tout ce qui pourrait fonctionner mieux et leur soutien quand, vraiment, ça aurait dû fonctionner beaucoup mieux !

Je remercie aussi tous mes collègues du café GMEI, souvent rallongé par l'actualité : Bruno, William, Patrick, Jacques, Sandrine, Tiphaine, Rémi, Vinciane, Jean-Marie, Thierry, Guylaine, Anne, Éric, Joël, Gilles, Diane, Patrick, Renaud... (et j'en sûrement oublié, désolé).

Je remercie chaleureusement mes parents. Malgré leur déconnexion totale avec le monde universitaire, leurs conseils et leur expérience m'ont toujours servis. Merci Claire et Maxence de m'avoir fait tonton récemment ! Merci aussi à ma grand-mère Raymonde, à qui je souhaite de garder encore longtemps sa vigueur exceptionnelle. Merci à Cathy, ma tante barrée qui a toujours cru en moi. Merci à toute la famille, qui restera ma ressource première (et c'est heureux qu'elle soit installée si près du Mont Aigoual, sans quoi je n'aurais peut-être jamais fait de météo).

Merci les copains, Yannick, Micka, Élise, Léo, Manu, Matthieu, Pierre-Emmanuel, Marc-Antoine, Maxime, Maxime, Océane, Marion, Camille, Camille, Linda, Boris. Malgré des névroses alimentaires assez sévères chez certains, ça a toujours été facile de se retrouver autour d'une table, que ce soit pour manger une petite blanquette, pour une partie de coinche ou pour "juste une bière" ! J'espère que ça va continuer encore longtemps.

Là, au milieu de mes cartons à une heure pas possible, je sens vraiment que c'est la fin de quelque chose. Mes pensées vont à mes colocataires, Bastien, Joana, Laurent (c'est comme si), Matthieu, Manuela, Clothilde, Mercedes, Léo, Jérémy. Même si je ne compte pas arrêter de cuisiner, je pense pas que je pourrai retrouver cette saveur insouciante de grand n'importe quoi qu'il y avait dans le pesto maison, la ratatouille ou la raclette. Merci à vous pour ces moments là.

Enfin merci Marine, tu as été en première ligne pendant presque tous les rebondissements de cette thèse, sans pour autant faire de moins bons gâteaux ! Blague à part, c'était toujours un plaisir de te retrouver le soir en rentrant et je te remercie du fond du cœur.

Je crois que je vais m'arrêter là, mes rapporteurs vont sûrement déjà soupirer en voyant l'épaisseur du bouquin...

Acronyms table

CNRM	Centre national de recherches météorologiques
IMT	Institut de Mathématiques de Toulouse
CDF	Cumulative distribution function
PDF	Probability density function
SIS	Sequential importance sampling
SIR	Sampling with importance resampling
TKE	Turbulent kinetic energy
EDR	Eddy dissipation rate
TTKE	Time turbulent kinetic energy
STKE	Spatial turbulent kinetic energy
LSTKE	Local spatial turbulent kinetic energy
LES	Large eddy simulation

Résumé substansif en français

Ce manuscrit présente les travaux effectués pendant les trois ans de mon doctorat sur le sujet suivant : *Analyse de sensibilité d'une méthode de filtrage des mesures de vent par lidar*. Les lidars sont des instruments de télédétection utilisés pour mesurer le vent en météorologie. Pour améliorer les mesures, une méthode de filtrage originale a été développée, mais son comportement est encore mal connu. L'analyse de sensibilité permet de quantifier l'influence des paramètres sur le code. Le travail présenté ici consiste à faire l'analyse de sensibilité de la méthode de filtrage originale. Il se compose de 8 chapitres principaux, encadrés par une introduction et une conclusion générales :

1. Turbulence dans la couche limite atmosphérique
2. Filtrage bayésien
3. Instrument et données
4. Théorie de l'analyse de sensibilité
5. Reconstruction de milieu turbulent
6. Résultats de l'analyse de sensibilité
7. Exploration avec les expériences 2-à-2
8. Régressions pénalisées pour l'estimation des indices de Sobol

Ces chapitres sont complétés par 3 annexes :

A. Rappels théoriques
B. Preuves
C. Résultats complets d'expériences 2-à-2

Chapitre 1 : Turbuluence dans la couche limite atmosphérique

Le premier chapitre est consacré à la turbulence atmosphérique. Il rappelle des bases afin de parler à des étudiants et/ou des novices. Le système d'équations de Navier-Stokes pour l'atmosphère est rappelé et simplifié pour l'étude de la couche limite atmosphérique (c'est-à-dire le premier kilomètre d'atmosphère, siège de turbulence de part l'influence du sol). Le

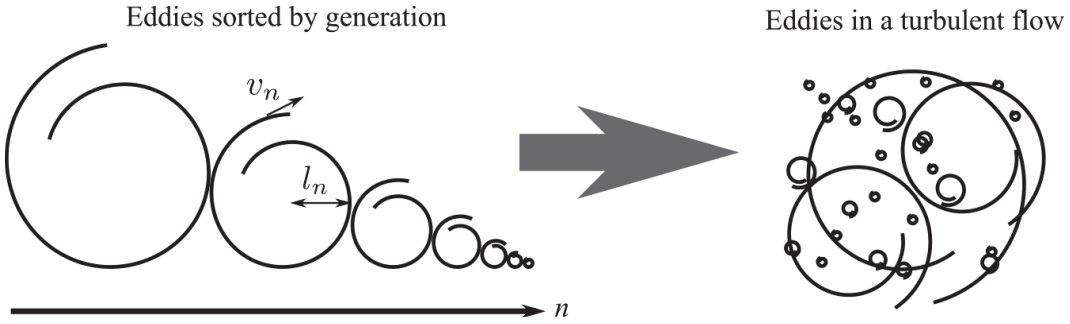
vent est la grandeur cible puisque c'est la seule mesurée. La particularité de la turbulence est qu'elle sépare le vent en une tendance déterministe et une fluctuation aléatoire.

$$\Psi = \overline{\Psi} + \Psi'$$

Le système de Reynolds ainsi obtenu est celui utilisé dans les paramétrisations de modèles numériques de prévision du temps. Certains termes revêtent une importance particulière, de part leur interprétation et leur place dans les équations. Il s'agit de l'énergie cinétique turbulente (TKE pour "turbulent kinetic energy") et son taux de dissipation (EDR pour "eddy dissipation rate"). La TKE, k, est l'énergie contenue par les fluctuations du vent, on peut la voir comme un indicateur de la "quantité de turbulence".

$$k = \frac{1}{2}\left(\overline{u'^2 + v'^2 + w'^2}\right)$$

L'EDR est le taux de destruction des tourbillons. Or les tourbillons se détruisent en tourbillons plus petits, détruit à leur tour, et ainsi de suite jusqu'à ce qu'il se dissipent en chaleur. L'EDR est donc aussi le taux de transfert des grands tourbillons vers les plus petits. Ce transfert d'énergie entre les tourbillons de différentes tailles crée une continuité dans les échelles de la turbulence.



Cette continuité donne des fonctions de structures (variance d'un incrément de vent) de forme exploitable, ce qui se manifeste par un spectre du vent décroissant avec une pente constante de -5/3 dans la zone inertielle. La pente du spectre est un critère de vérification que le vent est bien représentatif de la turbulence. Les fonctions de structures sont exploitées dans leur version lagrangiennes pour établir un modèle lagrangien stochastique de turbulence. Le modèle lagrangien stochastique décrit l'évolution de particules fluides de façon lagrangienne. C'est un mélange de l'équation de Langevin et des fonctions de structures données par la théorie de Kolmogorov pour la turbulence. Les représentations eulériennes et lagrangiennes donnent plusieurs voies d'estimation des quantités de la turbulence. Pour réunir la moyenne eulérienne et lagrangienne, un noyau de régularisation gaussien de paramètre ℓ est utilisé. Différents estimateurs de TKE sont définis et comparés: la TKE avec une variance temporelle (TTKE), la TKE avec une variance spatiale sur un niveau vertical (STKE), la TKE avec une variance spatiale utilisant le noyau de régularisation gaussien. Dans le cas 1D vertical (le seul traité dans le manuscrit), ils ne sont pas comparables car l'hypothèse de Taylor sur la turbulence gelée n'est pas vérifiée. Cette hypothèse dit que les tourbillons mesuré à deux

instants proches sont deux tourbillons distincts advecté par le vent moyen. Or le vent vertical moyen est nul, il ne peut donc pas advecter les tourbillons. Il en résulte une déconnexion des variations spatiales et temporelles.

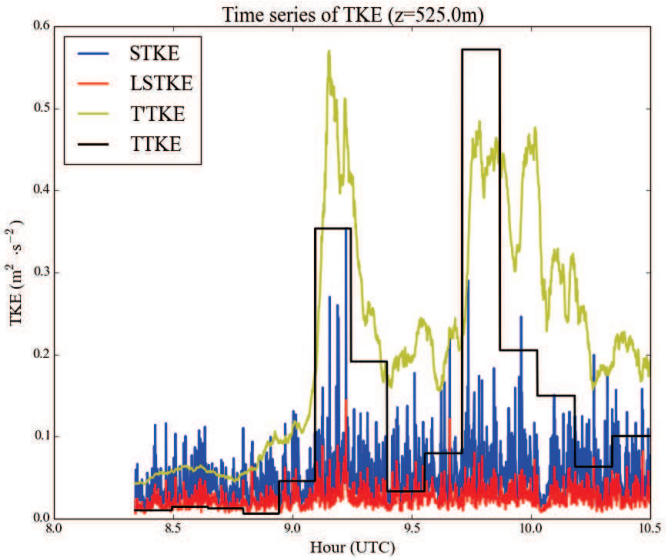


Figure 1 – Comparaison des estimateurs de TKE. Les variances spatiales et temporelles ne sont pas comparables.

Chapitre 2 : Filtrage bayésien

Le second chapitre, sobrement nommé "Filtrage bayésien", éclaire sur les origines de la méthode de filtrage étudiée par l'analyse de sensibilité. En effet, la reconstruction de milieu turbulent est issue de la communauté du filtrage non-linéaire via Christophe Baehr et Pierre Del Moral. Le chapitre 2 donne un tour d'horizon du filtrage bayésien, linéaire puis non-linéaire. L'idée globale du filtrage bayésien est de donner la meilleure estimation d'un processus cible, X_t, à partir d'une série d'observation $(Y_0, ..., Y_t)$ et d'un modèle imparfait d'évolution de X_t. À la fois le modèle et les observations sont sources d'erreurs. Il y a deux étapes dans le filtrage bayésien (dont les noms varient suivant les auteurs) : la mutation et la sélection. La mutation consiste à prédire l'état suivant X_{t+1} à partir de X_t avec le modèle. Elle donne l'état *a priori* du processus cible. La sélection consiste à corriger cet état en fonction des observations. Elle donne l'état *a posteriori* du processus cible.

On parle de filtrage linéaire lorque le modèle est linéaire et que l'opérateur d'observation est linéaire. Le filtre de Kalman donne une solution optimale au problème de filtrage linéaire avec des bruit gaussiens indépendant. La performance du filtre de Kalman a poussé la communauté à l'adapter aux cas non-linéaires. La première généralisation, l'EKF, qui consiste à utiliser une approximation de Taylor d'ordre 1 pour linéariser le problème ne donne pas de très

bon résultats (instabilité numérique, cout de calcul important). La généralisation suivante, l'UKF, utilise la transformation "unscented" qui a donné son nom au filtre et qui approxime la loi de probabilité plutôt que la transformation non-linéaire. Cette solution est encore très appliquée car elle donne de bons résultats dans une large gamme de problèmes. Son amélioration, l'EnKF, va plus loin dans l'approximation de la loi de probabilité en utilisant une méthode de Monte Carlo. Bien que plus gourmand en calcul, l'EnKF est une solution dans les cas fortement non-linéaire où l'UKF atteint ses limites. Les filtres à particules se basent sur la méthode de Monté Carlo dès la base du problème de filtrage. La similarité entre EnKF et filtre à particules n'est qu'apparente et leurs différences sont soulignées. L'étape de sélection dans un filtre à particules peut prendre différentes formes, correspondant à différent algorithmes. L'algorithme SIS pondère les particules par leur vraisemblance mais ne les modifient pas. L'algorithme SIR remplace la pondération par un rééchantillonage des particules suivant leur vraisemblance. L'algorithme de sélection génétique exécute l'algorithme SIR sur seulement une partie des particules, choisies aléatoirement suivant leur vraisemblance. Pour le filtrage des mesures de vent, l'algorithme de sélection génétique est utilisé.

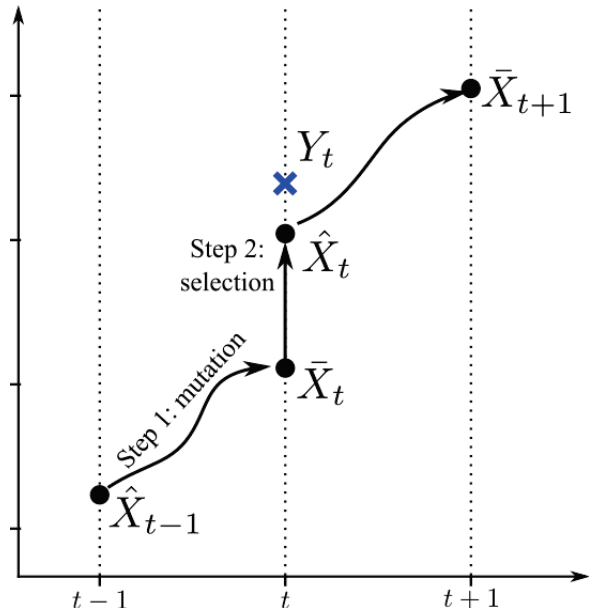


Figure 2 – Illustration des deux étapes du filtrage bayésien: mutation et sélection.

Chapitre 3 : Instrument et données

Le troisième chapitre est consacré à la technologie lidar et aux données qu'il produit. Le lidar émet une impulsion lumineuse dans l'atmosphère. Une partie de cette lumière est renvoyée par les particules (poussières, fumées, embruns...) présentes dans l'atmosphère. Lorsque les particules qui renvoient l'onde lumineuse bougent dans l'axe du rayon, la fréquence de l'onde incidente est modifiée par effet Doppler. La fréquence de la lumière récupérée sur le lidar sera

augmentée (resp. diminuée) si les particules rétro-diffusantes se déplacent vers le lidar (resp. s'éloignent du lidar), ce qui permet de mesurer le vent. Pour mesurer le décalage en fréquence, on utilise une détection hétérodyne. Le principe est de mélanger sur le récepteur le signal revenant de l'atmosphère et un oscillateur local. L'interférence entre l'oscillateur local et le signal reçu rend le décalage Doppler facilement isolable sur un spectre. Malheureusement, pour calculer un spectre correct (via une transformée de Fourier rapide) il faut un certain nombre de points. Plus on accumule de point pour le spectre, plus la l'onde lumineuse s'éloigne. La mesure est donc intégrée sur une porte, dont la taille est un compromis entre la qualité du spectre et la résolution spatiale. Pour le lidar utilisé, la résolution spatiale est de 50 mètres. L'explication du principe de mesure permet de se rendre compte de deux limitations importantes : seul la composante du vent dans l'axe du rayon est mesurée et le vent est intégré sur une certaine distance.

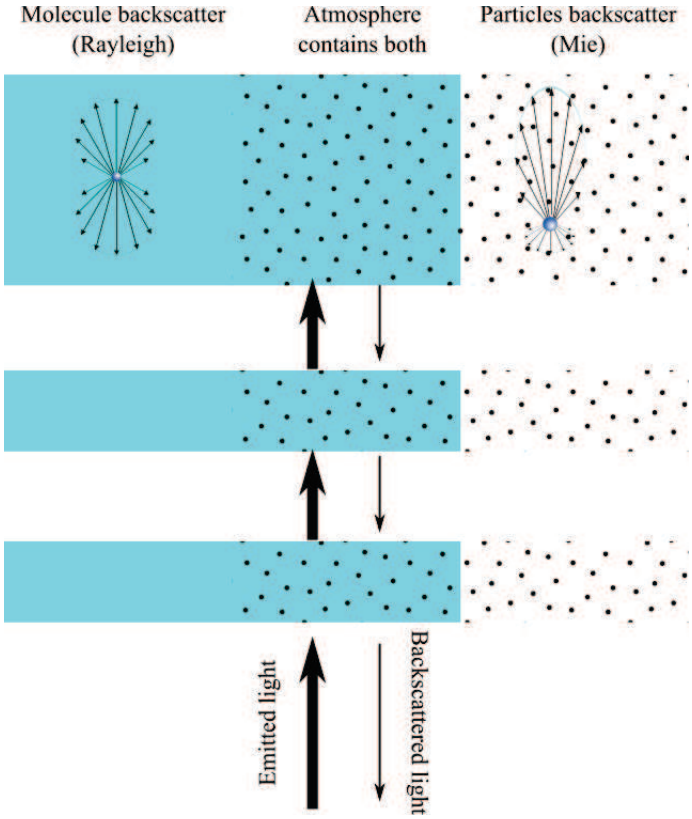


Figure 3 – Illustration de la rétrodiffusion dans l'atmosphère. C'est le principe utilisé par le lidar pour mesurer le vent.

Les données utilisées proviennent d'une campagne de terrain qui a eu lieu lors de l'été 2011 à Lannemezan (Hautes Pyrénées). Le projet de recherche associé, BLLAST (Boundary Layer Late Afternoon and Sunset Turbulence), a pour but de caractériser la turbulence dans la couche limite atmosphérique lors de la transition du coucher de soleil. C'est une transition difficile à appréhender car la source d'énergie pricipale (le soleil) se tarit, ce qui modifie les flux et la circulation sur la verticale. Les données choisies pour tester la reconstruction correspondent au domaine de pertinence du modèle lagangien stochastique : turbulence sèche,

bien développée. Il s'agit des mesures du 30 juin 2011, entre 14h42 to 16h45 (heure locale). Seules les mesures de vent vertical sont utilisées.

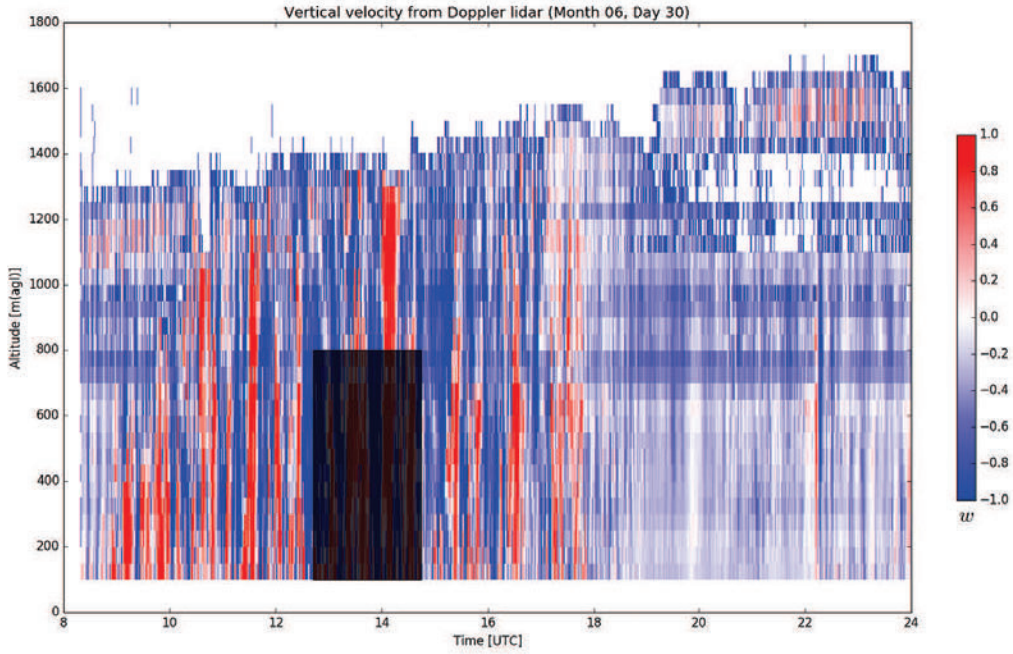


Figure 4 – Données mesurées par le lidar le 30 juin 2011 à Lannemezan. La partie utilisée pour cette étude est grisée.

Pour tester la méthode de reconstruction, nous voulions la tester sur le plus grand jeu de données possible. Le problème des données manquantes s'est alors posé. La solution retenue a été incluse dans le brevet (Baehr et al., 2016). D'abord nous distinguons plusieurs type de données manquantes: mesures irrégulières, valeurs manquantes à cause de l'instrument, valeurs manquantes aléatoirement. Les mesures irrégulières sont des arrêts non-signalés de l'instrument. Les valeurs manquantes à cause de l'instrument sont dues à ses limitations connues. Les valeurs manquantes (aléatoirement ou à cause de l'instrument) sont signalées (par des NaN). Ensuite, les mesures irrégulières sont remplies de valeurs manquantes. Si une plage de valeur manquantes est trop importante, le système redémarre à la prochaine donnée. Si les valeurs manquantes ne sont pas en trop grand nombre, l'étape de sélection est sautée et les estimations sont faites différemment.

Chapitre 4 : Théorie de l'analyse de sensibilité

Le chapitre 4 est une introduction à l'analyse de sensibilité basée sur la variance, celle qui a été faite ici. Les autres manières de faire l'analyse de sensibilité sont brièvement évoquées, puis les différents théorèmes sur lesquels se basent l'analyse de sensibilité. En guise de formation,

les démonstrations ont été refaites, elles sont jointes en annexe. L'analyse de sensibilité basée sur la variance est fondée sur une décomposition de variance très générale. La sortie du code est vue comme une fonction des variables d'entrée.

$$Y = f(X)$$

avec $Y \in \mathbb{R}$ et $X \in \mathbb{R}^p$. Cette fonction est décomposée par projections sur des espaces orthogonaux : les espaces de Hoeffding. La sortie s'écrit alors

$$\begin{aligned} Y &= \sum_{\mathbf{u} \in I} f_{\mathbf{u}}(X_{\mathbf{u}}) \\ &= f_{\emptyset} + \sum_{i=1}^{p} f_i(X_i) + \sum_{1 \leqslant i < j \leqslant p} f_{ij}(X_i, X_j) + \cdots + f_{1,\ldots,p}(X_1, \ldots, X_p) \end{aligned}$$

La variance de la sortie est ainsi exprimée comme une somme de termes attribués à chaque groupe de variable d'entrée $\mathbf{u}$. Le nombre de variable d'entrée comprises dans un groupe est appelé l'ordre du groupe (noté $|\mathbf{u}|$). On en déduit des indices de sensibilité : les indices de Sobol. La décomposition de Hoeffding définit les indices de Sobol simples. Elle peut s'appliquer à n'importe quel code, pourvu que sa variance soit finie. Depuis les indices de Sobol simples, on définit les indices de Sobol complets (qui incluent les contributions d'ordre inférieur des membres du groupe) et les indices de Sobol totaux (qui incluent les contributions à tout ordre de tous ses membres).

$$D_{\mathbf{u}} = V(f_{\mathbf{u}}(X_{\mathbf{u}})) = V(Y)\, S_{\mathbf{u}}$$

$$D_{\mathbf{u}}^{T} = \sum_{\substack{\mathbf{v} \in I \\ \mathbf{v} \cap \mathbf{u} \neq \emptyset}} D_{\mathbf{v}}$$

$$D_{\mathbf{u}}^{C} = \sum_{\substack{\mathbf{v} \in I \\ \mathbf{v} \subseteq \mathbf{u}}} D_{\mathbf{v}}$$

Dans cette thèse, seuls les indices simples d'ordre 1 et 2 et les indices totaux d'ordre 1 sont estimés. Plusieurs estimateurs existent. Les estimateurs utilisés sont précisés dans la table suivante :

Formule de l'estimateur	**Référence**
$\widehat{D_i} = \frac{1}{m} \sum_{e=1}^{m} f(\mathbf{x}^e)(f(\mathbf{z}_i^e, \mathbf{x}_{\tilde{i}}^e) - f(\mathbf{z}^e))$	(Saltelli et al., 2010) eq. (16)
$\widehat{D_i^T} = \frac{1}{2m} \sum_{e=1}^{m} (f(\mathbf{z}_i^e, \mathbf{x}_{\tilde{i}}^e) - f(\mathbf{x}^e))^2$	(Saltelli et al., 2010) eq. (19)
$\widehat{D_{ij}} = \frac{1}{m} \sum_{e=1}^{m} \left(f(\mathbf{z}_i^e, \mathbf{x}_{\tilde{i}}^e) f(\mathbf{z}_j^e, \mathbf{x}_{\tilde{j}}^e) - f(\mathbf{x}^e) f(\mathbf{z}^e) \right) - \widehat{D_i} - \widehat{D_j}$	(Saltelli, 2002a)

L'estimation directe des indices de Sobol est trop couteuse en temps de calcul. Ce problème se pose pour beaucoup de systèmes dont on voudrait faire l'analyse de sensibilité. La stratégie classique consiste à émuler le système grâce à des méthodes statistiques. A partir d'une

surface de réponse donnée (aussi grande que le calcul le permet), on construit un estimateur statistique que l'on peut ensuite utiliser pour prédire la réponse du système à très faible coût. L'estimateur statistique est appelé méta-modèle. Dans notre cas, il est obtenu par krigeage avec des processus gaussiens. Le variogramme a été choisi gaussien après examen des nuées variographiques. Toutes les expériences numériques d'analyse de sensibilité ont été effectuées avec les packages Python opensource `sklearn` pour le krigeage par processus gaussien et `SALib` pour l'estimation des indices de Sobol. Les résultats sont présentés dans le chapitre 7.

Chapitre 5 : Reconstruction de milieu turbulent

Le chapitre 5 est au cœur du manuscrit : il imbrique les chapitres précédent pour établir la méthode de reonctruction de milieu turbulent. La "reconstruction de milieux turbulents" désigne une nouvelle méthode de post-traitement des mesures rapides de vent, qui permet de filtrer les mesures et d'estimer des paramètres de la turbulence en temps réel. L'atmosphère y est représentée non pas par un modèle en point de grille mais par un ensemble de particules lagrangiennes. Ces particules lagrangiennes sont guidées par le modèle stochastique lagrangien présenté au chapitre 1. Elles sont aussi vues comme un ensemble de réalisation de la variable aléatoire "vent", ce qui permet d'utiliser les méthodes de filtrage présentées au chapitre 2. Enfin, la technique est appliquée aux données lidar, présentées au chapitre 3.

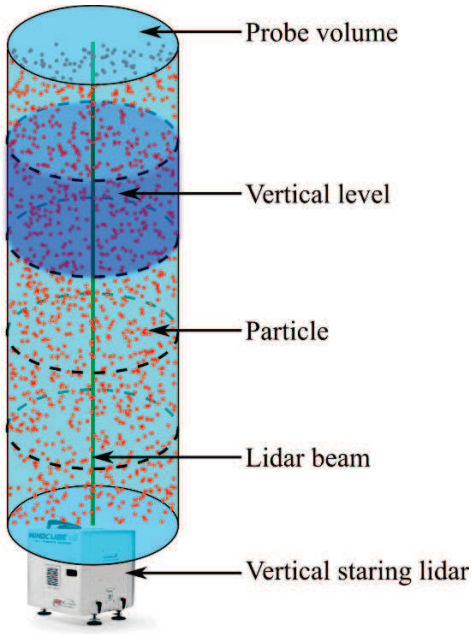


Figure 5 – Géométrie du problème de reconstruction 1D et vocabulaire.

La reconstruction de milieu turbulent est une méthode itérative qui répète quatre étapes à chaque nouvelle observation : mutation, conditionnement, sélection, estimation. La mutation prédit la position et la vitesse des particules au pas de temps suivant avec le modèle stochastique lagrangien. Le conditionnement s'assure que les particules sont correctement

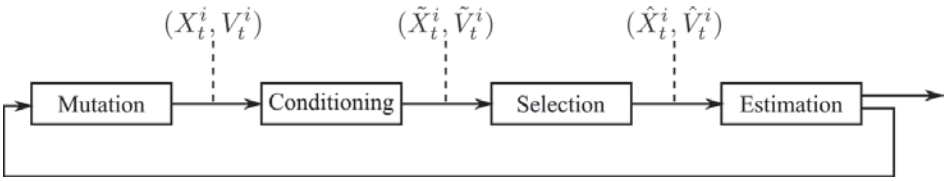


Figure 6 – Évolution du vecteur d'état des particules lors d'une itération.

réparties dans le volume de mesure. Deux choses sont vérifiées : qu'aucune particule n'est restée hors du volume de mesure et qu'il n'y a pas de concentration/vide de particule important. La sélection incorpore l'observation dans le système de particules via l'algorithme de sélection génétique : les particules prochent de l'observation restent à leur place, les autres sont rééchantillonées (suivant l'algorithme SIR). L'estimation ne modifie pas le vecteur d'état mais l'utilise pour mettre à jour les quantités utilisées dans le modèle lagrangien stochastique et estimer les variables de sortie (vent, TKE). Chaque étape est détaillée et illustrée. Les paramètres qui apparaissent çà et là sont répertoriés car ils seront les entrées de l'analyse de sensibilité.

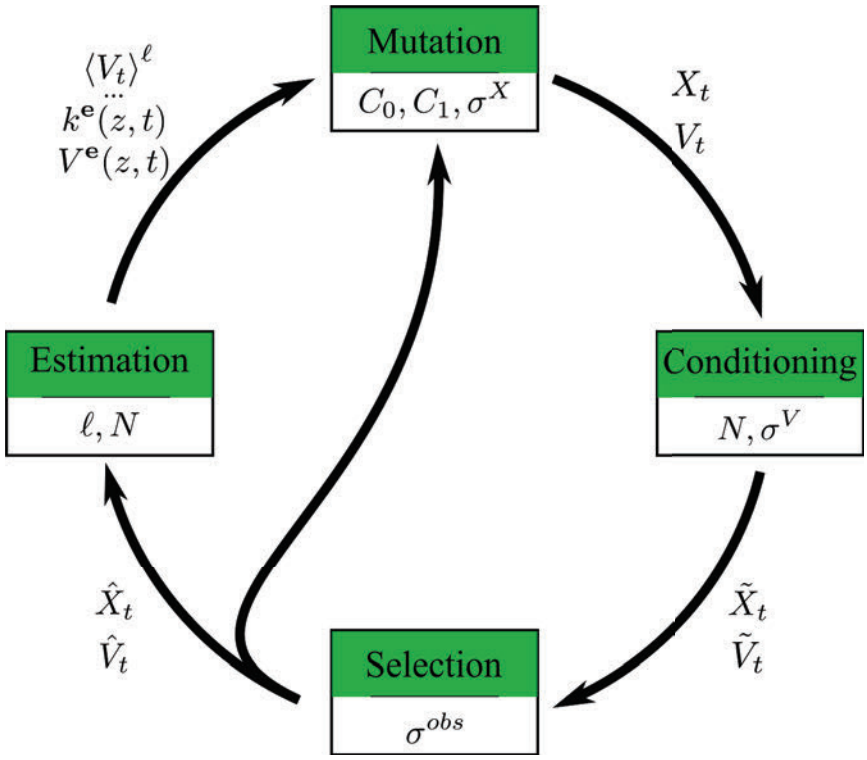


Figure 7 – Organisation des quatre étapes d'une itération de la méthode de reconstruction avec pour chaque étape les paramètres qui l'influence.

Pour faire l'analyse de sensibilité, une expérience de validation est mise en place et des scores sont définis. Cinq scores sont les sorties du système soumis à l'analyse de sensibilité. Le nombre de potentiels nuls, N_{G0}, compte le nombre de fois où la sélection ne peut pas se

faire, faute de particules proche de l'observation. De tels cas sont plus où moins fréquents suivant les réglages. Quand ils sont nombreux (N_{G0} élevé), c'est le symptôme de dégénération du filtre : les particules partent à l'infini à cause de mauvais réglages. La pente du spectre de vent, b, a une valeur théorique de -5/3 dans la turbulence inertielle. Lorsqu'il reste du bruit dans le signal, le spectre remonte en hautes fréquences, ce qui augmente b. A l'inverse, lorsque le filtre fait trop de zèle, b diminue en dessous de -5/3. L'erreur quadratique moyenne (RMSE) sur la TKE, r_k, est un score de comparaison entre la LSTKE (calculée grâce aux particules à haute cadence) et la TTKE (calculée sans particules de façon classique). Si la LSTKE est comparable à la TTKE, ce score devrait être minimum pour les bons réglages. L'erreur quadratique moyenne sur le vent, r_V, est un score de comparaison entre un vent de référence et le vent estimé à l'issue du filtrage. Si le bruit est bien retiré, ce score devrait être bas. Le temps d'exécution chronomètre l'exécution d'un seul run car c'est une contrainte pratique importante.

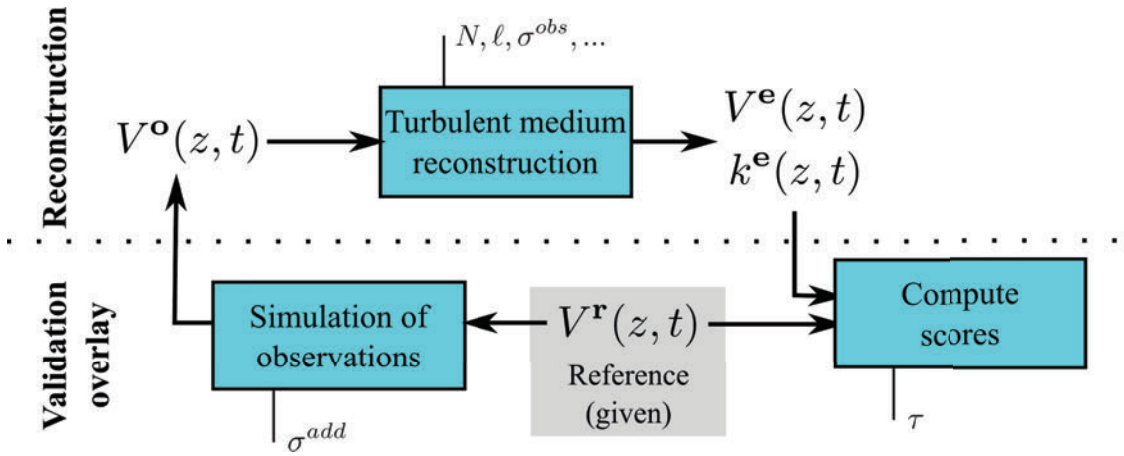


Figure 8 – Schéma de l'expérience de validation du système de reconstruction qui permet de définir les scores.

Pour certains de ces scores, des résultats théoriques sont connus. La décroissance de l'erreur sur le vent en racine du nombre de particules a été établie par Baehr (2010); **?**. La décroissance exponentielle du nombre de potentiel nul avec le nombre de particules s'établie facilement depuis un le théorème 7.4.1 de Del Moral (2004). On établit dans ce manuscrit le résultat suivant :

Theorem 0.1 (Influence de σ^{obs} et σ^{add} sur N_{G0})**.** *Si les hypothèses suivantes sont satisfaites*

- *Le vent réel $V^{\mathbf{r}}_{z,t}$ est stationnaire d'order 2 et ergodique en temps et en espace.*
- *Les particules après l'étape de conditionnement $\tilde{V}_t$ sont gaussiennes avec la même moyenne et variance que $V^{\mathbf{r}}_{z,t}$.*

Alors, le nombre moyen de potentiel nuls N_{G0} est majoré par une fonction de σ^{obs} et σ^{add}.

$$\mathbb{E}\left[N_{G0}\right] \leqslant N_t N_z \frac{(\sigma^{add})^2 + 2k}{-((\sigma^{obs})^2 + 2k)\log\left(\iota^2 2\pi((\sigma^{obs})^2 + 2k)\right)} \tag{1}$$

avec $\iota = 10^{-16}$, le seuil pour le zero machine, et $k = \frac{1}{2}V\left(V^{\mathbf{r}}_{z,t}\right)$ (constante de part l'hypothèse de stationnarité).

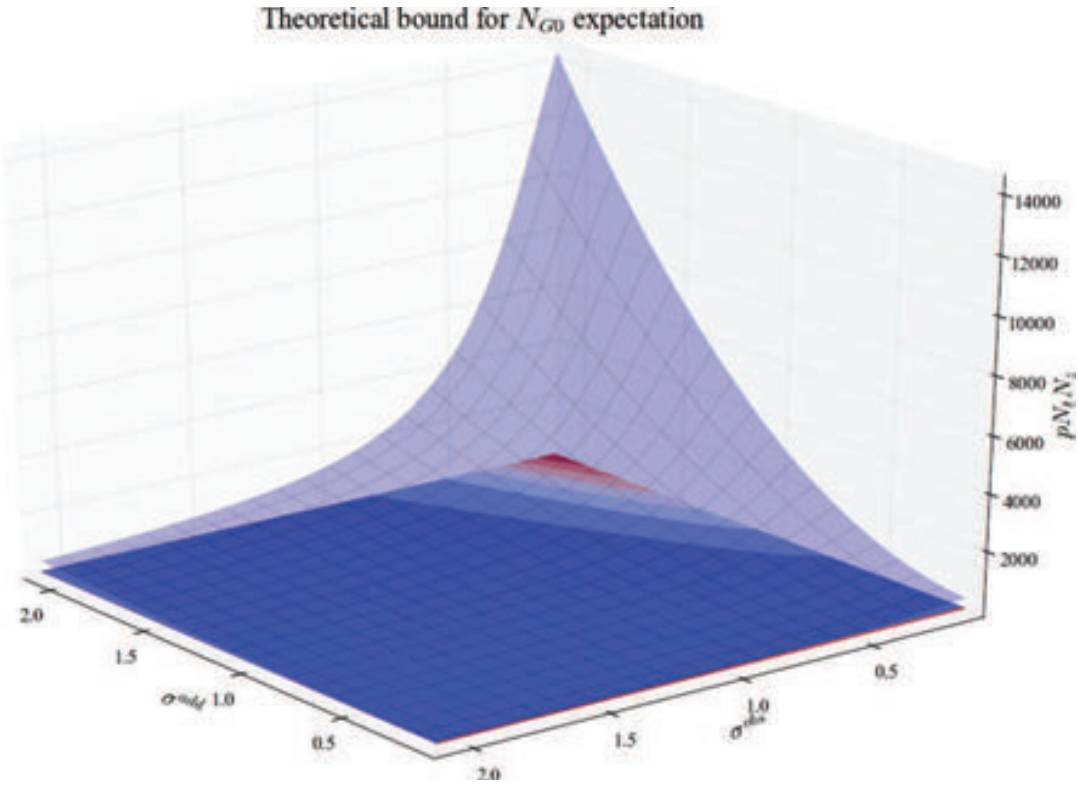


Figure 9 – Forme du majorant de l'espérance de N_{G0} suivant σ^{add} et σ^{obs}.

Chapitre 6 : Résultats de l'analyse de sensibilité

Le chapitre 6 présente les résultats de l'analyse de sensibilité et interprète les résultats. On rappelle que les indices de Sobol sont estimés à partir des prédictions d'un méta-modèle

gaussien et que les estimateurs utilisés sont présentés au chapitre 4. Le système sur lequel est fait l'analyse de sensibilité inclut des mesures de bonnes qualités prises comme référence. Les sorties sont les scores calculés grâce à la référence (introduits au chapitre 5). Les entrées sont les paramètres qui apparaissent dans la description de la reconstruction au chapitre 5 complétés par le bruit d'observation réel, σ^{add} (puisque les observations sont simulées depuis la référence) et par le temps d'intégration, τ, qui permet de calculer le RMSE sur la TKE. Au total, neuf paramètres sont en entrée:

- C_0 constante de Kolmogorov, dans le modèle lagrangien.
- C_1 coefficient devant la fluctuation, dans le modèle lagrangien.
- ℓ longueur d'interaction, dans la moyenne locale avec un noyau de régularisation gaussien.
- N le nombre total de particules.
- σ^{add} le bruit d'observation réel, ajouté à la référence pour simuler les observations.
- σ^{obs} le bruit d'observation donné au filtre, utilisé à l'étape de sélection.
- σ^{V} la variance de vent par défaut, utilisé pour éviter les paquets de particules à l'étape de conditionnement.
- σ^{X} l'erreur attribuée à la discrétisation par un schéma d'Euler explicite, à l'étape de mutation.
- τ le temps d'intégration, utilisé pour calculer la TTKE depuis la référence et moyenner la LSTKE.

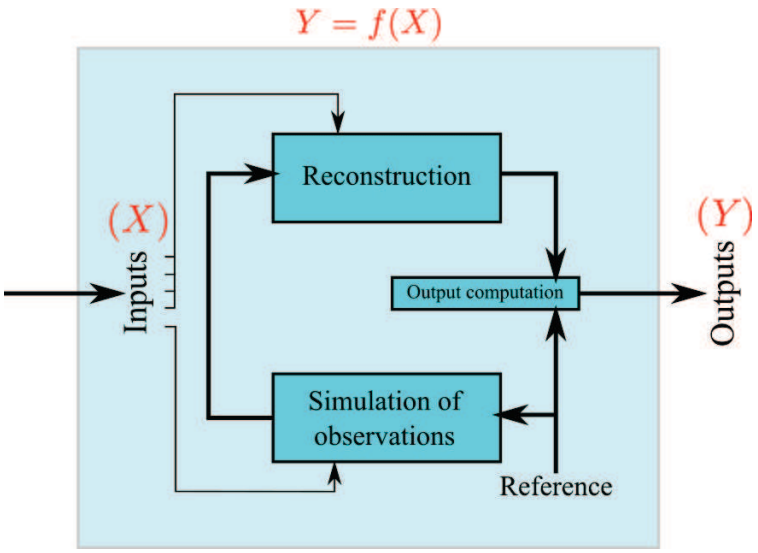


Figure 10 – Schéma du système sur lequel est fait l'analyse de sensibilité.

Des graphiques variés sont utilisés pour étayer les commentaires. D'abord, les sorties sont analysées une par une. Le temps d'exécution, T_{exe}, n'est influencé que par le nombre

de particules, N. La pente du spectre du vent, b, est influencé par les bruits d'observation σ^{add} et σ^{obs} (respectivement le bruit réel et bruit donné au filtre). L'erreur sur le vent, r_V, est influencée par le bruit d'observation réel, σ^{add}, le nombre de particules, N, la constante de Kolmogorov, C_0 et le bruit d'observation donné au filtre, σ^{obs}. L'erreur sur le TKE et le nombre de potentiel nuls sont influencés par beaucoup de variables dont une large part d'interactions. Les interactions impliquent des effets secondaires lors du réglage des entrées. Par conséquent, ces deux sorties sont difficilement utilisables pour régler les entrées.

Ensuite, les sorties sont prises toutes ensembles pour adopter un point de vue global. Les indices de Sobol des cinq entrées sont sommés. On peut les voir comme les marqueurs d'influence sur une sortie globale qui pondère les cinq sorties par 1. Dans ce résumé, seul le pavage des indices simple d'ordre 1 et 2 et le graphe d'interaction sont montrés. Sur chacun d'eux, les entrées qui ressortent sont N, σ^{add} et σ^{obs}, surtout de part leur influence seuls (leur indice de Sobol total est presque égal à leur indice de Sobol simple). Dans un second temps, C_0 et C_1 sont bien visibles aussi, mais leur part d'interaction est beaucoup plus marquée. Enfin σ^X, ℓ, σ^V, τ ressortent peu, malgré que σ^X et ℓ se remarquent mieux de part leur influence en interaction. On peut donc dresser le tableau suivant, où les entrées sont regroupées en groupe d'influence comparable.

Overall influence

Simple	N	σ^{add}	σ^{obs}	C_0	C_1	σ^X	ℓ	τ	σ^V
	0.223	0.188	0.115	0.042	0.034	0.016	0.007	-0.002	-0.004
Total	σ^{add}	N	σ^{obs}	C_0	C_1	σ^X	ℓ	σ^V	τ
	0.373	0.322	0.235	0.215	0.158	0.115	0.074	0.04	0.038
	Groupe 1			Groupe 2		Groupe 3		Groupe 4	

Table 1 – Indices de Sobol d'ordre 1 moyens des cinq sorties.

L'analyse des sorties permet d'identifier celles qui sont le plus pertinentes pour faire un réglage du système : la pente du spectre du vent, b, l'erreur sur le vent, r_V et le temps d'exécution, T_{exe}. L'analyse des entrées fait ressortir des groupes de paramètres d'entrée d'influence comparable, dont le premier contient le bruit d'observation réel, σ^{add}, le bruit d'observation donné au filtre, σ^{obs} et le nombre de particules, N. Ces résultats permettent de réduire le système initial, assez complexe, à un système simple de 3 entrées et 3 sorties.

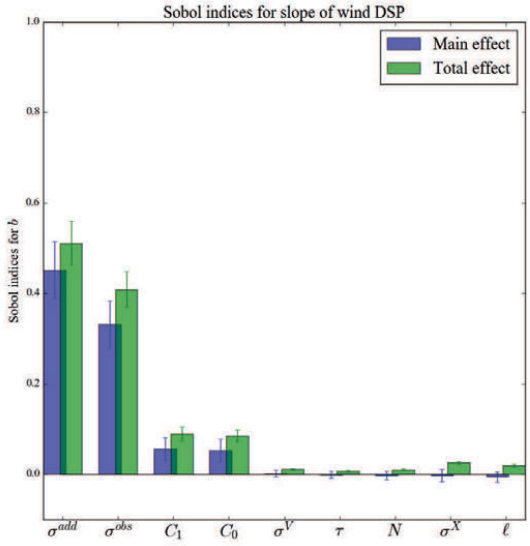


Figure 11 – Indices de Sobol d'ordre 1 : simple (blue) et total (green) pour la pente du spectre du vent.

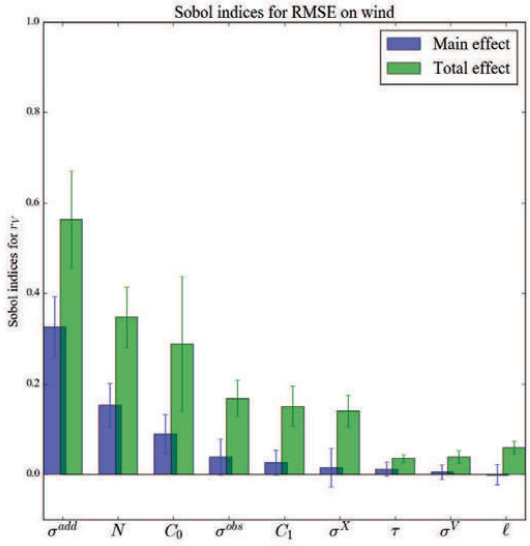


Figure 12 – Indices de Sobol d'ordre 1 : simple (blue) et total (green) pour l'erreur sur le vent.

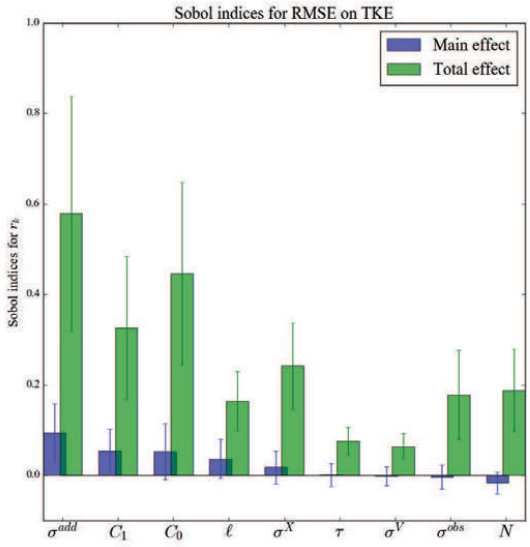


Figure 13 – Indices de Sobol d'ordre 1 : simple (blue) et total (green) pour l'erreur sur la TKE.

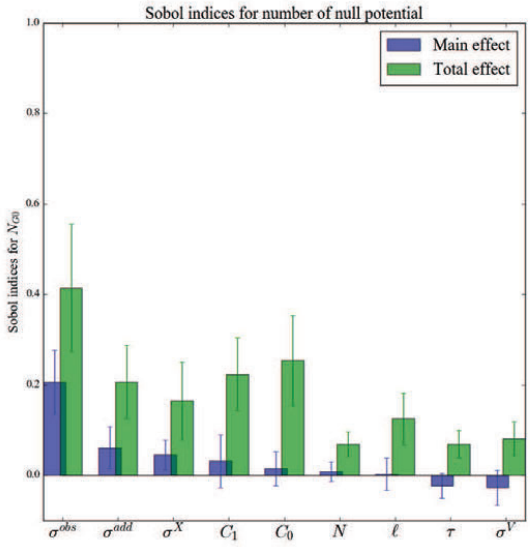


Figure 14 – Indices de Sobol d'ordre 1 : simple (blue) et total (green) pour le nombre de potentiels nuls.

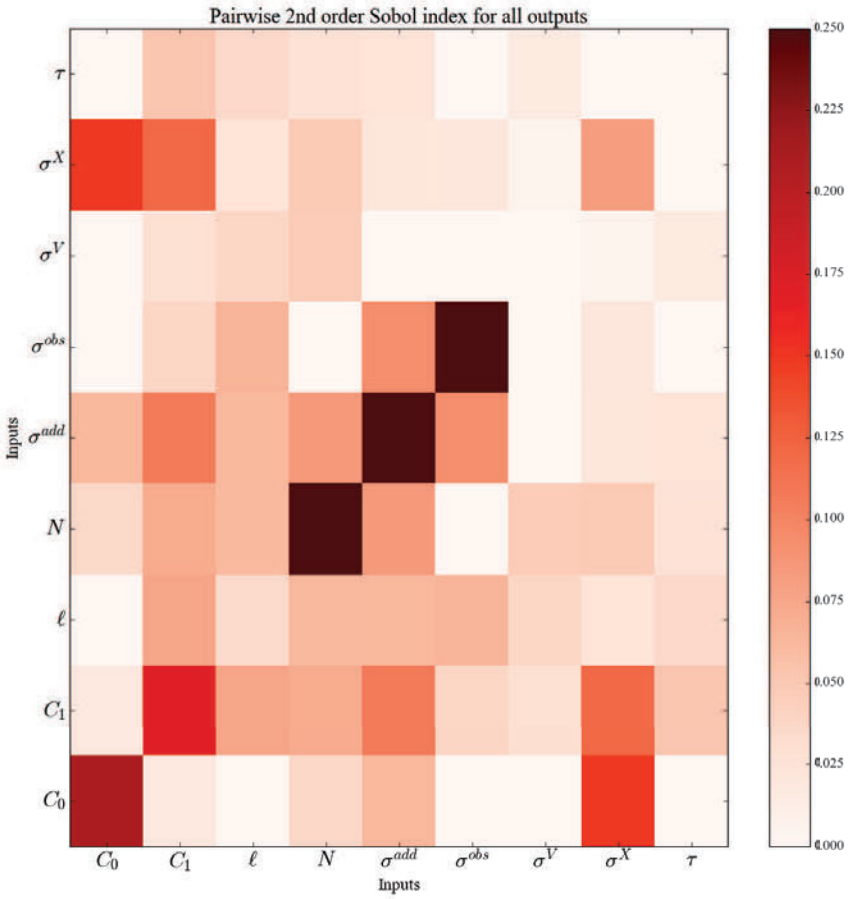


Figure 15 – Somme des indices de Sobol simple d'ordre 2 (hors diagonale) et d'ordre 1 (diagonale) de toutes les entrées.

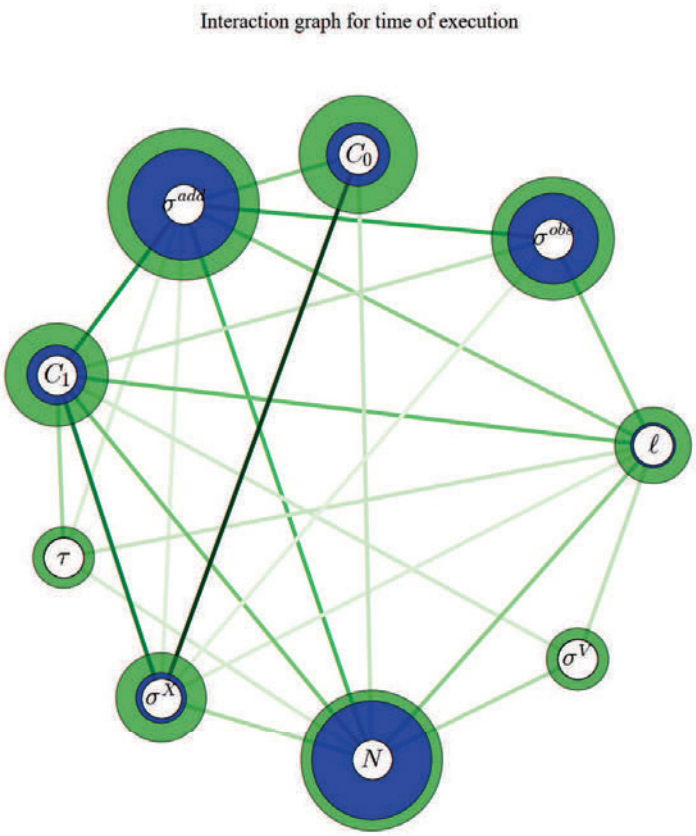


Figure 16 – Graphe d'interaction de la "sortie globale". Couronne intérieure : indice simple d'ordre 1, couronne extérieure : indice total d'ordre 1, opacité des arrêtes : indice simple d'ordre 2.

Chapitre 7 : Exploration avec les expériences 2-à-2

Le chapitre 7 présente les résultats d'expériences complémentaires appelées expériences 2-à-2. Elles consistent à ne faire varier que deux paramètres d'entrée, laissant les autres à une valeur nominale. L'avantage de telles méthodes est de visualiser les variations des sorties en fonction des paramètres. Si les indices de Sobol apportent bien une information sur la force de l'influence, les expériences 2-à-2 permettent de voir dans quel sens elle s'exerce. Ces expériences ont été utilisées pour établir la stratégie de réglage sur le système réduit qui ressort de l'interprétation des indices de Sobol.

Les variations de la pente du spectre du vent en fonction des bruits d'observation σ^{add} et σ^{obs} (réel et donné au filtre, respectivement) montre que la pente du spectre est égale à la valeur théorique de -5/3 quand $\sigma^{add} = \sigma^{obs}$. Ce résultat dit que le spectre du vent est correct lorsque le bruit qu'on donne au filtre est égal au bruit réel, ce qui n'a rien de choquant. Lorsqu'on regarde l'erreur sur le vent en fonction des bruits d'observation σ^{add} et σ^{obs} on constate que pour un σ^{add} donné (en pratique ce paramètre est fixe mais inconnu), elle est minimale lorsque $\sigma^{obs} = \sigma^{add}$. Comme les variations de la pente du spectre sont maximale autour de la ligne $\sigma^{obs} = \sigma^{add}$ et qu'on dispose d'une valeur cible, il est judicieux d'utiliser cette sortie pour régler le paramètre σ^{obs} de façon à ce que $b = -5/3$. Alors, on a expérimentalement réglé σ^{obs} à la même valeur que σ^{add}.

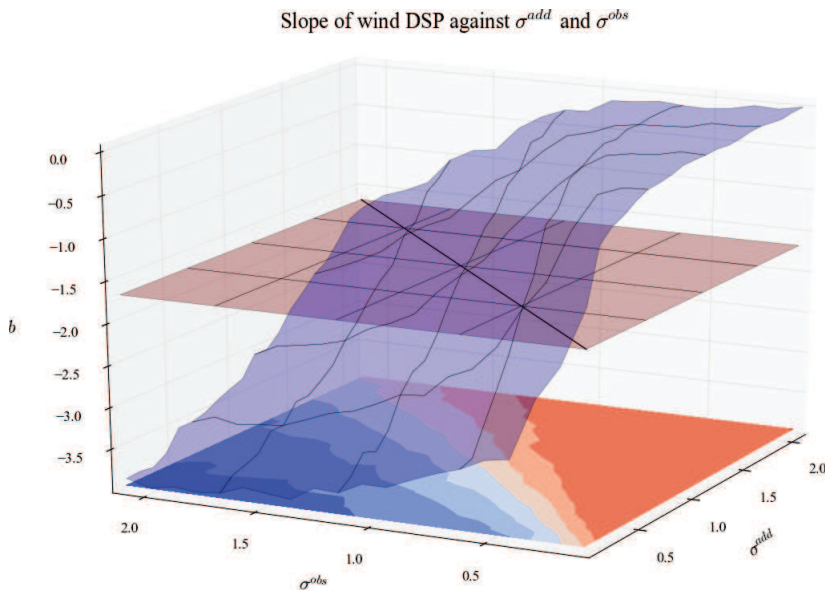


Figure 17 – Évolution de la pente du spectre, b, quand seulement σ^{add} et σ^{obs} varient. Le plan rouge est au niveau $b = -5/3$ (valeur théorique attendue).

Une fois que σ^{obs} et σ^{add} sont confondus, on note toujours leur valeur σ^{obs}. L'évolution de l'erreur sur le vent en fonction du nombre de particules montre une décroissance en $\sqrt{N}$, comme prévu par la théorie. L'erreur est aussi proportionnelle au bruit d'observation. Elle

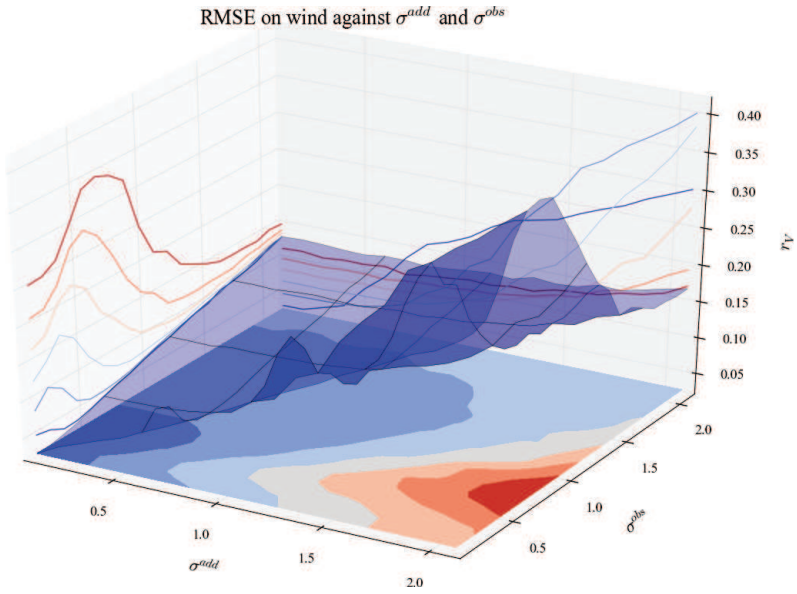


Figure 18 – Évolution de l'erreur sur le vent, r_V, quand seulement σ^{add} et σ^{obs} varient.

peut donc être estimée par la formule suivante

$$r_V = K\frac{\sigma^{obs}}{\sqrt{N}}$$

où est une constante estimée à $K = 2.33$.

Par ailleurs, on montre que le temps exécution croit proportionnellement à $N^{1.75}$. Si l'on double la quantité de particules, le temps de calcul sera multiplié par 3.36. Ce chapitre ce conclut donc par l'établissement d'une stratégie de réglage très simple :

1. Mettre N à une valeur faible, de façon à ce que T_{exe} soit très petit.
2. Pour σ^{obs} prenant plusieurs valeurs encadrant le bruit réel, calculer la pente du spectre du vent, b.
3. Mettre σ^{obs} à la valeur qui donne b le plus proche de -5/3. σ^{obs} est expérimentalement égal à σ^{add}.
4. Augmenter N autant que possible. L'erreur sur le vent est alors minimum, estimée par $K\frac{\sigma^{obs}}{\sqrt{N}}$ avec $K = 2.33$.

D'autres résultats se retrouvent dans les expériences 2-à-2, comme la variation du nombre de potentiel nuls, N_{G0} en fonction de N et aussi de σ^{obs} et σ^{add}. L'influence de C_0 et C_1 est aussi décortiquée.

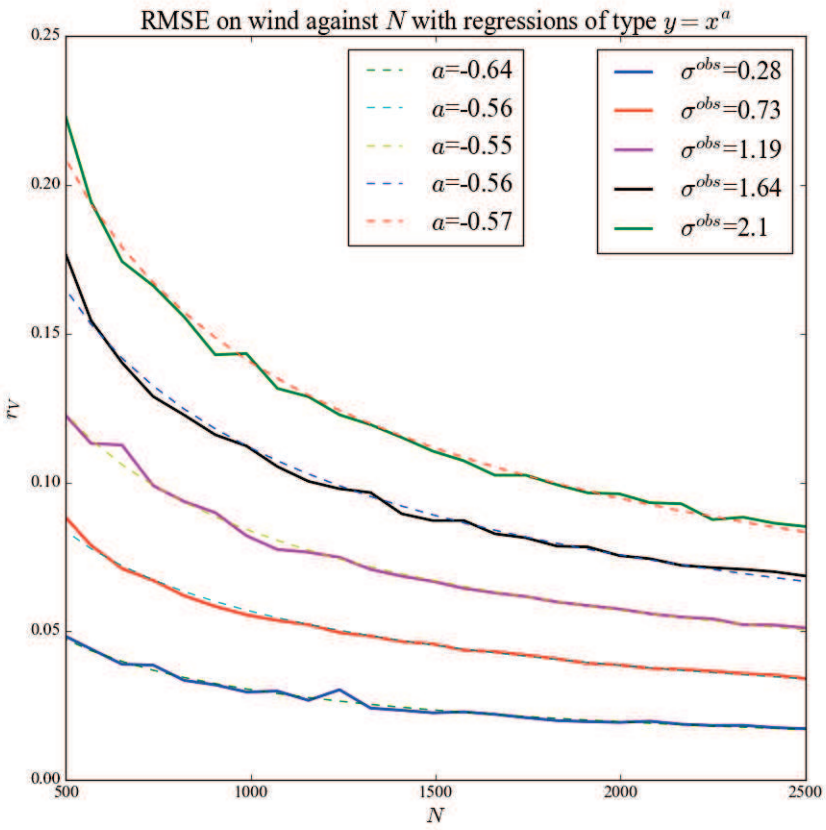


Figure 19 – Évolution de l'erreur sur le vent en fonction de N quand $\sigma^{obs} = \sigma^{add}$, pour différente valeurs de σ^{obs}. Les régressions (lignes pointillées) montrent que la décroissance se fait en racine carrée.

Chapitre 8 : Régressions pénalisées pour l'estimation des indices de Sobol

Ce huitième et dernier chapitre fait l'expérience d'estimer les indices de Sobol par des régressions pénalisées. Pour un code avec p paramètres d'entrée, il y a 2^p groupes de paramètres, donc autant d'indices de Sobol à estimer. Le nombre d'indices de Sobol croit exponentiellement avec le nombre d'entrées. En pratique, seulement quelques uns sont intéressants. Le problème n'est donc pas tant d'avoir une estimation précise, mais plutôt une estimation qui fasse le tri.

Les régressions pénalisées fournissent des estimateurs biaisés, mais dont les plus petits coefficients vont aller d'eux même à zero. Cela facilite l'interprétation finale et enlève la part de subjectivité dans le choix des indices interprétés. Leur construction part du constat que les indices de Sobol peuvent être exprimés comme la solution d'une régression linéaire. Lorsqu'on écrit l'estimateur correspondant aux moindres carrés ordinaires, on retombe sur l'estimateur de Sobol (2001). Ce sera le point de comparaison pour les nouveaux estimateurs.

Les nouveaux estimateurs sont obtenus en rajoutant une pénalité sur les coefficients dans le problème de minimisation des moindres carrés. En rajoutant une pénalité L^1, on obtient l'estimateur Lasso. En rajoutant une pénalité L^0, on obtient l'estimateur de meilleur sous-ensemble. Les estimateurs sont notés de la façon suivante :

$$\widehat{\mathbf{S}}^{ls} = \arg\min_{\mathbf{a}} \left\{ \sum_{\mathbf{u}\in I'} \|Y - a_{\mathbf{u}} Y_{\mathbf{u}}\|_2^2 + \|Y_{\mathbf{u}} - a_{\mathbf{u}} Y\|_2^2 \right\}$$

$$\widehat{\mathbf{S}}^{l1} = \arg\min_{\mathbf{a}} \left\{ \sum_{\mathbf{u}\in I'} \|Y - a_{\mathbf{u}} Y_{\mathbf{u}}\|_2^2 + \|Y_{\mathbf{u}} - a_{\mathbf{u}} Y\|_2^2 + \lambda\|\mathbf{a}\|_1 \right\}$$

$$\widehat{\mathbf{S}}^{l0} = \arg\min_{\mathbf{a}} \left\{ \sum_{\mathbf{u}\in I'} \|Y - a_{\mathbf{u}} Y_{\mathbf{u}}\|_2^2 + \|Y_{\mathbf{u}} - a_{\mathbf{u}} Y\|_2^2 + \lambda\|\mathbf{a}\|_0 \right\}$$

avec $Y = f(X)$, $Y_{\mathbf{u}} = f(Z_{\mathbf{u}}, X_{\bar{\mathbf{u}}})$, $\|\mathbf{a}\|_1 = \sum_{\mathbf{u}} |a_{\mathbf{u}}|$ et $\|\mathbf{a}\|_0 = \sum_{\mathbf{u}} \mathbf{1}_{a_{\mathbf{u}}\neq 0}$.

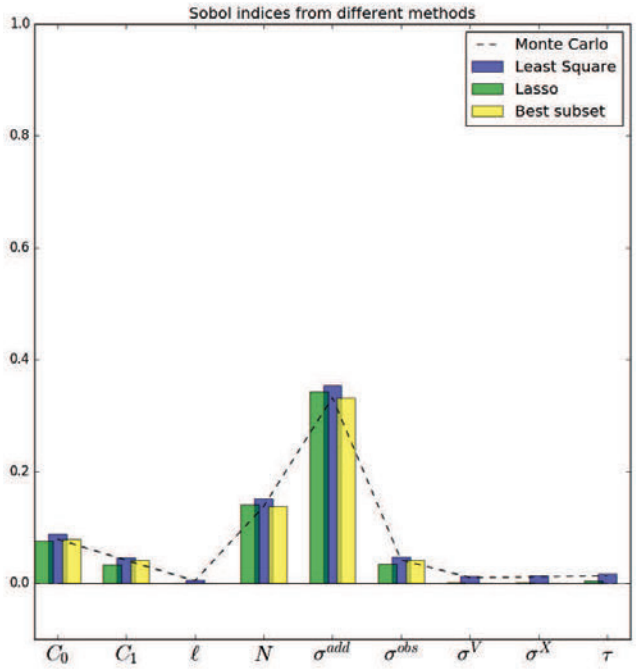


Figure 20 – Trois estimateurs sont issus de régressions (barres de couleur) sontcomparé à l'estimateur de Monte Carlo (ligne noire poitnillée). On peut voir que l'estimateur aux moindres carrés ordinaire (blue) conne la même estimation que Monte Carlo (plus ou moins un alea dû à l'estimation). Le meilleur sous-ensemble est exactement égal au Monte Carlo, mis à part les indices plus petit qu'un seuil, qui sont à 0. L'estimateur Lasso donne une estimation toujours plus faible dont la réduction s'arrête à 0.

L'estimateur Lasso sous-estime systématiquement l'indice de Sobol. On peut l'exprimer

en fonction de l'estimateur aux moindres carrés grâce un seuillage doux:

$$\widehat{S}_{\mathbf{u}}^{l1} = \max\left(\widehat{S}_{\mathbf{u}}^{ls} - \varepsilon_1, 0\right)$$

L'estimateur de meilleur sous-ensemble a l'avantage d'être moins biaisé. On peut l'exprimer en fonction de l'estimateur aux moindres carrés grâce un seuillage dur:

$$\widehat{S}_{\mathbf{u}}^{l0} = \widehat{S}_{\mathbf{u}}^{ls} \mathbf{1}_{\widehat{S}_{\mathbf{u}}^{ls} > \varepsilon_0}$$

Malheureusement, la résolution du problème de minimisation des moindres carrés avec pénalité L^0 est NP-difficile. Il n'existe pas d'algorithme qui converge en temps fini. L'estimation donnée a été obtenue par seuillage dur de l'estimateur de Monte Carlo. L'estimateur de meilleur sous-ensemble ne fait donc pas l'économie d'estimer les indices de Sobol par Monte Carlo, alors que l'estimateur Lasso est accessible par d'autres algorithmes de minimisation.

En l'état, les régressions pénalisées ne ferait que déplacer la subjectivité de la sélection des indices de Sobol vers le choix de la pénalité, λ. Pour choisir objectivement λ, nous avons choisi de calculer l'erreur quadratique moyenne pour plusieurs valeurs de λ et de choisir celle qui donne l'erreur la plus faible. Ces méthodes ont été testées favorablement sur l'exemple montré (sortie r_V, estimation des indices simple d'ordre 1).

Conclusion générale

Ce travail fait l'analyse de sensibilité d'une méthode de filtrage pour les mesures de vent. La méthode de filtrage y est décrite très précisément et un effort de rationalisation a été nécessaire pour définir les système sur lequel porte l'analyse de sensibilité. L'analyse de sensibilité consiste à estimer les indices de Sobol d'ordre 1 (simple et totaux) et 2 (simple) à l'aide d'un méta-modèle obtenu par krigeage gaussien. Le système retenu comporte 9 entrées et 5 sorties. L'analyse de sensibilité permet de le réduire à 3 entrées et 3 sorties. Des expériences complémentaires (expériences 2-à-2) permettent d'explorer la façon dont les paramètres principaux influencent les sorties. Il en résulte une stratégie très simple de réglage des paramètres les plus importants. Par ailleurs, une nouvelle méthode d'estimation-sélection des indices de Sobol par régression pénalisées a été testée. Les indices de Sobol peuvent s'écrire comme la solution d'un problème de régression. L'ajout d'une pénalité permet de sélectionner automatiquement les indices les plus pertinents. Un exemple d'estimation-sélection autonome est donné pour deux types de pénalités (L^1 et L^0).

Annexe A : Rappels théoriques

La première annexe a pour but d'unifier les bases théoriques communes à la plupart des lecteurs. Il veut accessible à de futurs étudiants donc reprend assez en amont la théorie des probabilités. Tribu, mesures, probabilité sont définies rapidement avec quelques exemples

et contre-exemples. La notion de variable aléatoire est étendue aux "fonctions aléatoires", c'est-à-dire les processus stochastiques. Deux exemples de processus stochastiques sont donnés à titre d'illustration : le mouvement brownien et l'équation de Langevin. Ils permettent d'aborder des questions récurrentes en traitement du signal, telles que la stationnarité et l'ergodicité. En effet, la mesure la couche limite atmosphérique n'y fait pas exception, le "signal" que l'on cherche est aléatoire (c'est un processus stochastique) alors que la mesure ne donne qu'un "signal" temporel (une réalisation du processus stochastique). Stationnarité et ergodicité sont les deux hypothèses nécessaires pour pouvoir déduire d'une seule réalisation des quantité représentative du processus stochastique (moyenne, auto-corrélation...). Le mouvement brownien et l'équation de Langevin les illustre bien car le premier n'est ni stationnaire ni ergodique, alors que le second est stationnaire au sens faible et ergodique. Les résultats sur l'équation de Langevin re-servent lors de l'établissement du modèle Lagrangien de turbulence au chapitre suivant.

La génération de nombres aléatoires y est aussi abordée. En effet, un générateur de mauvaise qualité détériore le filtrage et peut être difficile à détecter. Les générateurs aléatoires des deux langages de programmation utilisés (Scilab et Fortran 90) ont été testés. Tous deux sont de bonne qualité, avec un net avantage de rapidité pour le générateur Fortran. L'algorithme d'inversion de la fonction de répartition est réutilisé dans le chapitre 6 pour expliquer le rééchantillonage des particules.

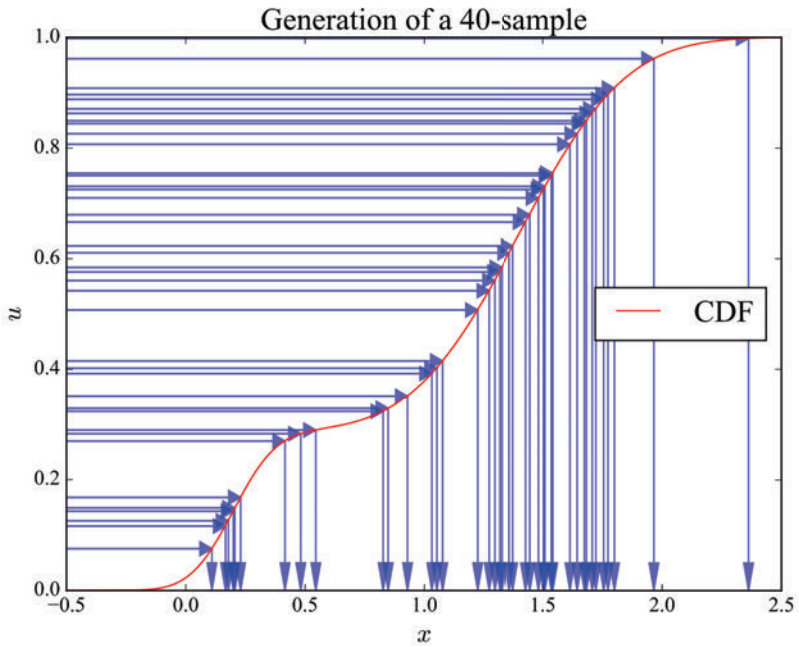


Figure 21 – Exemple de rééchantillonage par inversion de la fonction de répartition.

Annexe B : Preuves

L'annexe B contient les démonstrations de certains résultats théoriques utilisés dans le corps du manuscrit. Elles sont organisées en trois sous-sections :

B1. Formules sur les sorties de la reconstruction

B2. Théorèmes d'analyse de sensibilité

B3. Résultats relatifs à la régression pénalisée

Certaines preuves sont des réécritures de preuves existantes avec les notations de ce manuscrit. La référence vers la preuve initiale est alors donnée en tout début de preuve. Initialement écrites pour de l'auto-formation, ces preuves sont soit un peu plus détaillées soit présentées différemment des preuves initiales. Les preuves où aucune référence n'est donnée au tout début sont un travail personnel.

Annexe C : Résultats complets des expériences 2-à-2

L'annexe C contient les résultats complets des expériences 2-à-2 présentées au chapitre 7. Les résultats majeurs sont synthétisés dans le chapitre 7. L'annexe rappelle le cadre des expériences (liste des entrées, des sorties, schéma du système) et donne un tableau d'hyperliens pour faciliter la navigation. Les résultats sont listés par expériences (14 au total) puis par sorties (5 pour chaque expérience). Elle permet de vérifier que l'effort de synthèse du chapitre 7 ne passe pas à côté de commentaire important et donne la possibilité au lecteur de les faire par ailleurs.

General introduction

– What do you do for living?
– I'm doing a PhD on the sensitivity analysis of a filtering method for wind measurements.
– Cool, I suppose! But, what is that for?

Usual beginning of conversation with new people during these last three years.

The path to connect research to the "real world" can be long and tricky. The job of an introduction is to make it short and simple. To do short, I start with the title and try to make it simple:

Sensitivity analysis of a filtering algorithm for wind lidar measurements

Wind is the movement of the air. We will abord only the wind in the boundary layer (first kilometer of atmosphere). In this area of atmosphere, there is turbulence (rapid variations of wind around its average value) and humans. It makes the boundary layer very important because of the direct impact on human activities. However, meteorological models hardly catch turbulence and boundary layer processes. To improve them, scientists need measurements.

Lidar instruments are well suited to measure wind in the boundary layer. They are remote sensors with good time and space resolution (for example, the lidar used in this thesis has 50m of vertical resolution and 4s of time resolution). To improve the knowledge of wind and turbulence we get from them, measurements are filtered.

Filtering consists in removing the noise of a signal. This thesis is about a particular filtering method that does more than denoise the wind measurment. It also provides turbulence estimation in real time, which was previously unseen. But this method is new and the estimation must be qualified. In particular, the influence of the settings has never been assessed.

Sensitivity analysis quantifies the influence of input parameters on a given output. Sensitivity indices (Sobol indices) are estimated for the filtering method. The settings of the filtering method will be our input parameters and some scores will be defined to be outputs. Knowning the influence of parameters allow to build tuning strategies, which help to improve the quality and the self-reliance of the wind measurements and to adapt to new conditions.

How is it useful?

Improving the measurements of wind and turbulence is useful for many sectors of activity, meteorology on top. The need of measurements to build know better the processes at stake and build relevant meteorological models is obvious. The improvement of meteorological models has then positive fallout on our daily life. We can quote at least two sectors where the benefit of good wind and turbulence measurements is direct: wind energy and aeronautics.

Wind energy turn trendier each year thanks to the increasing concern of people to environmental questions (as shown by the figure 22). Indeed, energy is the dominant contributor to climate change, accounting for around 60 per cent of total global greenhouse gas emissions. This is also a social challenge, given that one in five people still lacks access to modern electricity. *Affordable and sustainable energy* is the 7th Sustainable Development Goal targeted by the United Nations. Solar or wind energy are well suited to answer a local, decentralized, autonomous demand. No greenhouse gas is emitted during the production of power. The energy source is free and of unlimited supply. Industrial risk is reduced to nearly zero in comparison with nuclear power plant and dams. Nevertheless, their costs are still higher than those of fossil-based and nuclear energy. Indeed, fossil-based and nuclear energy are well known technologies, relying on a well trained net of companies and customs. For wind energy, such net is only emerging and technologies are not as mature. Moreover, efficient solutions to deal with the intermittent nature of the energy source are still to be found. Smart grids can alleviate the need of storage (not possible at large scale) but algorithms need information about the energy source.

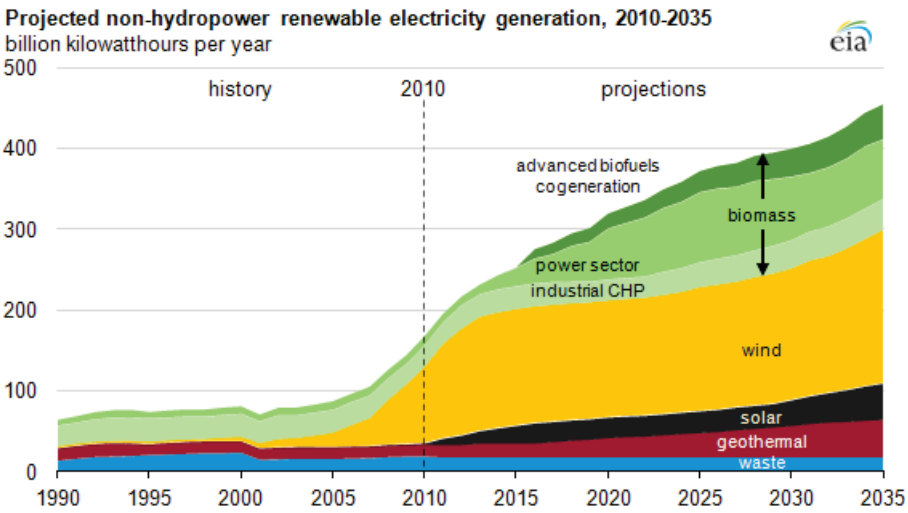


Figure 22 – Projected evolution of renewable energy mix in the the US.
Source: US Energy Information Administration, Annual Energy Outlook 2012.

For wind energy, the need of wind measurements increases because it helps assessing the energy source. Wind turbines are about 100m high (and their height tends to increase). Surface stations are not sufficient to provide such information. Remote sensors, such as

lidars, are voted in by many wind energy companies as a solution to measure the wind around turbines. The benefits of having accurate measurements of wind aloft are three fold for wind energy:

- Measure the wind ahead of a turbine in order to make short term prediction of production.
- Measure the wind behind a turbine, to characterize wakes created by the turbine.
- Measure the wind on a prospective site, to assess the opportunity of building a wind farm.

The two last benefits can be combined to optimize the wind farm layout[1] (illustrated in figure 23). Beside the wind, the turbulence is also a valuable information to manage wind farm. Turbulence reduces the aerodynamic performance of the blades and apply random variations of forces which damage the turbine faster. Having information about turbulence allow to include maintenance plan in the management of the farm.

Figure 23 – Offshore wind farm. The turbulence created by the first layer of turbines is visible thanks to the condensation.
Picture: Vattenfall Wind Power, Horns Rev wind farm, Denmark

[1]Indeed, wind turbines of the same farm are oriented toward the dominant wind. The first layer of turbines profits of the full force of wind, but then the wind left behind is lower (about 5%). Moreover, it creates a turbulent wake which damage the mechanical components in the long term (Maalouf, 2010). Wind farm layout optimization consists in distributing the turbines on the site such that the total power provided is optimized (taking into account these umbrella effects).

Aeronautics is marked by a tremendous growth of traffic. The figure 24 shows how fast the worldwide traffic increased between 1944 and 2016. It is expressed in revenue passenger-kilometer. This measure of traffic is the product of the number of users (passengers paying for the flight, crew members are not included) with the kilometers travelled. One can see the steady increase, despite several crises affecting it. On the other hand, air transport safety must be guaranteed. To deal with the increasing traffic, airport management must gain in efficiency. Between two uses of the same runway, a minimum time is imposed because of the turbulence created by the aircrafts. One of the limiting factors for the succession of landings or take-off is thus the turbulence. The time imposed are standard thresholds depending on the weight of the preceding aircraft. Measurement-driven management of runway should safely reduce the time of unused runways.

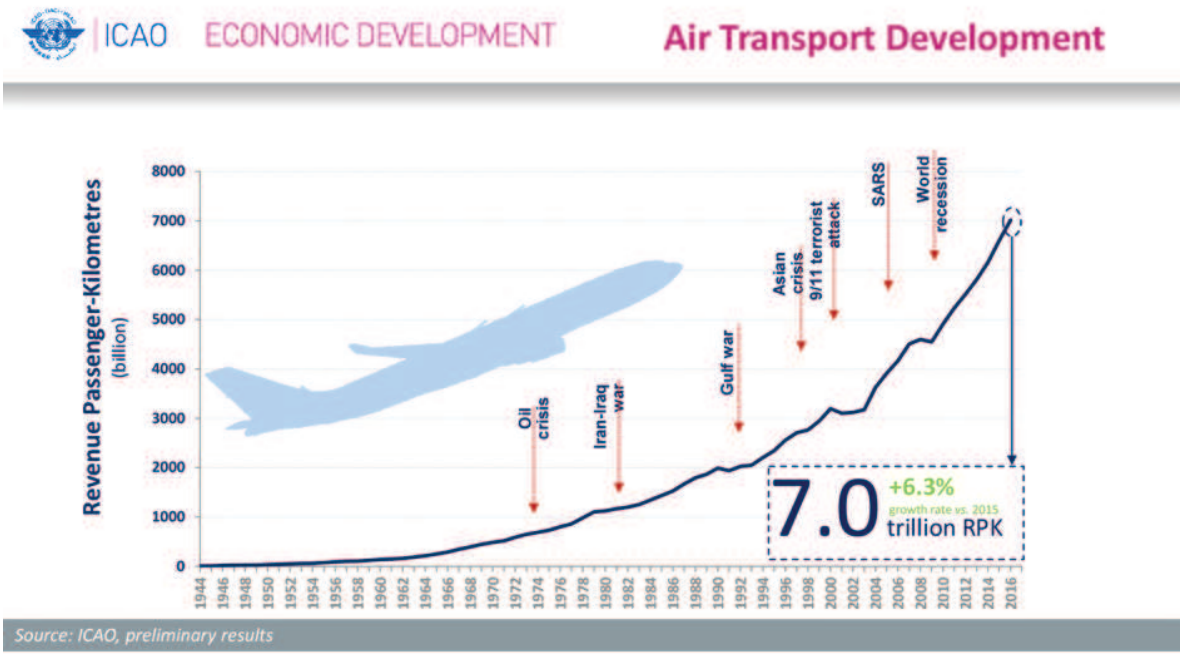


Figure 24 – Evolution of the world passenger air traffic from 1950 to 2016. The traffic is expressed in revenue passenger-kilometer.
Source: International Civil Aviation Organiation (ICAO), preliminary results of the 2016 study.

For aeronautics, the need of wind measurements increases because it warns about turbulences threats. Critical phases are take-off and landing. Hence, in addition to ground measurements, the first hundreds of meters must be measured. On airports, lidars are less cumbersome than met mast (although eye-safety is limiting the laser use). With an appropriate scanning geometry, wind can be measured in 3D. Thanks to that, they can detect in real time

- Wind direction in the runway and above.
- Turbulence in the wake of aircrafts.

Aircrafts have to move against the wind for both take-off and landing. Some airports are subject to reversals of the wind direction in the few ten meters of altitude (vertical wind shear), or to reversal of the wind within the distance of the runway (horizontal wind shear, as in figure 25). In such cases, the pilot must go around and attempt to land later, or elsewhere. The wind direction is thus an important factor to manage the airport. Mapping the wind as a lidar can do is thus a valuable tool for airport management.

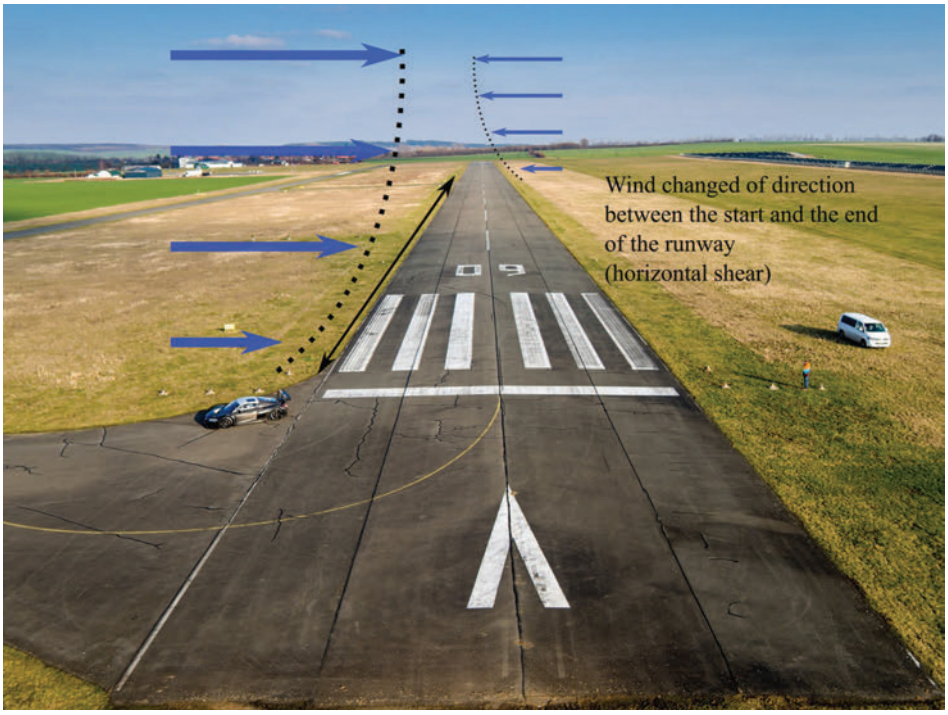


Figure 25 – Illustration of horizontal wind shear. Landing and take-off are difficult in such conditions and can be postponed.

Organisation of the manuscript

The filtering method (so-called *reconstruction*) is interdisciplinary, and the sensitivity analysis arrives on top of it. To clarify it the best I can, the manuscript is separated in height chapters and three appendices:

1. Turbulence in the atmospheric boundary layer
2. Bayesian filtering
3. Instrument and material
4. Sensitivity analysis theory
5. Reconstruction of turbulent medium
6. Results of the sensitivity analysis
7. Exploration with 2-by-2 experiments
8. Penalised regressions to estimate Sobol indices

A. Theoretical background
B. Proofs
C. Complete results of the 2-by-2 experiments

The four first depict a goal-oriented state of the art in the different areas of research involved in the reconstruction. Chapter 1 presents turbulence in a broad way and then focus on questions raised by the Lagrangian representation in the reconstruction. Chapter 2 presents the filtering technique (genetic selection algorithm) through an introduction to Bayesian filtering. Chapter 3 focuses on the lidar technology (principle, strength, limitations) and the data used in this thesis. Chapter 4 explains how sensitivity analysis is carried out (Sobol indices definition and estimation, Gaussian meta-model built).

The four last chapters present applications of the previous ones and the results. Chapter 5 explains the reconstruction method in details, set the testing framework and gives original results about the reconstruction. Chapter 6 presents and comments the results of the sensitivity analysis. Chapter 7 sustains the highlighted mechanisms with additional results (so-called *2-by-2 experiments*). Chapter 8 presents the testing of a new estimation way of Sobol indices, with penalized regression, which makes insignificant indices go themselves to exactly zero.

The three appendices complete the main matter. Appendix A set a common basis of mathematical tool and notations for all readers, including students. Appendix B gives the proofs of some results stated in the chapters 5, 6 and 8. Appendix C lists all the results obtained with 2-by-2 experiments.

Part I

Goal-oriented state-of-the-art

Chapter 1

Turbulence in the atmospheric boundary layer

Contents

1.1 Generalities in fluid mechanics

This section aims to give the essential basis for the comprehension of atmospheric turbulence, its context and its issues. Informations stated here variously come from reference books such as Holton and Hakim (2012); Stull (1988) and Malardel (2005); de Moor (2006); Chassaing (2000) (in French).

1.1.1 Eulerian and Lagrangian description

A fluid can be described from 2 different referentials :

- From a Galilean referential : it is the *Eulerian description*. The fluid is described by a fixed observer. Time and space are its coordinates. It is the most common point of view because it has the advantage to handle easily boundary conditions and to be easier to compare with observations.

- From a local referential : it is the *Lagrangian description*. The fluid is described by an observer in the fluid, moving with the fluid (passively). Coordinates are then the time and the initial position of the observer. It has the advantage to consider closed systems (fluid particles), subject to external forces with no intern variations.

Dynamic and thermodynamic principles only hold for closed system. They are applied to fluid particles : infinitesimal volumes of fluid.

Definition 1.1 (Fluid particle)**.** *A fluid particle is an infinitesimal volume of fluid containing enough particles to correctly define statistical quantities (temperature, pressure...) but small enough to consider that quantities as constant within the volume.*

The size of a fluid particle has to be large compared with the mean free path of air molecules. At standard pressure (1013 hPa) and temperature (20°C), there are roughly $2.7{\cdot}10^{19}$ molecules per cm^3 and their mean free path is approximately 68 nm (according to Jennings (1988), table 2 and 5). A fluid particle of 1mm will contain approximately 10^{16} molecules and be 5 order of magnitude larger than the mean free path. In a $20m^3$ volume of atmosphere[1], there are thus approximately $2{\cdot}10^{10}$ fluid particles. Considering these order of magnitude, it is reasonable to assume that there are an infinity of fluid particles in the probe volume of lidar.

To describe fluid dynamics, we consider the closed system of a fluid particle that evolve in a part of space $\mathcal{D} \subset \mathbb{R}^3$. This part of space can be divided into a set of fluid particles. We denote $X(i,t)$ the position of the particle i at time t, and $u(i,t)$ its speed. Let Ψ be a scalar quantity of the fluid (for example Ψ can be the temperature, or the vertical velocity).

When given for one location and one time, it is an Eulerian variable:

$$\Psi^E : \begin{array}{ccc} \mathcal{D} \times \mathbb{R}_+ & \longrightarrow & \mathbb{R} \\ (x,t) & \longmapsto & \Psi^E(x,t) \end{array}$$

[1]The typical volume of representativity for a lidar measurement. In figure 1.6 gates are 50m large and the measure is considered as valid in a section of 1m of diameter

When given for one fluid particle and one time, it is a Lagrangian variable :

$$\Psi^L : \begin{array}{ccc} \mathcal{I} \times \mathbb{R}_+ & \longrightarrow & \mathbb{R} \\ (i,t) & \longmapsto & \Psi^L(i,t) \end{array}$$

where $\mathcal{I}$ is the set of particles index. The link between Eulerian and Lagrangian point of view is given by the relation:

$$\Psi^L(i,t) = \Psi^E(X(i,t),t) \tag{1.1}$$

Hence, the derivative in time for Lagrangian quantities (which is linked to external forces through the 2^{nd} law of Newton) can be expressed in Eulerian terms in this way[2]:

$$\begin{aligned} \frac{\partial \Psi^L(i,t)}{\partial t} &= \frac{\partial \Psi^E}{\partial t} + \frac{\partial X_\alpha(i,t)}{\partial t}\frac{\partial \Psi^E}{\partial x_\alpha}(X(i,t),t) \\ &= \frac{\partial \Psi^E}{\partial t} + u^E_\alpha(X(i,t),t)\frac{\partial \Psi^E}{\partial x_\alpha}(X(i,t),t) \end{aligned}$$

1.1.2 Navier-Stokes equations and Boussinesq approximation

Thermodynamic principles that are at the base of fluid dynamics only hold for closed system. In fluids, the closed system is usually a fluid particle. Several thermodynamic principles apply to a fluid particle :

- Mass conservation
- Momentum conservation (2^{nd} law of Newton)
- Thermodynamic equilibrium
- Ideal gaz law

These equations involve six unknowns: the temperature T, the pressure P, the density of air ρ, the eastward velocity u, the northward velocity v, the vertical velocity w. Only dry air is considered. To ease the writing of equation, the three components of wind are numbered: $(u,v,w) = (u_1,u_2,u_3)$. Using the link between Lagrangian variables (for which thermodynamic principles hold) and Eulerian variables, we can write 6 equations for the 6 variables (T,P,ρ,u,v,w) that are the Navier-Stokes equations (from de Moor (2006), page 56) :

$$\frac{\partial \rho}{\partial t} = -\rho\frac{\partial u_\alpha}{\partial x_\alpha} \tag{1.2}$$

$$\forall i \in \{1,2,3\},\ \frac{\partial u_i}{\partial t} + u_\alpha\frac{\partial u_i}{\partial x_\alpha} = -\frac{1}{\rho}\frac{\partial P}{\partial x_i} + \nu\frac{\partial^2 u_i}{\partial x_\alpha^2} + \frac{\nu}{3}\frac{\partial}{\partial x_i}\left(\frac{\partial u_\alpha}{\partial x_\alpha}\right) + g\delta_{3,i} - 2\epsilon_{i,\alpha,\beta}\Omega_\alpha u_\beta \tag{1.3}$$

[2]Einstein's summation convention is used for Greek letter indices: $x_\alpha := \sum_\alpha x_\alpha$.

$$\frac{\partial C_v T}{\partial t} = -\frac{P}{\rho}\frac{\partial u_\alpha}{\partial x_\alpha} + \nu_\theta \frac{\partial^2 T}{\partial x_\alpha^2} \tag{1.4}$$

$$P = \rho R T \tag{1.5}$$

with $\delta_{i,j} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{else} \end{cases}$ and $\epsilon_{i,j,k} = \begin{cases} 0 & \text{if } i = j \text{ or } i = j \text{ or } j = k \\ -1 & \text{if } j > k \\ 1 & \text{if } j < k \end{cases}$.

The remaining variables are the kinematic viscosity of air ν, the gravitational acceleration g, the Earth rotation vector $(\Omega_1, \Omega_2, \Omega_3)$, the specific heat capacity C_v, the thermal diffusivity ν_θ and the specific air constant R.

Although Navier-Stokes equations provide a complete closed system of equations to describe fluids, it is not usable directly in practice. Assumptions are made to simplify the system for a special application.

To study convection and atmospheric boundary layer, it is usually assume that the state of atmosphere is "not too far" from a hydrostatic state at rest ($\Psi = \Psi_0 + \Psi_1$, for Ψ being T, P, ρ, u, v, w). The temperature variable T is modified into the potential temperature $\theta = T\left(\frac{P_0}{P}\right)^{2/7}$ (which has the advantage to be invariant by adiabatic compression). The use of potential temperature θ instead of the temperature T is so common that it is frequent to call θ "the temperature". This yields to the Boussinesq system (still from de Moor (2006), page 63):

$$\frac{\partial u_\alpha}{\partial x_\alpha} = 0 \tag{1.6}$$

$$\forall i \in \{1,2,3\},\ \frac{\partial u_i}{\partial t} + u_\alpha \frac{\partial u_i}{\partial x_\alpha} = -\frac{1}{\rho_0}\frac{\partial p_1}{\partial x_i} + \nu \frac{\partial^2 u_i}{\partial x_\alpha^2} + \beta(\theta - \theta_0)\delta_{3,i} - 2\epsilon_{i,\alpha,\beta}\Omega_\alpha u_\beta \tag{1.7}$$

$$\frac{\theta}{\partial t} + u_\alpha \frac{\partial \theta}{\partial x_\alpha} = \nu_\theta \frac{\partial^2 \theta}{\partial x_\alpha^2} \tag{1.8}$$

$$\frac{\rho_1}{\rho_0} = -\frac{\theta - \theta_0}{\theta_0} \tag{1.9}$$

This system is at the basis of atmospheric boundary layer description, because this layer ground-driven by definition and consequently, leading fluxes are vertical. But it is still a deterministic partial differential equation, while turbulence is known for its randomness.

1.2 Focus on turbulence

1.2.1 Reynolds system : equation of atmospheric turbulence

A fluid is called "turbulent" by opposition to another state : "laminar" fluid. Although it seems clear for human eye what is turbulent and what is not, the criterion to distinguish turbulent fluid from laminar fluid is quite fuzzy. It is based on the Reynolds number.

Definition 1.2 (Reynolds number)**.**

With words *The Reynolds number is a dimensionless number, ratio in order of magnitude between the advective transport and the diffusive transport in a fluid.*

Formally *With the notations*

- L *: characteristic length scale of the flow*
- U *: characteristic speed of the flow*
- ν *: cinematic viscosity of the fluid*

$$Re = \frac{U \cdot L}{\nu}$$

Figure 1.1 depicts a von Kármán vortex street created by the Canaries islands in a turbulent flow. Such eddies are the signature of a punctual obstacle in a turbulent flow. On this example, the characteristic length scale and the characteristic speed of the flow are illustrated in the picture. The Reynolds number of the flow is deduced from them. A fluid with "large" Reynolds number is said turbulent, and a fluid with "low" Reynolds number is said laminar. But the thresholds for "large" and for "low" are different. For example[3], in a flow in a pipe of given diameter, the threshold for "low" Reynolds number is $Re < 2300$ but the threshold for "large" Reynolds number is $Re > 4000$. In between, the fluid has mixed characteristics. Moreover, "characteristic" parameters (length scale and speed) can be subject to discussion for some flows. For example, in wall bounded flows, one can define a local Reynolds number with boundary layer characteristics and a global Reynolds number with the freestream characteristics. That's why the contour of turbulence us quite fuzzy.

The particularity of turbulence is its rapid random variations that introduce a new formalism. Parameters are split in 2 components :

$$\Psi = \bar{\Psi} + \Psi' \tag{1.10}$$

where $\bar{\cdot}$ is an average operator satisfying the Reynolds axioms:

[3]From Wikipedia (Reynolds number page). It quotes Holman, "Heat transfer" (2002), but this reference is not accessible.

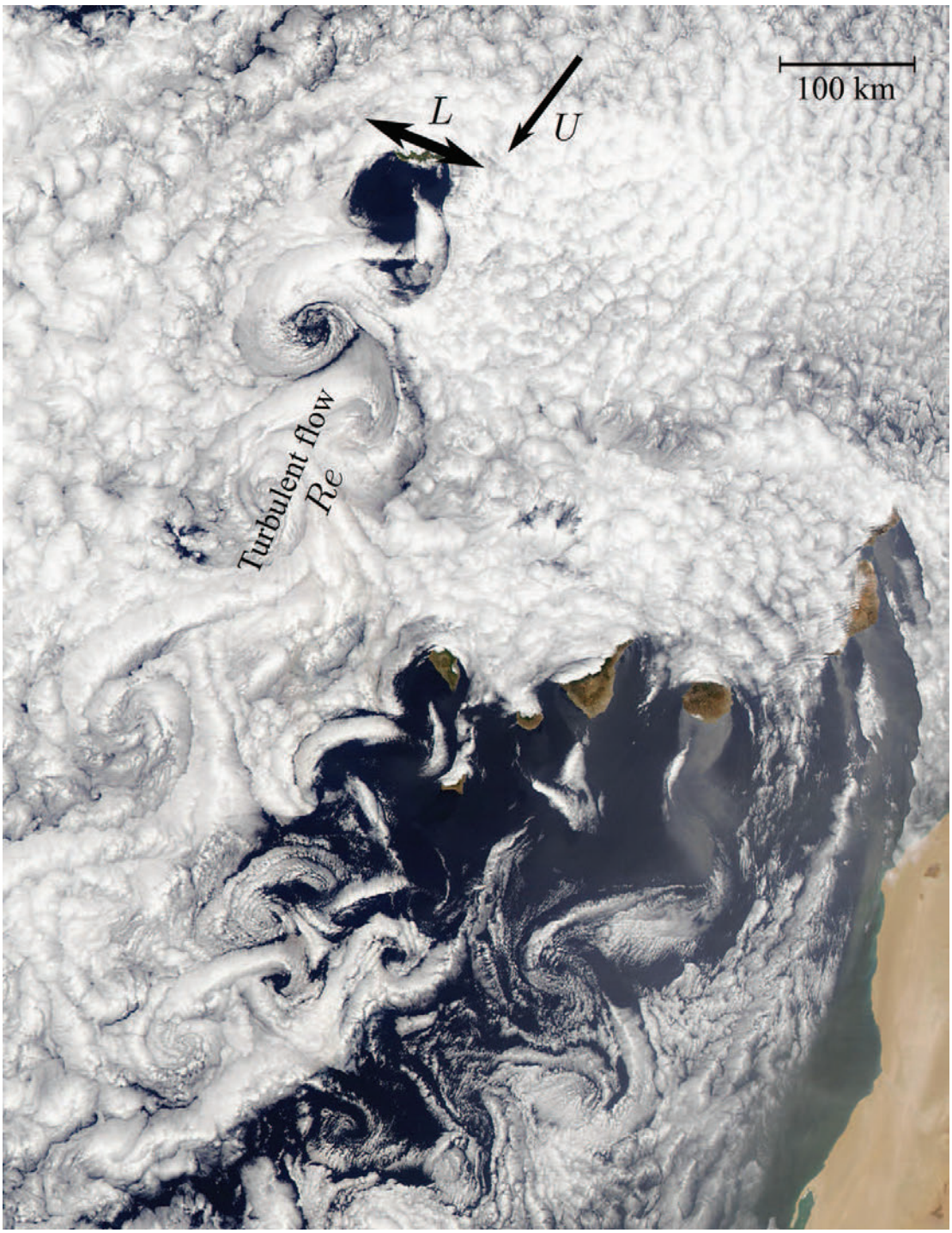


Figure 1.1 – MODIS image of the Canaries and Madeira islands creating a turbulent flow downstream. Characteristic length and speed involved in the Reynolds number estimation are highlighted (original image from NASA, public domain).

Linearity	:	$\overline{\Psi+\Phi}=\bar{\Psi}+\bar{\Phi}$
Commutativity with derivative	:	$\overline{\frac{\partial\Psi}{\partial x}}=\frac{\partial\bar{\Psi}}{\partial x}$
and integral	:	$\overline{\int\Psi}=\int\bar{\Psi}$
Idempotence	:	$\overline{\Psi\bar{\Phi}}=\bar{\Psi}\bar{\Phi}$

Table 1.1 – Reynolds axioms for the average operator.

In the atmospheric boundary layer, we neglect the variation of ρ since we are in a small portion of the densest part of the atmosphere. To relax this assumption, it is possible to consider $\overline{\rho\Psi}/\bar{\rho}$ (so-called Favre average) instead of $\bar{\Psi}$. We also give up the Coriolis force, because it is really small for kilometric scales. By applying the average operator to Boussinesq's system we get the Reynolds system (de Moor, 2006, p.125).

$$\frac{\partial\overline{u_\alpha}}{\partial x_\alpha}=0 \tag{1.11}$$

$$\forall i\in\{1,2,3\},\ \frac{\partial\overline{u_i}}{\partial t}+\frac{\partial\overline{u_\alpha u_i}}{\partial x_\alpha}+\frac{\partial\overline{u'_\alpha u'_i}}{\partial x_\alpha}=-\frac{1}{\rho_0}\frac{\overline{\partial p_1}}{\partial x_i}+\nu\frac{\partial^2\overline{u_i}}{\partial x_\alpha^2}+\beta(\overline{\theta}-\theta_0)\delta_{3,i} \tag{1.12}$$

$$\frac{\overline{\theta}}{\partial t}+\frac{\partial\overline{u_\alpha}\overline{\theta}}{\partial x_\alpha}+\frac{\partial\overline{u'_\alpha\theta'}}{\partial x_\alpha}=\nu_\theta\frac{\partial^2\overline{\theta}}{\partial x_\alpha^2} \tag{1.13}$$

In this system, one can see second order terms appear : $\overline{u'_\alpha u'_i}$ and $\overline{u'_\alpha\theta'}$. They make this system unclosed because they are new unknowns. Although it is possible to write new equations about them by crossing Boussinesq ($\partial\Psi/\partial t$) and Reynolds ($\partial\bar{\Psi}/\partial t$) systems to get the fluctuation equations ($\partial\Psi'/\partial t$), the non-linear terms will bring third order terms that let the system unclosed. It is done in de Moor (2006), p.129. This problem is known as the *turbulence closure problem* (Malardel (2005) p. 398, Stull (1988) p.197).

1.2.2 The turbulence closure problem

Many approaches are possible to bypass the closure problem. Most of them link empirically high order moments to lower order moments with a formula of various complexity. Depending on the order of the highest moments fully described, the closure methods are classified in *zero*, *first* or *second* order closure (Stull, 1988, p. 199). For example, first order closure assume a relationship of the type

$$\overline{u'_i\Psi'}=-K\frac{\partial\overline{\Psi}}{\partial x_i} \tag{1.14}$$

with K either a given constant or depending of others known parameters, and Ψ a scalar quantity.

A variation of these methods is to build empirical equation not for all high order moment

but only for ones that have a physical meaning. The advantage is double : reduce the number of equations and build a physically interpretable equation. An interesting parameter made from second order moments is the turbulent kinetic energy (TKE) :

Definition 1.3 (Turbulent kinetic energy)**.**

With words *Turbulent kinetic energy is half the sum of wind variance on each component. It represents the kinetic energy due to turbulent fluctuations.*

Formally

$$k = \frac{1}{2}\left(\overline{u'^2 + v'^2 + w'^2}\right)$$

The stronger the turbulence, the stronger the fluctuations and the higher the TKE. It thus a good tracer of turbulence, for which we can derive a prognostic equation from Reynold's system (see de Moor (2006) p. 275 and Stull (1988) p. 214, Chassaing (2000) p.158) :

$$\frac{\partial k}{\partial t} = -\overline{w'u'}\frac{\partial \bar{u}}{\partial z} - \overline{w'v'}\frac{\partial \bar{v}}{\partial z} - \frac{\partial}{\partial z}\left(\overline{w'k} + \overline{w'(p/\rho_0)}\right) + \beta\overline{w'\theta'} - \varepsilon \tag{1.15}$$

with ε the *eddy dissipation rate* (EDR). In this kind of closure, the factor K is supposed of the form

$$K = a \cdot l_m\sqrt{k} \tag{1.16}$$

with a a constant, l_m the *mixing length* and k the TKE. Other second-order moments are retrieved with the one-order closure, but the TKE (which is a second-order term) is explicitly calculated. It is thus called a 1.5-order closure.

Some new terms have been introduced in the 1.5-order closure but not discussed. They are the mixing length l_m and the eddy dissipation rate ε.

The mixing length l_m is a characteristic length of the flow. The difficulty is to make a rigorous interpretation of it, and thus its estimation is quite arbitrary. Nevertheless, it can be seen as the maximum length of excursion of a fluid particle moving vertically by consuming its turbulent kinetic energy. This interpretation is concrete and also yields to accessible calculations:

$$l_m = \sqrt{l_{up}l_{down}} \quad \text{with} \quad k(z) = \int_z^{z+l_{up}} \beta(\bar{\theta}(z) - \bar{\theta}(z'))dz' = \int_{z-l_{down}}^{z} \beta(\bar{\theta}(z') - \bar{\theta}(z))dz'$$

It is known as the Bougeault and Lacarrere (1989) mixing length.

The eddy dissipation rate ε is the rate of dissipation of turbulent kinetic energy in the absence of other source or sink (advection, friction...). It is also equal to the rate of energy transfer from large eddy to smaller eddies. The establishment of this equation from Reynolds' system yields to an expression of ε at high Reynolds number (de Moor (2006) p.138 and 280, Chassaing (2000) p.158) that we use as definition.

Definition 1.4 (Eddy dissipation rate)**.**

With words *Eddy dissipation rate is the rate of dissipation of turbulent kinetic energy in the absence of other source or sink (advection, friction...).*

Formally

$$\varepsilon = \nu \overline{\frac{\partial u'_\alpha}{\partial x_\beta} \frac{\partial u'_\alpha}{\partial x_\beta}} = \nu \overline{\left(\frac{\partial u'_1}{\partial x_1}\right)^2} + \nu \overline{\left(\frac{\partial u'_1}{\partial x_2}\right)^2} + \cdots + \nu \overline{\left(\frac{\partial u'_3}{\partial x_3}\right)^2}$$

It is an important term for energetics of turbulence. It is known, both theoretically and empirically that turbulence energy is spread continuously among scales (it is even a part of the definition of the *turbulence* phenomenon for Chassaing, (Chassaing (2000) p.9). In theoretical description of turbulence energetics (such as Kolmogorov's theory of 1941, (Kolmogorov, 1941)), EDR is often used as the rate of transfer from large eddy to small eddies, and to smallest eddies to dissipation in heat.

1.2.3 Kolmogorov spectrum

Turbulence is characterized by superimposition of eddies of different sizes. Large eddies are created by large scale forcings (generalized surface heating, gravity waves, mesoscale wind shear...) and they are splitting into smaller ones, until they reach a critical size where they are dissipated by viscosity. All the generations of eddies are superimposed in the fluid, as it is illustrated in figure 1.2. As a consequence, energy is spread continuously among all scales. Frisch et al. (1978) describe turbulence in terms of eddy generations. They denote l_0 the largest eddy in the fluid (l_0 is the boundary layer height) and the n-th generation of eddies is of size

$$l_n = l_0 2^{-n}$$

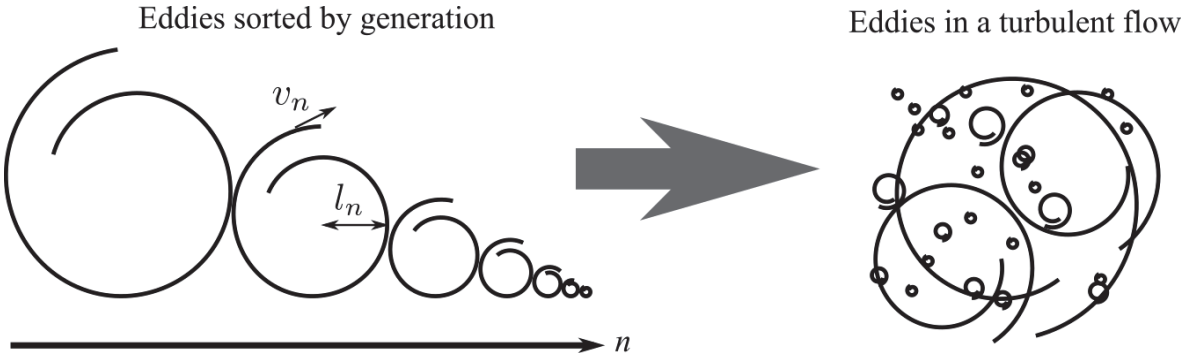


Figure 1.2 – Illustration of eddies in turbulence. Big eddies generate smaller eddies and they are all mixed in the same volume.

The n-th generation of eddies has l_n for characteristic size, v_n for characteristic velocity, t_n for turnover time and E_n for kinetic energy. These quantities are linked by the following

relations ($\simeq$ stands for equality in order of magnitude).

$$E_n \simeq v_n^2$$

$$v_n \simeq l_n/t_n$$

The rate of transfer from generation n to generation $n+1$ is denoted ε_n. From the previous notations we can state

$$\varepsilon_n \simeq E_n/t_n \simeq v_n^3/l_n$$

At each time unit, a proportion ε_n of eddies of generation n will be split in two smaller eddies of generation $n+1$. This transfer is repeated for all scales.

Kolmogorov, in its original paper (Kolmogorov, 1941), makes the assumption that $\varepsilon_n = \varepsilon$ does not depend on n. From this assumption ensues an expression of the Eulerian structure function (so-called *2/3 law*, Frisch (1995); Monin and Yaglom (1963)):

$$\mathbb{E}\left[(U_{t,x+r} - U_{t,x})^2\right] = C_0 \varepsilon^{2/3} r^{2/3} \tag{1.17}$$

with $U_{t,x}$ the Eulerian wind, x a point in the fluid and r a displacement from that point.

It is important to precise the equation (1.17) is valid for Eulerian wind. Indeed, for Lagrangian wind, the result is not the same (Yakhot, 2008; Mordant et al., 2001):

$$\mathbb{E}\left[(V_{t+\tau} - V_t)^2\right] = C_0 \varepsilon \tau \tag{1.18}$$

with V_t the Lagrangian wind ($V_t = U_{t,X_t}$). Another difference between equations (1.18) and (1.17) is that one is a spatial structure function, while the other is time structure function. The constant C_0 is called the Kolmogorov's constant. Although presented as universal, many studies come out on different values (DU, 1997; Du et al., 1995). In this study, the chosen value is $C_0 = 2.1$, according to Pope (1994). Its influence will be assessed in the sensitivity analysis.

An eddy of size l would be described from the Eulerian point of view. Its kinetic energy is given by the equation (1.17) by taking x the center of the eddy and $r = l$:

$$E(l) = C_0 (\varepsilon l)^{2/3} \tag{1.19}$$

with l the characteristic size of the eddy of kinetic energy $E(l)$ and C_0 the so-called *Kolmogorov constant*. The rate of transfer from different size ε is assumed to not depend on the size (see Kolmogorov (1941); Frisch et al. (1978)). By taking the Fourier transform of 1.19, we get the -5/3 spectrum:

$$\begin{aligned} E(k) &= \int_0^{+\infty} E(l) e^{-ilk} dl \\ &= \int_0^{+\infty} C_0 (\varepsilon l)^{2/3} e^{-ilk} dl \\ &= C_0 \varepsilon^{2/3} \int_0^{+\infty} \left(\frac{\alpha}{k}\right)^{2/3} e^{-i\alpha} \frac{d\alpha}{k} \\ &= C_0' \varepsilon^{2/3} k^{-5/3} \end{aligned}$$

The energy is coming in at the largest scales, of size comparable to boundary layer height. With common order of magnitude ($l_0 \simeq 1\text{km}$, $v_0 \simeq 1\text{m}\cdot\text{s}^{-1}$ and $\nu \simeq 10^{-5}\text{m}^2\cdot\text{s}^{-1}$) , one can estimate the Reynolds number at that scale:

$$Re_0 = \frac{v_0 l_0}{\nu} \simeq 10^8$$

This very large Reynolds number confirms the atmospheric boundary layer is turbulent. At the smallest scales, the viscosity become important and dissipates the eddies in heat. The scale at which the eddies are dissipated is called the Kolmogorov scale and denoted l_K. From the interpretation of the Reynolds number, it should be close to 1 at such scale. Moreover, the characteristic speed can be expressed with the relation $\varepsilon \simeq v_K^3 / l_K$. It is a way to estimate the Kolmogorov scale:

$$\begin{aligned} Re_K = 1 &\iff \frac{v_K l_K}{\nu} \simeq 1 \\ &\iff \frac{(\varepsilon l_K)^{1/3} l_K}{\nu} \simeq 1 \\ &\iff l_K \simeq \left(\frac{\nu^3}{\varepsilon}\right)^{1/4} \end{aligned}$$

with common order of magnitude ($\varepsilon \sim 10^{-1}\text{m}^2\cdot\text{s}^{-3}$, $\nu \sim 10^{-5}\text{m}^2\cdot\text{s}^{-1}$), it gives $l_K \simeq 3\text{mm}$. All the scale between l_0 and l_K represent the *inertial range* of turbulence. In the inertial range, the wind spectrum decreases continuously (it is represented in semilog-scale in figure 1.3). This a characteristic feature of atmospheric turbulence and a very efficient way to check if measurements are noisy. If the spectrum is flat at high frequencies, it is the sign that some noise remains in the signal.

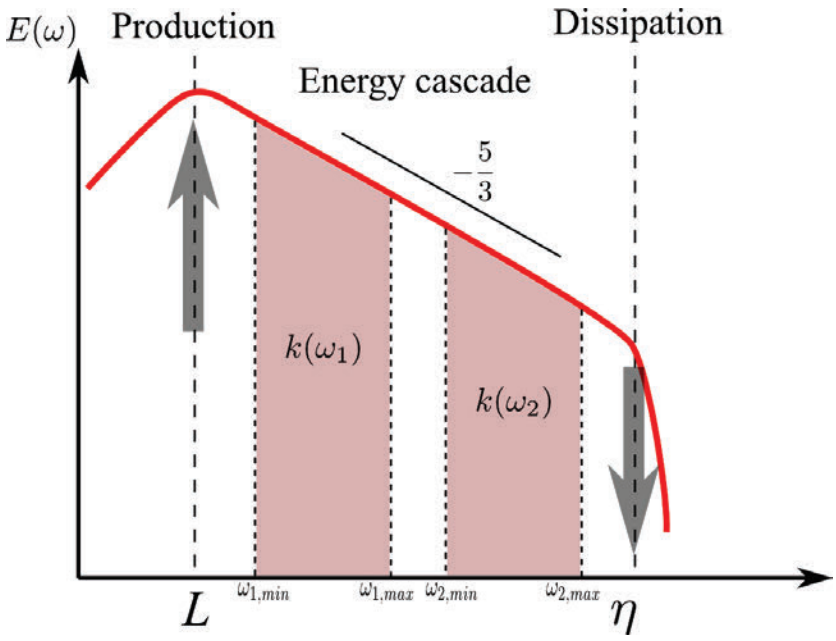


Figure 1.3 – Kolmogorov spectrum: wind spectrum has a steady -5/3 slope in the inertial range. On the x-axis are frequencies in log scale, on the y-axis is power in decibels.

1.3 Stochastic Lagrangian models

1.3.1 Introduction to Lagrangian modelling

The principle of a Lagrangian model is to describe the fluid motion by the movement of a set of particles. It is opposed to Eulerian models, such as grid points models used in numerical weather prediction, which describe the fluid by a set of "probe" where the fluid speed is known. Even though Eulerian point of view is the most in use in fluids dynamics (it matches the constraint of having instrument at fixed location and boundary conditions), Lagrangian modelling still has attractive features. It is very useful to describe the propagation of pollutants (Rotach, 2001) or reactive compounds (Pope, 1985). Reversely, they are used to build footprint models (Schmid, 2002). They are also used in more micro-scale processes, such as inter-particles collision in a turbulent flow (Sommerfeld, 2001). For a historical review of Lagrangian modelling see Thomson and Wilson (2013). For a more technical review, see Wilson and Sawford (1996).

We consider only 1-dimensional model, since we do not use higher dimension in this work, but higher dimensional models are written very similarly Pope (1985). Our instrument measures only vertical velocity, hence a 1-dimensional model is enough. All stochastic Lagrangian models are of the form 1.20.

$$\begin{cases} dX_t &= V_t dt \\ dV_t &= a(X_t, V_t, t)dt + b(X_t, V_t, t)dB_t \end{cases} \tag{1.20}$$

The first equation tells the particles are displaced according to their speed. The second equation accounts all the forces acting on the particle. The term $a(X_t, V_t, t)$ is dedicated to larger forces, which induce a global trend. The term $b(X_t, V_t, t)$ is dedicated to small scales forces, which have no global trend by add dispersion among fluid particles (dB_t is a Brownian motion infinitesimal increment).

The study of Lagrangian models starts in the early 20^{th} century, with the Langevin equation mentioned in the appendix A. The Langevin equation was at the origin the application of the second Newton's law to a large particle dropped into a fluid at rest. The innovation of Langevin was to introduce a "random force" which describe the microscopic effects of the surrounding fluid. Later it has been formulated in terms of stochastic processes, as it has been done in the appendix A (equation A.18). The Langevin equation is recalled here:

$$dV_t = -\frac{1}{T_L} V_t dt + \sigma dB_t$$

We have seen that the process solution of this stochastic differential equation is the Ornstein-Ulhenbeck process. Its average is an exponential function decreasing at the rate T_L and its variance is bounded. Asymptotically (that is to say after few T_L), it is stationary and ergodic. It means that the stochastic process can be approached by a single realisation, which is what one get from measurements. The asymptotic variance has been calculated in the appendix A: $\sigma_\infty^2 = \sigma^2 \frac{T_L}{2}$. It can be used as parameter instead of σ. The Langevin equation is thus written as

$$dV_t = -\frac{1}{T_L} V_t dt + \sqrt{\frac{2\sigma_\infty^2}{T_L}} dB_t \tag{1.21}$$

The advantage of this change is that σ_∞^2 is more interpretable and accessible under physical assumptions. Indeed σ_∞^2 stands for the stationary variance of one component of the wind. When the turbulence is 3-dimensional and isotropic, it is linked to the TKE: $\sigma_\infty^2 = \frac{1}{3}(\overline{u'^2} + \overline{v'^2} + \overline{w'^2}) = \frac{2}{3}k$.

1.3.2 The stochastic Lagrangian model used

The stochastic Lagrangian model used in the recontruction has been contructed by Baehr (2008) from the model of Pope (2000) (also Das and Durbin (2005) for the 3-dimensional version). Pope establishes his 1-dimensional model from a Langevin equation (equation 21 of Pope (1994)).

$$dV_t = -\frac{1}{T_L} V_t dt + \sqrt{\frac{2\sigma_\infty^2}{T_L}} dB_t$$

From Mordant et al. (2001) (equation 4) the Lagrangian structure function in the inertial domain is expressed with C_0 the Kolmogorov's universal constant and the eddy dissipation

rate ε:

$$\mathbb{E}\left[(V_{t+s} - V_t)^2\right] = C_0 \varepsilon s \tag{1.22}$$

In the appendix A, we have seen the variogram of this process is

$$\gamma_V(s) = \sigma_\infty^2 \left(1 - e^{-\frac{s}{T_L}}\right) \simeq \sigma_\infty^2 \frac{s}{T_L}$$

The two results are consistent (linear variogram) when $s \ll T_L$. We identify the terms to get an expression of T_L. If the turbulence is isotropic, the stationary variance is linked to the TKE: $\sigma_\infty^2 = \frac{2}{3}k$. Thus, the final expression for T_L is

$$T_L = \frac{4k}{3C_0\varepsilon} \tag{1.23}$$

The Langevin equation is thus written as follow:

$$dV_t = -\frac{3}{4} C_0 \frac{\varepsilon}{k} V_t dt + \sqrt{C_0 \varepsilon} dB_t \tag{1.24}$$

Then, Pope makes three changes on the equation 1.24 to get his model.

- Pressure force is taken into account.
- The restoring term no longer restore to 0 but to the Eulerian average wind $\langle U_{t,X_t} \rangle$.
- The coefficient $\frac{3}{4}C_0$ of the restoring term is changed into $\frac{1}{2} + \frac{3}{4}C_0$.

The resulting model is given in Pope (1994), equation 32:

$$dV_t = -\langle \nabla p_t \rangle \, dt - \left(\frac{1}{2} + \frac{3}{4} C_0\right) \frac{\varepsilon}{k} \left(V_t - \langle U_{t,X_t} \rangle\right) dt + \sqrt{C_0 \varepsilon} dB_t \tag{1.25}$$

This model is initially built for 3-dimensional homogeneous isotropic tubulence. In the atmosphere, the vertical component is much different than the two horizontal. The stratified model of Das and Durbin (2005) takes it into account. Baehr (2008) uses both Pope's and Das-Durbin's to make the 3-dimensional model used for the reconstruction. The difference with Das-Durbin's is that the temperature evolution has been removed since the reconstruction filters only wind. This model is used in Baehr (2010); Rottner (2015); Rottner et al. (2017); Suzat et al. (2011). In this work, we consider only the vertical component. Hence the model used is very close to 1.25, at the difference that the pressure force is replaced by a term A_t which accounts for all external forces acting on the vertically moving particles (flottability, pressure). The coefficient of the restoring term is denoted $C_1 = \left(\frac{1}{2} + \frac{3}{4}C_0\right)$ and will be considered as an independent constant. Thus, the model in use for this work is

$$dV_t = -A_t dt - C_1 \frac{\varepsilon}{k} \left(V_t - \langle U_{t,X_t} \rangle\right) dt + \sqrt{C_0 \varepsilon} dB_t \tag{1.26}$$

It is applied to guide independently a set of N particles. Each particle has a speed

and position (X_t^i, V_t^i) which is realisation of the random variable (X_t, V_t). For any particle $i \in [\![1, N]\!]$, its speed and position are given by

$$\left\{ \begin{array}{rcl} dX_t^i & = & V_t^i dt \\ dV_t^i & = & -A_t^i dt - C_1 \frac{\varepsilon^i}{k^i} \left(V_t^i - \left\langle U_{t,X_t^i} \right\rangle \right) dt + \sqrt{C_0 \varepsilon^i} dB_t \end{array} \right. \tag{1.27}$$

This model is in continuous time. In practice, it is used to move from one time step to the next. The discretization of time is done using an explicit Euler scheme. It induces an error, that we suppose Gaussian of standard deviation σ^X added on the position of the particles. Consider we have N_t time steps. For any $t \in [\![1, N_t]\!]$ the discrete time model is given by:

$$\left\{ \begin{array}{rcl} X_{t+1}^i & = & X_t^i + V_t^i \Delta t + \sigma^X \zeta_t^X \\ V_{t+1}^i & = & V_t^i - A_t^i \Delta t - C_1 \frac{\varepsilon^i}{k^i} \left(V_t^i - \left\langle U_{t,X_t^i} \right\rangle \right) \Delta t + \sqrt{C_0 \varepsilon^i \Delta t} \zeta_t^V \end{array} \right. \tag{1.28}$$

with ζ_t^X and ζ_t^V centred Gaussian random variables of variance 1.

Some terms are not explicit in this model: the large scale action A_t, the eddy dissipation rate ε, the turbulent kinetic energy k and the Eulerian wind average $\langle U_{t,X_t} \rangle$. The next section explains how they are estimated from the set of particles.

1.3.3 Closure of the model and local average

Some quantities in the Lagrangian model have not been explicited yet. There are four of them:

- The large scale effect A_t.
- The eddy dissipation rate ε.
- The Eulerian average wind $\langle U_{t,X_t} \rangle$.
- The turbulent kinetic energy k.

A solution to close the model is to take such quantities from outside of the particle system. This is what is done by Rottner et al. (2017) to get sub-grid estimation of turbulence. The large scale effect A_t and the eddy dissipation rate ε are taken from a large scale grid points model and the turbulent kinetic energy k is estimated from the particles, which gives the sub-grid contribution to turbulence, as illustrated by the figure 1.4. In the paper, the Meso-NH model is used at two different resolutions: a coarse one ($\Delta x = 160m$) which stands for the large scale, and a fine one ($\Delta x = 40m$), used for the assessment of the sub-grid contribution. See figure 2 of the paper. The purpose of the paper is to perform downscaling from a grid point model to sub-grid particle system.

The purpose of reconstruction is different: get fast estimate of wind and turbulence from

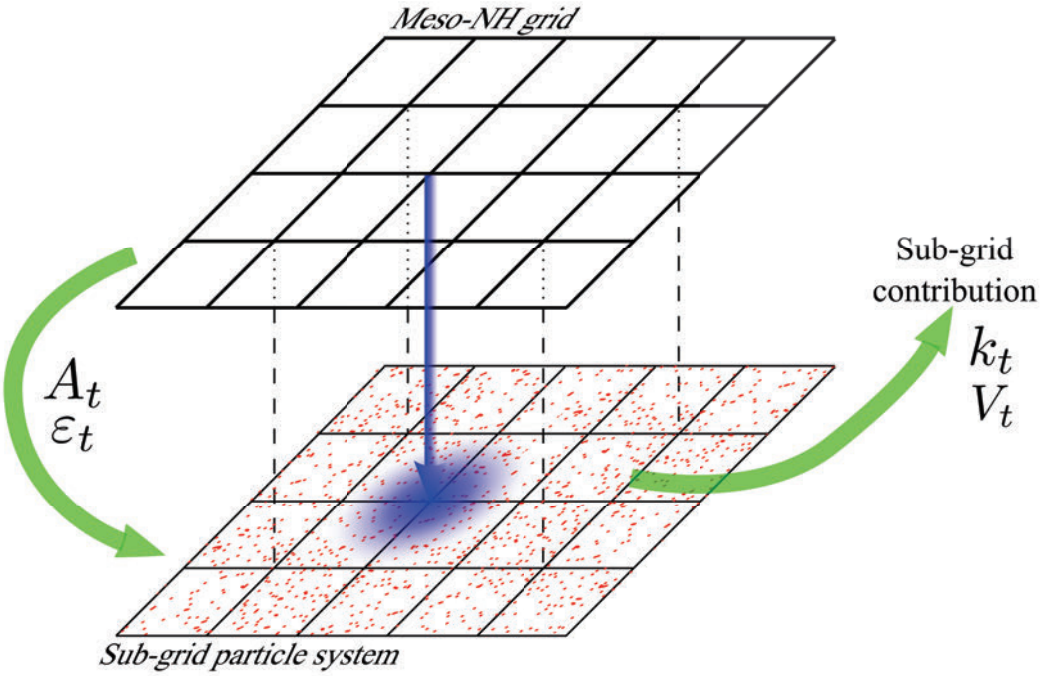


Figure 1.4 – For downscaling applications, A_t and ε are taken from a grid point model and the sub-grid particle system is driven by these commands. Particles provide sub-grid contribution k and V_t to turbulence and wind. Figure from Rottner et al. (2017).

lidar measurement. No model is involved, hence the commands of the model must be estimated from the particle system.

The turbulent kinetic energy estimation will be treated in a dedicated section since there are many things to say about it. First, we treat the Eulerian average wind, then the large scale effect, and eventually the eddy dissipation rate.

1.3.3.1 Eulerian average wind

The problem to estimate the Eulerian average wind is that the particle system provides only Lagrangian quantities. The question is how to estimate Eulerian quantities with only Lagrangian information? It has been addressed by Baehr (2010) with the formalisation of acquisition processes. The couple $(X_t, U_{t,x})$ is an acquisition system for the random field of velocity (definition 1.1 of the paper). The speed of particles $V_t = U_{t,X_t}$ is an acquisition process of the velocity field (definition 1.2 of the paper). The Eulerian average wind, denoted $\langle U_{t,X_t} \rangle$ is actually the expectation of the Eulerian velocity field at the point where is the particle.

$$\langle U_{t,X_t} \rangle \overset{def.}{=} \mathbb{E}\left[U_{t,x} | X_t = x\right]$$

This expectation can be expressed with the Lagrangian quantities X_t and V_t knowing that $V_t = U_{t,X_t}$.

$$\mathbb{E}\left[U_{t,x}|X_t = x\right] = \mathbb{E}\left[V_t|X_t\right] = \int v\mathbb{P}\left(V_t \in dv|X_t\right)$$

with the probability $\mathbb{P}\left(V_t|X_t\right)$ given by Bayes' formula:

$$\mathbb{P}\left(V_t \in dv|X_t = x\right) = \frac{\mathbb{P}\left(V_t \in dv, X_t \in dx\right)}{\int \mathbb{P}\left(V_t \in dv, X_t \in dx\right)}$$

The problem with this expression is that it is impossible to estimate if there is no particles on a given point of the space. Ideally, with $N = +\infty$ such that there are particles at every point in the space, then $\langle U_{t,X_t}\rangle$ is the average speed of all the particles located at the point x. In practice, even with N very large, there are points uncovered by particles. To bypass this problem, the condition $X_t = x$ is relaxed with a regularisation kernel. That is to say, instead of computing the average of all the particles located at the point x, we compute the average of all the particles in a neighbourhood of x. Baehr (2010) suggested a Gaussian neighbourhood, as represented in figure 1.5, in accordance with Kraichnan's turbulence theory.

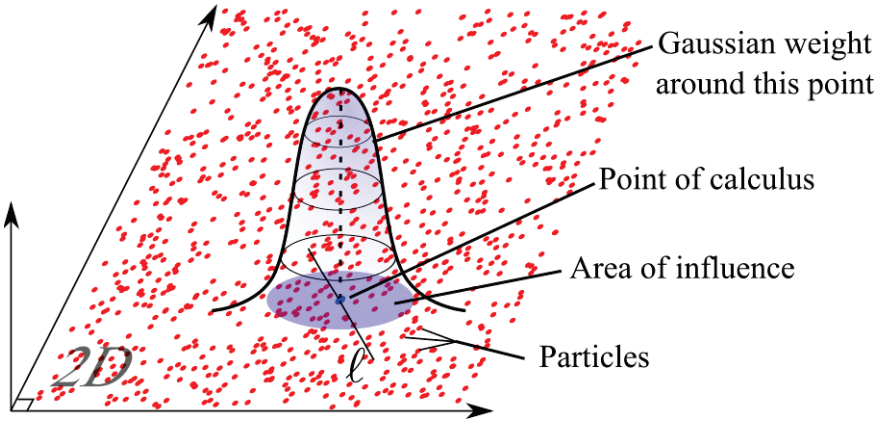


Figure 1.5 – Illustration of the Gaussian local average on a 2-dimensional system of particles.

The regularisation kernel used is thus a Gaussian function with standard deviation ℓ.

$$\Phi_\ell(x, y) = \exp\left(-\frac{(x-y)^2}{2\ell^2}\right) \tag{1.29}$$

The Eulerian average wind is then a weighted average on all the particles.

$$\langle U_{t,X_t}\rangle^\ell = \frac{\int v\Phi_\ell(X_t, x)\mathbb{P}\left(V_t \in dv, X_t \in dx\right)}{\int \Phi_\ell(X_t, x)\mathbb{P}\left(V_t \in dv, X_t \in dx\right)}$$

The regularisation fades when ℓ gets smaller:

$$\langle U_{t,X_t}\rangle = \lim_{\ell\to 0}\langle U_{t,X_t}\rangle^\ell$$

Since the particle system consists in a given number of particles, this term is estimated with the particles available. Given a particle of number i, the Eulerian average wind is approached by the local average on particles around

$$\langle U_{t,X_t} \rangle^\ell \simeq \left\langle V_t^i \right\rangle^\ell = \frac{\sum_{j=1}^N V_t^j \Phi_\ell(X_t^i, X_t^j)}{\sum_{j=1}^N \Phi_\ell(X_t^i, X_t^j)} \tag{1.30}$$

In the rest of this manuscript, the notation $\langle V_t^i \rangle^\ell$ will be used instead of $\langle U_{t,X_t} \rangle^\ell$ in the Lagrangian model. Although it hides the physical construction of the model, it better reflects what is done in practice.

1.3.3.2 Large scale effect

The large scale effect A_t should be given by external source of information. Since no other source of information is available, A_t must have a value which is consistent with the particle system. The solution proposed by Baehr (2010) is to take the average of the velocity increments (equation 4.3 of the paper). Indeed, if one takes the expectation of the stochastic Lagrangian model, one gets:

$$\mathbb{E}\left[dV_t\right] = -\underbrace{\mathbb{E}\left[A_t\right]}_{=A_t} dt - C_1 \underbrace{\mathbb{E}\left[\frac{\varepsilon}{k}\left(V_t - \langle V_t \rangle^\ell\right)\right]}_{\simeq 0 \text{ if } \ell \to 0} dt + \underbrace{\mathbb{E}\left[\sqrt{C_0 \varepsilon} dB_t\right]}_{=0}$$

The first term is deterministic. The second term goes to zero if the regularisation kernel is well set. The third term is centred. In the end, only the large scale effect remains. To get it, we only have to estimate $\mathbb{E}\left[dV_t\right]$. In the discrete time model, dV_t is approached by the increment of one time step. Such increment is known for all particles, thus the expectation is approached by the average over many particles. As the large scale effect may change with height, we only average increment of particles in at the same vertical layer.

With the notations introduced before, the estimation of the large scale effect is written:

$$A_t^i = -\frac{1}{\Delta t} \sum_{j \in B(z(i),t)} \frac{(V_t^j - V_{t-1}^j)}{|B(z(i),t)|}$$

1.3.3.3 Eddy dissipation rate

For the eddy dissipation rate, the problem is the same as for the large scale effect. In the absence of external source of information, a consistent value must be taken. In the generalized Langevin model (equation 1.20), the following relationship is commonly used:

$$\mathbb{E}\left[(dV_t)^2\right] = b(X_t, V_t, t)^2 \tag{1.31}$$

Since the term $b(X_t, V_t, t)$ is identified thanks to the equation 1.26, the equality 1.31 is written in discrete time as

$$\mathbb{E}\left[(\Delta V_t)^2\right] = C_0 \varepsilon \Delta t$$

The expectation is then approached by an ensemble average on the particles. As for the large scale trend, each vertical layer is distinguished. Thus, the following estimation for ε is consistent with the model:

$$\varepsilon_t^i = -\frac{1}{C_0 \Delta t} \sum_{j \in B(z(i),t)} \frac{(V_t^j - V_{t-1}^j)^2}{|B(z(i),t)|}$$

Other closure exist: Bernardin et al. (2009); Suzat et al. (2011) use a so-called $k-\varepsilon$ closure (Launder and Spalding, 1974), based on dimensional analysis, it links the EDR and the TKE through the relation

$$\varepsilon = \frac{C_\varepsilon}{l_m} k^{2/3}$$

with C_ε a constant and l_m a characteristic length (the mixing length is often in use). The resulting EDR depends on t. Although is has been omitted previously, this notation will be conserved.

1.4 Turbulent kinetic energy estimation

We have seen that the turbulent kinetic energy (TKE) is a valuable quantity to measure when one studies turbulence (definition 1.3). It is usually estimated by the variance on the time series of wind measurements. By its representation of the fluid with particles, the reconstruction system allows alternative computation that are still to qualify. Hence, beside TKE estimation from time series, this section broaches estimation of TKE from particle system. Some of this material was submitted in Rieutord et al. (2016).

1.4.1 Geometry of the problem

In this work we consider only vertical staring lidar. The wind is measured at several heights (the resulting value is an average over a given layer, these layers are numbered according to their vertical level; see chapter 3 for more detail on the instrument). As the lidar stays at the same place, the measurements are Eulerian. The lidar provides times series at each vertical level $V^{\mathbf{o}}(z,t)$ ("$\mathbf{o}$" for *observation*). In addition to the measurements, numerical particles $(X^i, V^i)_{[\![1,N]\!]}$ are considered in the probe volume. They stand for fluid particles and are leaded by the stochastic Lagrangian model presented in the previous section. Particles, vertical levels and lidar are represented in the figure 1.6. In the whole manuscript, we adopt the following notations:

- N_z is the number of vertical levels.

- $z \in [\![1, N_z]\!]$ is the vertical level number.
- Δz is the size of a vertical level.
- $z(i) = \left\lfloor \frac{X^i - X_{min}}{\Delta z} \right\rfloor + 1$ is the vertical level number of particle i.
- $\mathcal{B}(z) = [X_{min} + (z-1)\Delta z, X_{min} + z\Delta z[$ is the layer numbered with the vertical level z.
- $B(z,t) = \{i,\ X_t^i \in \mathcal{B}(z)\}$ is the set of particles at the vertical level z at time t.

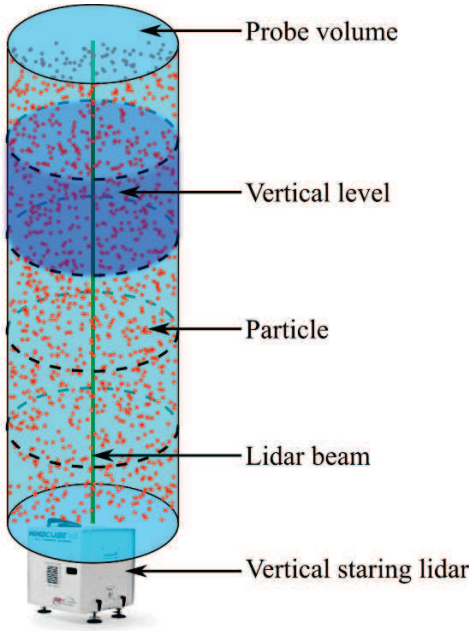


Figure 1.6 – Scanning geometry and vocabulary.

1.4.2 Average operators and TKE estimators

1.4.2.1 Time average

TKE profiles are obtained by computing the variance on a time series of wind measurements. The length of the time series required to get a single estimate of TKE is still an open question. It is 30 minutes in Darbieu et al. (2015), 15 minutes in Tucker et al. (2009), 3 or 5 min (depending on instrument) in O'Connor et al. (2010). This question is related to the stationarity and ergodicity properties, introduced in the appendix A. If the integration time is short, stationarity is a weak assumption but ergodicity is a strong assumption. If the integration time grows, it ensures ergodicity but stationarity becomes a strong assumption. Lenschow et al. (1994) distinguish the error due to the use of a time average instead of an ensemble average (so-called systematic error) and the error due to the use of a single realization (so-called random error). They only consider stationary processes, they conclude the longer the better (to ensure ergodicity), with a $1/\tau$ decrease of the error. Pichugina et al. (2008) show

the non-stationarity is a severe issue for stable conditions. Because of it, they fail to apply Lenschow's results. Germano (1992) discuss the question in the framework of LES. In any case, a time average operator is used.

Definition 1.5 (Time average)**.**

With words *Given an integration time τ, the time average operator transforms any L^2 function of time defined on a subdivision of resolution $\Delta t < \tau$ into another L^2 function defined on a subdivision of resolution τ.*

Formally *Let τ be the integration time. Let $\mathcal{T}(\alpha) = \{t_1, t_2, ...$ such that $\forall i$, $t_{i+1} - t_i = \alpha\}$ be a subdivision of time of resolution α. Let $\phi \in L^2(\mathcal{T}(\Delta t))$ a time function known on $\mathcal{T}(\Delta t)$ with $\Delta t < \tau$. Then its time average $\overline{\phi}^\tau$ is defined on $L^2(\mathcal{T}(\tau))$.*

$$\begin{array}{rccl} \overline{\phi}^\tau : & \mathcal{T}(\tau) & \longrightarrow & \mathbb{R} \\ & t' & \longmapsto & \overline{\phi}^\tau(t') = \frac{1}{\tau}\sum_{t=t'}^{t'+\tau} \phi(t) \end{array}$$

Thus the time average is defined as

$$\begin{array}{rccl} \overline{\bullet}^\tau : & L^2(\mathcal{T}(\Delta t)) & \longrightarrow & L^2(\mathcal{T}(\tau)) \\ & \phi & \longmapsto & \overline{\phi}^\tau(t') = \frac{1}{\tau}\sum_{t=t'}^{t'+\tau} \phi(t) \end{array} \tag{1.32}$$

The time average has thus an effect on the resolution: the time average of a vector of size N_t will be of size $N_t \frac{\Delta}{\tau}$. This average operator is the one classically used, because it requires only a time series of measurement, that any device provides. Using the time average operator, the TKE is estimated on time series and designed as *time TKE* (TTKE).

Definition 1.6 (TTKE: time turbulent kinetic energy)**.**

$$k^T(z,t') = \frac{1}{2}\overline{\left(V^{\mathbf{o}}(z,t) - \overline{V^{\mathbf{o}}(z,t)}^\tau\right)^2}^\tau$$

1.4.2.2 Spatial average

With the use of a Lagrangian model, several independent particles are at different location at a given time, as shown in figure 1.6. Unlike grid points, which are always at the same location, particles move from a time step to another. Hence, instead of computing the variance on the time series of the wind at a fixed location, one can define an area and compute the variance of the speed of the particles in this area.

Definition 1.7 (Spatial average)**.**

With words *Given an vertical level z of size Δz, the spatial average is the average over all the particles at this vertical level.*

Formally *Let $\phi \in L^2([\![1, N]\!])$ be a function known on the set of particles $(X^i, \phi^i)_{[\![1,N]\!]}$ and $B(z) = \{i,\ X^i \in \mathcal{B}(z)\}$. Then its spatial average $\langle\phi\rangle$ is defined on $L^2([\![1, N_z]\!])$.*

$$\langle\phi\rangle : \begin{array}{ccc} [\![1, N_z]\!] & \longrightarrow & \mathbb{R} \\ z & \longmapsto & \langle\phi\rangle(z) = \frac{1}{B(z)} \sum_{i \in B(z)} \phi^i \end{array}$$

Thus the spatial average is defined as

$$\langle\bullet\rangle : \begin{array}{ccc} L^2([\![1, N]\!]) & \longrightarrow & L^2([\![1, N_z]\!]) \\ \phi & \longmapsto & \langle\phi\rangle(z) = \frac{1}{B(z)} \sum_{i \in B(z)} \phi^i \end{array} \tag{1.33}$$

The spatial averaging also reduces the resolution: from one information for each particles, we got one information for each vertical level. This way of estimation gives a *spatial TKE* (STKE).

Definition 1.8 (STKE: spatial turbulent kinetic energy)**.**

$$k^S(z,t) = \frac{1}{2} \left\langle \left(V_t^i - \left\langle V_t^i \right\rangle \right)^2 \right\rangle$$

1.4.2.3 Local spatial average

To compare Eulerian quantities to Lagrangian quantities a regularisation kernel was introduced in the previous section. Moreover, another spatial average has been presented in the description of the Lagrangian model. This other spatial average that all particles into account, but weight them with a Gaussian (see figure 1.5). Depending on the parameter ℓ, this average is more local than the previous one. Thus it will be designed as *local spatial TKE* (LSTKE).

Definition 1.9 (Local spatial average)**.**

With words *Given a location x, the local spatial average at this point is the average over all the particles with a Gaussian weight centred on x.*

Formally *Given a location x and $\ell > 0$. Let $\phi \in L^2([\![1, N]\!])$ be a function known on the set of particles $(X^i, \phi^i)_{[\![1,N]\!]}$ and $\Phi_\ell(x, y) = \exp\left(-\frac{(x-y)^2}{2\ell^2}\right)$. Then its local spatial average $\langle\phi\rangle^\ell$ is defined on $L^2([X_{min}, X_{max}])$.*

$$\begin{array}{rccl} & [X_{min}, X_{max}] & \longrightarrow & \mathbb{R} \\ \langle\phi\rangle^\ell : & x & \longmapsto & \langle\phi\rangle^\ell(x) = \dfrac{\sum_{j=1}^N \phi^j \Phi_\ell(x, X_t^j)}{\sum_{j=1}^N \Phi_\ell(x, X_t^j)} \end{array}$$

Thus the local spatial average is defined as

$$\begin{array}{rccl} & L^2([\![1, N]\!]) & \longrightarrow & L^2([X_{min}, X_{max}]) \\ \langle\bullet\rangle^\ell : & \phi & \longmapsto & \langle\phi\rangle^\ell(x) = \dfrac{\sum_{j=1}^N \phi^j \Phi_\ell(x, X_t^j)}{\sum_{j=1}^N \Phi_\ell(x, X_t^j)} \end{array} \tag{1.34}$$

Conversely to the two previous, this operator does not reduce the resolution. It even enhances it in some sense: $|[X_{min}, X_{max}]| > |[\![1, N]\!]|$. The reason is that it is not a proper average operator: $\left\langle\langle\phi\rangle^\ell\right\rangle^\ell \neq \langle\phi\rangle^\ell$, while $\langle\langle\phi\rangle\rangle = \langle\phi\rangle$ and $\overline{\overline{\phi}^\tau}^\tau = \overline{\phi}^\tau$. None of the three operators presented satisfies the idempotence property of Reynolds' axioms (see table 1.1). But the local spatial average is not even a projection. Nevertheless, it converges towards the Eulerian average when ℓ goes to zero.

The stochastic Lagrangian model needs the TKE at the location of the particles. Hence, the local spatial average is applied only at the location $(X^i)_{[\![1,N]\!]}$. In the Lagrangian model, there is one value of TKE for each particle. For the turbulence estimation, we keep one value per vertical level. The resulting TKE estimator, the *local spatial TKE* (LSTKE) is thus:

Definition 1.10 (LSTKE: local spatial turbulent kinetic energy)**.**

$$k^{LS}(z,t) = \frac{1}{2}\left\langle\left\langle\left(V_t^i - \left\langle V_t^i\right\rangle^\ell\right)^2\right\rangle^\ell\right\rangle$$

Combining the time average operator and the spatial average operator, an additional TKE estimator have been defined. It is the temporal variance of the particles speed, along their trajectory. To get a single estimate by vertical level, the spatial average is applied next. It is denoted $k^{T'}$, the time TKE on particles (T'TKE).

Definition 1.11 (T'TKE: local spatial turbulent kinetic energy)**.**

$$k^{T'}(z,t) = \frac{1}{2}\left\langle\overline{\left(V_t^i - \overline{V_t^i}^\tau\right)^2}^\tau\right\rangle$$

This estimator will be useful in the comparison.

1.4.3 Comparison of TKE estimators

Four TKE estimates have been defined. The time series of such estimates at a given vertical level is shown in figure 1.7. The integration time was set to $\tau = 10$ minutes. The interaction length was set to $\ell = 10$ meters. One can see a clear difference of behaviour: estimates using time variance (TTKE and T'TKE) have a similar shape, evolving slowly in comparison to estimates using spatial variances (STKE and LSTKE). The effect of average on resolution is visible too: spatial variances are at much higher resolution than time variance. Nevertheless, it does not look comparable.

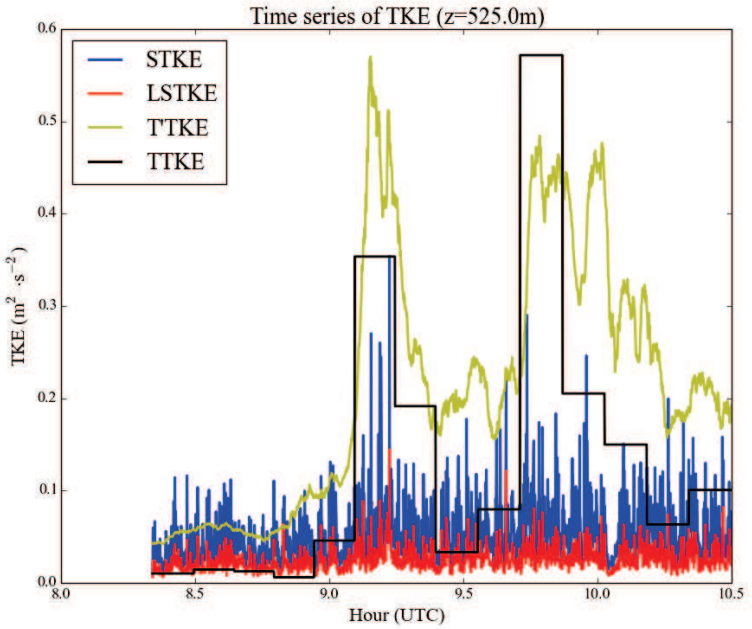


Figure 1.7 – Time series of the four TKE estimates at a given vertical level. Time variances, TTKE (black) and T'TKE (yellow), and spatial variances, STKE (blue) and LSTKE (red), are represented.

There is a disconnection between temporal evolution and spatial evolution. The connection between temporal evolution and spatial evolution in turbulence is usually made thanks to the Taylor's frozen turbulence hypothesis. This hypothesis states the eddies are advected by the flow such that measuring at t and $t + \Delta t$ is the same as measuring the eddies at x and $x - \Delta x$. It holds only if the mean wind is dominant over the fluctuation:

$$w = \overline{w} + w' \quad \text{with} \quad \overline{w} \gg w'$$

But the mean vertical component of the wind in the atmosphere is zero in the absence of convection or at large scale ($\overline{w} = 0$). Since the lidar measures only the vertical component of

the wind, there is no chance for the Taylor hypothesis to be satisfied. The eddies are advected by horizontal wind (unreachable by this lidar), thus eddies at z and $z + \Delta z$ are decorrelated. It explain why spatial variances are underestimated compared to the time variances. This is sustained by figure 1.8, which depicts the particles at a given vertical level and the trajectory of one of them picked at random. One can see the particles located at the same vertical level are quite similar. Conversely, through time, the selected particles will cross several vertical level and have very different speed. In the end, the time variance is logically higher than the space variance.

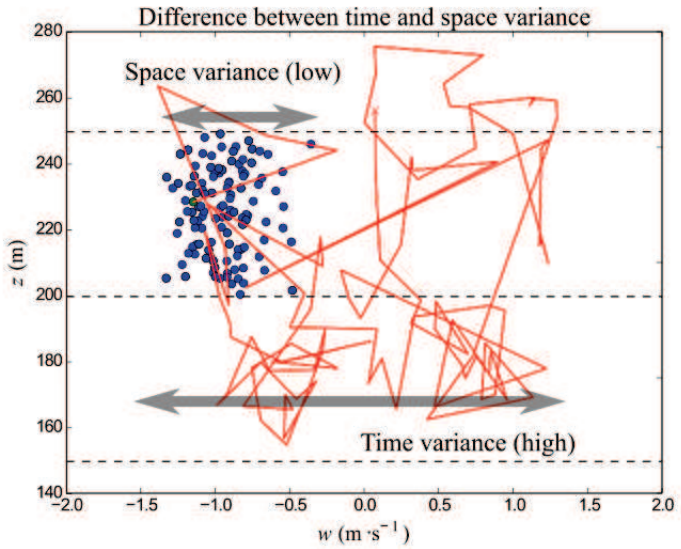


Figure 1.8 – Particles at the same vertical at t and trajectory of one of them during the 50 next time steps.

1.5 Conclusion

This chapter was intended as both an introduction to readers not familiar with turbulence and the place to discuss some of the features specific to the reconstruction (Lagrangian modelling, TKE estimations). The two first section sustained the first goal, while the two last sustained the second.

Eulerian and Lagrangian descriptions are two referential (one is fixed, the other goes with the flow). Eulerian is the most in use, but the particles in the reconstruction need a Lagrangian model. Fluids movements are described by the Navier-Stokes equations. They are simplified to be solvable for the target problem. Turbulence, our target problem, has the particularity to be random. Reynolds average splits up the deterministic trend and the random fluctuations. But this split induces a loss of information and the resulting system of equation (Reynolds system) is unclosed. To close it, semi-empirical closure exist. They highlight the importance of the turbulent kinetic energy (TKE) and the eddy dissipation rate (EDR) for the study of turbulence. The consideration of structure functions (variance of speed increments) yields to the Kolmogorov' spectrum with a famous -5/3 slope in log-log scale. This continuous decrease of the spectrum denotes the permanent transfer of energy from the large scales to the small scales. All these features are combined in a Langevin equation (seen in chapter A) to build the stochastic Lagrangian model used for the particles.

The reconstruction is based on the representation of the fluid with a set of particles. This study is limited to 1-dimensional vertical movements. In the stochastic Lagrangian model, some Eulerian terms are to estimate. There are approached thanks to a Gaussian regularization kernel of parameter ℓ (homogeneous to an interaction length). This representation with particles allows several way of estimation for the turbulent kinetic energy (TKE). The usual way (with only one measurement point) is to compute the variance of the wind time series. It works well as long as the turbulence is stationary, but there is no consensus on how long the integration time, τ, should be. With the particles of the reconstruction, the TKE can be estimated by a space variance, either on a whole vertical level, either using the Gaussian regularization kernel. Four estimators of TKE are defined (TTKE, STKE, LSTKE, T'TKE) and compared. Space variances (STKE and LSTKE) are no comparable to the time variances (TTKE, T'TKE) because the Taylor's hypothesis of frozen turbulence does not apply here: eddies are advected horizontally while the lidar provides only vertical velocity measurements.

Chapter 2

Bayesian filtering

Contents

2.1 Introduction to non-linear filtering

2.1.1 The filtering problem

The general problem of filtering is to remove noise from a signal. In Bayesian filtering, a prior information on how the signal should behave is given. Since, in practice, the signal we are dealing with are discrete, we will focus only on discrete signals. The time step is denoted t.

We are interested in the knowledge of a quantity X evolving with time (it can be the position of satellite, the price of a ton of wheat, the wind at a particular location...). We make the assumption (slightly philosophical) that its evolution can be described by some equation:

$$X_{t+1} = \mathscr{F}_t(X_t) \tag{2.1}$$

The function $\mathscr{F}_t$ is not perfectly known, and it is approached by another function f_t: the model. By doing this, we introduce an error (random by essence) ϵ_t^X. Thus $(X_t)_t$ is modelled by a stochastic process such as

$$X_{t+1} = f_t(X_t, \epsilon_t^X) \tag{2.2}$$

with

$$f_t : \begin{array}{ccc} \mathbb{R}^d \times \mathbb{R}^d & \longrightarrow & \mathbb{R}^d \\ (X, \epsilon) & \longmapsto & f_t(X, \epsilon) \end{array}$$

This process is only known through some observations of it, that we denote as $(Y_t)_t$. Ideally, there is a direct link between the observation and the quantity observed :

$$Y_t = \mathscr{H}_t(X_t) \tag{2.3}$$

But this link is not perfect, it usually relies on assumptions, or approximations made in the measurement design. The theoretical function $\mathscr{H}_t$ is replaced by an existing function h_t, and it brings an error ϵ_t^Y (the measurement error).

$$Y_t = h_t(X_t, \epsilon_t^Y) \tag{2.4}$$

with

$$h_t : \begin{array}{ccc} \mathbb{R}^d \times \mathbb{R}^m & \longrightarrow & \mathbb{R}^m \\ (X, \epsilon) & \longmapsto & h_t(X, \epsilon) \end{array}$$

In the case where there are less observations than unknowns[1], the problem can be seen as in figure 2.1. The target state X_t lives in the state space E of dimension d. The model describes the trajectory of the system into this space (the spiralish curve). The observations belong in a space E' of reducted dimension $m < d$. The aim of filtering is to retrieve the volumic spiral from the flat spiral.

Hence, the knowledge of X is tarnished by errors. Some errors come from the model, others come from measurements. But both model and measurements bring valuable information about X. The problem of Bayesian filtering is to get the best knowledge of X from imperfect measurements *and* imperfect modelling. If we denote the so-called historical process of X (resp. Y) is denoted $X_{0:t} = (X_0, ..., X_t)$ (resp. $Y_{0:t} = (Y_0, ..., Y_t)$). The targets of the filtering problem are the *a priori* and *a posteriori* probability laws (respectively equations (2.5) and (2.6)).

$$\eta_t = \mathbb{P}\left(X_t | Y_{0:t-1}\right) \tag{2.5}$$

$$\hat{\eta}_t = \mathbb{P}\left(X_t | Y_{0:t}\right) \tag{2.6}$$

That is to say, the filtering problem is solved when, for any test function ϕ, we can estimate

[1]For example, it is the case for numerical weather prediction: there are about $d = 10^7$ unknowns and about $m = 10^6$ observations.

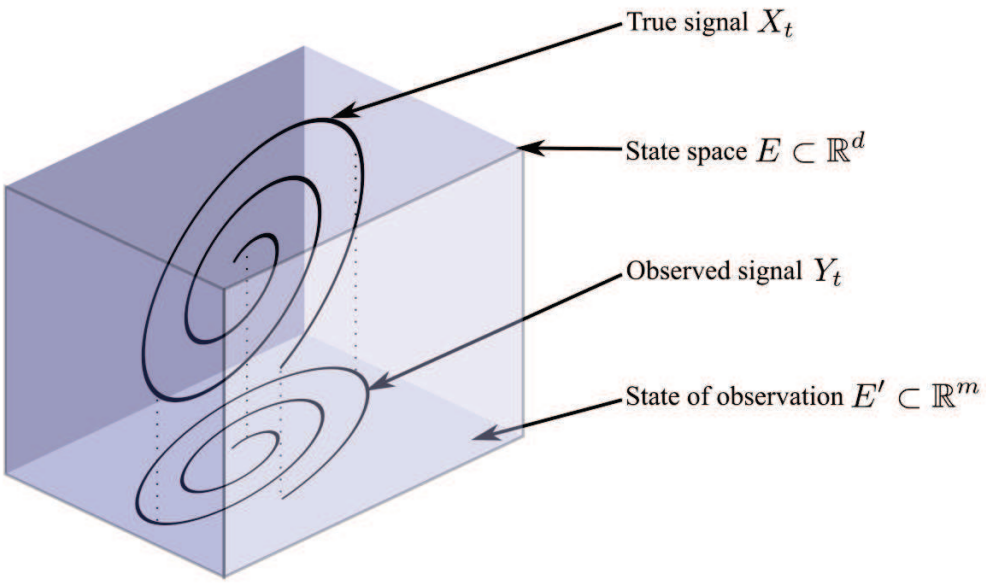


Figure 2.1 – Visually, Y_t can be seen as the projection of X_t onto a plan accessible to measure.

the integrals 2.7 and 2.8 (distribution notation).

$$\eta(\phi) := \mathbb{E}\left[\phi(X_t)|Y_{0:t-1}\right] = \int \phi(x)\eta_t(dx) \tag{2.7}$$

$$\hat{\eta}_t(\phi) := \mathbb{E}\left[\phi(X_t)|Y_{0:t}\right] = \int \phi(x)\hat{\eta}_t(dx) \tag{2.8}$$

In particular, the cases $\phi(x) = x$ and $\phi(x) = x^2$ are interpreted easily. The mean state $X_t^{\mathbf{e}} := \mathbb{E}\left[X_t|Y_{0:t}\right]$ gives the central trend of the stochastic process X_t. It is the average trajectory of all realisations of X_t which fit with the observations $Y_{0:t} = y_{0:t}$. The variance $\mathbb{E}\left[(X_t)^2|Y_{0:t}\right]$ is a score of the dispersion of such realisations. Both give valuable and easy to interpret information. In the Gaussian case, they are the only information needed to describe completely the probability laws η_t and $\hat{\eta}_t$.

The resolution of this problem requires some assumptions. Observations are assumed to have the following properties:

- Conditionally on the hidden state, observations are independent.

$$\mathbb{P}\left(Y_{0:t} \in dy_{0:t}|X_{0:t} = x_{0:t}\right) = \prod_{k=0}^{n} \mathbb{P}\left(Y_k \in dy_k|X_k = x_k\right)$$

- Y_t depends only on the current (hidden) state.

$$\mathbb{P}\left(Y_t \in dy_t|X_{0:t} = x_{0:t}\right) = \mathbb{P}\left(Y_t \in dy_t|X_t = x_t\right)$$

- The law Y_t given X_t has a density with respect to a measure Λ independent of the state X.

$$\mathbb{P}\left(Y_t \in dy_t | X_t = x_t\right) = g_t(x_t, y_t)\Lambda(dy_t)$$

When the previous properties are satisfied, $\{Y_k\}_{k\in\mathbb{N}}$ is called a *memoryless channel* (Le Gland, 2009). The function $g_t(x_t, y_t)$ is the *likelihood* of the observations. Most of the time, the observations are assumed to have Gaussian error, that is to say :

$$g_t(x_t, y_t) = (2\pi)^{-1/2} \exp\left(-\frac{1}{2}\|y_t - h_t(x_t, 0)\|^2_{R_t}\right)$$

Although the likelihood is the probability of the observations given the a priori state, it is seen as a function of the a priori state. As a consequence, the notation $G_t(x) := g_t(x, y_t)$ will be adopted.

In summary, the filtering problem aims to find $\eta(\phi)$ and $\hat{\eta}_t(\phi)$ for any function ϕ, given a model of evolution f_t, the observation operator h_t and a set of observations $y_{0:t}$. *Linear filtering* gather all the techniques where the model of evolution and the observation operator are both linear functions. This case is of particular interest because very efficient solution exists (the Kalman family) and the linear assumption suits to many simple systems and is still a relevant first approach for non-linear systems. The solution given by the Kalman filters will be briefly presented, with their strength and limitations. Then we focus on non-linear filtering, and particles filters are presented.

2.1.2 Mutation and selection

The process $(X_t)_t$ solution of the system

$$\begin{cases} X_{t+1} &= f_t(X_t, \epsilon_t^X) \\ Y_t &= h_t(X_t, \epsilon_t^Y) \end{cases}$$

is a hidden Markov process: the random variable X_t depends only on its last state and the value of X is not reachable directly. It is only known through observations Y_t. The *a priori* and *a posteriori* probability laws will be estimated recursively from an initial distribution. The link between probability laws is made with stochastic kernels.

Definition 2.1 (Stochastic kernel)**.**

With words *A stochastic kernel is a probability of transition from a space to another. To each point of the departure space, it associates a probability measure on the arrival space.*

Formally *K is a stochastic kernel between the measurable space $(E, \mathcal{E})$ and $(F, \mathcal{F})$ if*

$$K : \begin{array}{ccc} E \times \mathcal{F} & \longrightarrow & [0,1] \\ (x, A) & \longmapsto & K(x, A) \end{array}$$

such that

- $\forall A \in \mathcal{F},\ K(\cdot, A) : x \mapsto K(x, A)$ *is a measurable function.*
- $\forall x \in E,\ K(x, \cdot) : A \mapsto K(x, A)$ *is a probability measure.*

This definition is illustrated by figure 2.2.

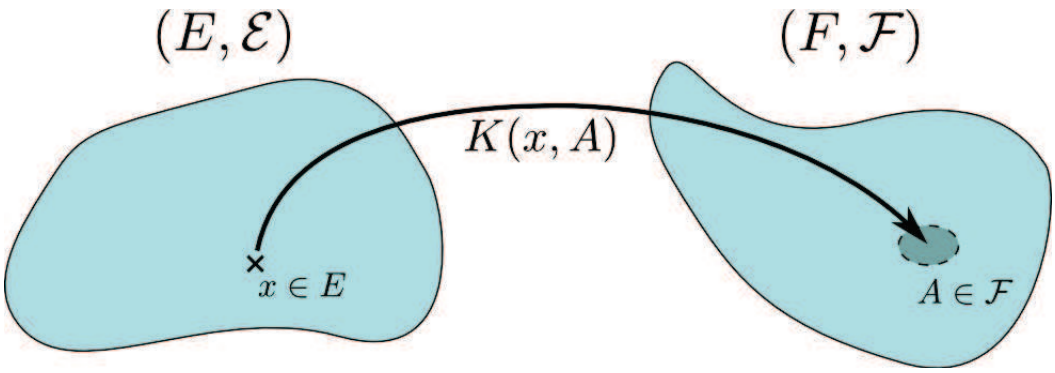


Figure 2.2 – The stochastic kernel K associates each point x in the departure space to a probability in the arrival space.

In the filtering problem, two stochastic kernels are used: one to go from $\hat{\eta}_{t-1}$ to η_t, another to go from η_t to $\hat{\eta}_t$. The stochastic kernel which transforms $\hat{\eta}_{t-1}$ into η_t is called the *mutation* kernel M_t. Given a position $\hat{x}_{t-1}$ at $t-1$, it tells the probability law of the system at the next time step.

$$M_t(x_{t-1}, dx_t) = \mathbb{P}\left(X_t \in dx_t \,|X_{t-1} = x_{t-1}, Y_{0:t-1} = y_{0:t-1}\right) \tag{2.9}$$

The stochastic kernel which transforms η_t into $\hat{\eta}_t$ is called the selection kernel S_{t,η_t}. Given a position $\bar{x}$ of the system before it is observed, it tells the probability law of the system after a new observation is assimilated.

$$S_{t,\eta_t}(\bar{x}, d\hat{x}) = \mathbb{P}\left(\hat{X}_t \in d\hat{x} \,|\bar{X}_t = \bar{x}\right) \tag{2.10}$$

$$\hat{\eta}_{t-1} \xrightarrow{M_t} \eta_t \xrightarrow{S_{t,\eta_t}} \hat{\eta}_t$$

In both transforms, the departure space and the arrival space are equal. For example, it the target process is a particle evolving in 2D space the illustration 2.2 can be changed in figure 2.3. The kernel associates to every point $x \in \mathbb{R}^2$ a probability to arrive close to $y \in \mathbb{R}^2$, in the area $dy \in \mathcal{B}(\mathbb{R}^2)$. The move from x to y is either due to the time evolution (the mutation kernel does this transform) or to the correction after a new observation arrived (the selection kernel does this transform). The subscript η_t on the selection kernel highlights that the selection kernel depends on the full probability law η_t, not only on the current state $\bar{x}$. The mutation or the selection associate to any point a probability law: that is why, without further simplification, they can only be described by stochastic kernels.

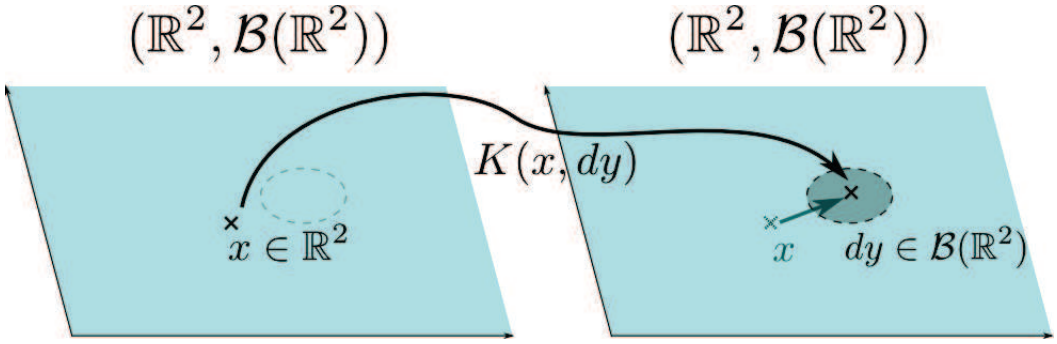


Figure 2.3 – Illustration of stochastic kernel to describe the movement of a particle in a 2D space.

The action of stochastic kernel is to integrate the conditional probabilities (2.9) and (2.10) over the departure point.

$$\eta_t(dx_t) = \int \hat{\eta}_{t-1}(dx_{t-1}) M_t(x_{t-1}, dx_t) \tag{2.11}$$

$$\hat{\eta}_t(d\hat{x}) = \int \eta_t(d\bar{x}) S_{t,\eta_t}(\bar{x}, d\hat{x}) \tag{2.12}$$

The expression of the mutation kernel 2.9 and the relation 2.11 directly come from the formula of total probability. For the selection kernel, the link between the expression 2.10 and the relation 2.12 will be discussed in the section dedicated to the particles filters.

The filtering algorithm will be thus be split in two steps: a first step of mutation, a second step of selection (illustrated in figure 2.4. The mutation step is a modelling part: only a better knowledge about the studied phenomenon can improve the mutation kernel. It is specific to a single phenomenon. The selection step, conversely, is more generic. Since the modelling of turbulence is discussed elsewhere, this section focuses on the selection kernel. On top the memoryless channel assumption, the selection step takes profit of additional assumption on f_t and h_t.

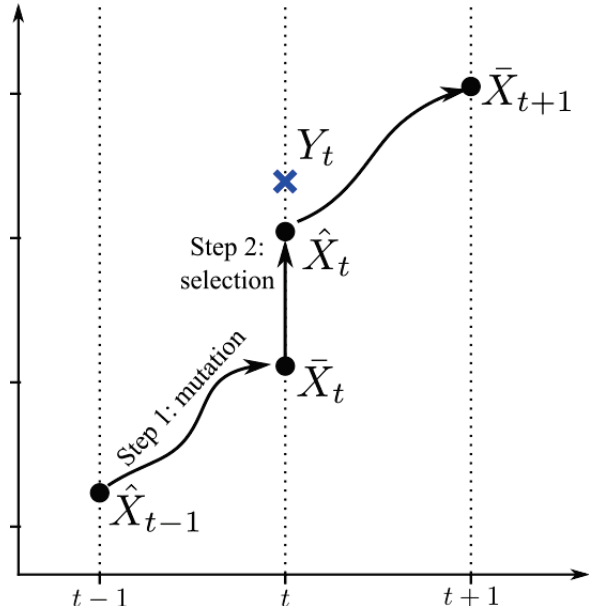


Figure 2.4 – Bayesian filtering has two steps for each time step. The first step propagate the *a posteriori* state of previous time $\hat{X}_{t-1}$ to the current time (mutation step). The second time step correct the *a priori* state $\bar{X}_t$ with the observation available at this time Y_t (selection step). And so on...

2.2 The Kalman family

Rudolf Kalman in 1960 found an exact solution to the filtering problem when the functions f_t and h_t are linear and on the errors ϵ_t^X and ϵ_t^Y are Gaussian and independent (Kalman et al., 1960). Under such assumptions, the probability law of the target is defined entirely by its mean and variance. We distinguish the mean and variance before the assimilation of the last observation ($\bar{X}_t, \bar{P}_t$: *a priori*) and after ($\hat{X}_t, \hat{P}_t$: *a posteriori*). The error covariance of ϵ_t^X is denoted $B_t \in \mathbb{R}^{d\times d}$ and the error covariance of ϵ_t^Y is denoted $R_t \in \mathbb{R}^{m\times m}$. It is assumed both B_t and R_t are known for any t. It yields to the following four unknowns.

$$\bar{X}_t = \mathbb{E}\left[X_t | Y_0, \ldots, Y_{t-1}\right] \tag{2.13}$$

$$\hat{X}_t = \mathbb{E}\left[X_t | Y_0, \ldots, Y_t\right] \tag{2.14}$$

$$\bar{P}_t = \mathbb{E}\left[\left(X_t - \bar{X}_t\right)\left(X_t - \bar{X}_t\right)^T\right] \tag{2.15}$$

$$\hat{P}_t = \mathbb{E}\left[\left(X_t - \hat{X}_t\right)\left(X_t - \hat{X}_t\right)^T\right] \tag{2.16}$$

The original Kalman gives the four unknown analytically. The given estimator is unbiased and of minimum variance. With the success of the original Kalman filter, the temptation was strong to extend it to non-linear cases. Extended Kalman filter, unscented Kalman filter and ensemble Kalman filter export the orginal technique to the non-linear cases with gradual

complexity.

2.2.1 Original Kalman filter

The equations (**??**) and (**??**) are written in the following form when the functions f_t and h_t are linear and on the errors ϵ_t^X and ϵ_t^Y are Gaussian and independent:

$$\left\{ \begin{array}{rcl} X_t &=& F_t X_{t-1} + \epsilon_t^X \\ Y_t &=& H_t X_t + \epsilon_t^Y \end{array} \right. \tag{2.17}$$

with $F_t \in \mathbb{R}^{d \times d}$, $H_t \in \mathbb{R}^{m \times d}$, ϵ_t^X and ϵ_t^Y respectively d and m-dimensional centred Gaussian vectors.

The solution is given by the system (2.18). The estimator $\hat{X}_t$ is optimal: it is unbiased and has a variance equal to the Cramer-Rao bound (Arulampalam et al., 2002; Le Gland, 2009; Welch and Bishop, 1995). It is a compromise of the information brought by the model ($\bar{X}_t$) and the information brought by the observation ($Y_t - H_t\bar{X}_t$) pondered by their uncertainty (repectively B_t and R_t). If the observation is perfect ($R_t = 0$), then $\hat{X}_t = H_t^T Y_t$. The model is dismissed, only the observation is kept. If the model is perfect ($B_t = 0$), then $\hat{X}_t = \bar{X}_t$. The observation is not used, only the model is trusted.

$$\left\{ \begin{array}{rcl} \textit{Prediction :} & & \\ \bar{X}_t &=& F_t \hat{X}_{t-1} \\ \bar{P}_t &=& F_t \hat{P}_{t-1} F_t^T + B_t \\ \textit{Correction :} & & \\ K_t &=& \bar{P}_t H_t^T \left(H_t \bar{P}_t H_t^T + R_t \right)^{-1} \\ \hat{X}_t &=& \bar{X}_t + K_t \left(Y_t - H_t \bar{X}_t \right) \\ \hat{P}_t &=& \bar{P}_t - K_t H_t \bar{P}_t \end{array} \right. \tag{2.18}$$

The expression of (2.18) can be found by maximizing the likelihood by a variational approach. Both solution are equivalent. They can be found in Le Gland (2009); Welch and Bishop (1995). The variational approach clearly sustains the interpretation of $\hat{X}_t$ as a compromise between observation and model.

$$\bar{X}_t = \arg\min_X \left\{ \frac{1}{2} \| X - F_t \hat{X}_{t-1} \|_{B_t}^2 \right\} \tag{2.19}$$

$$\hat{X}_t = \arg\min_X \left\{ \frac{1}{2} \| X - F_t \hat{X}_{t-1} \|_{B_t}^2 + \frac{1}{2} \| Y_t - H_t X \|_{R_t}^2 \right\} \tag{2.20}$$

2.2.2 Extended Kalman filter

The extended Kalman filter (EKF) exports the original Kalman filter to the non-linear case almost directly. The system (2.17) is complexioned into the sytem (2.21):

$$\left\{ \begin{array}{rcl} X_t & = & f_t(X_{t-1}, \epsilon_t^X) \\ Y_t & = & h_t(X_t, \epsilon_t^Y) \end{array} \right. \tag{2.21}$$

with the non-linear functions f_t and h_t:

$$f_t : \begin{array}{ccc} \mathbb{R}^d \times \mathbb{R}^d & \longrightarrow & \mathbb{R}^d \\ (X, \epsilon) & \longmapsto & f_t(X, \epsilon) \end{array} \quad \text{and} \quad h_t : \begin{array}{ccc} \mathbb{R}^d \times \mathbb{R}^m & \longrightarrow & \mathbb{R}^m \\ (X, \epsilon) & \longmapsto & h_t(X, \epsilon) \end{array}$$

The functions f_t and h_t are assumed differentiable, are their Taylor expansion at order 1 is used as a linear approximation of them.

$$f_t(X_t, \epsilon_t^X) = f_t(\hat{X}_t, 0) + F_t\left(X_t - \hat{X}_t\right) + C_t\epsilon_t^X + \mathcal{O}\left(\|(X_t - \hat{X}_t, \epsilon_t^X)\|^2_{\mathbb{R}^d \times \mathbb{R}^d}\right)$$

$$h_t(X_t, \epsilon_t^Y) = h_t(\bar{X}_t, 0) + H_t\left(X_t - \bar{X}_t\right) + D_t\epsilon_t^Y + \mathcal{O}\left(\|(X_t - \bar{X}_t, \epsilon_t^Y)\|^2_{\mathbb{R}^d \times \mathbb{R}^m}\right)$$

with the matrices

$$F_t = \left(\frac{\partial f_t^{(i)}}{\partial X^{(j)}}(\hat{X}_t, 0)\right)_{1 \leqslant i,j \leqslant d} \quad C_t = \left(\frac{\partial f_t^{(i)}}{\partial \epsilon^{(j)}}(\hat{X}_{t-1}, 0)\right)_{1 \leqslant i,j \leqslant d}$$

$$H_t = \left(\frac{\partial h_t^{(i)}}{\partial X^{(j)}}(\bar{X}_t, 0)\right)_{\substack{1 \leqslant j \leqslant d \\ 1 \leqslant i \leqslant m}} \quad D_t = \left(\frac{\partial h_t^{(i)}}{\partial \epsilon^{(j)}}(\bar{X}_t, 0)\right)_{1 \leqslant i,j \leqslant m}$$

With this linearisation, the same computation can be done. The ensuing estimator is the EKF, given by equations (2.22). One can see that it is not very different from the original Kalman filter. The only difference is the direct propagation from $\hat{X}_{t-1}$ with no noise to get $\hat{X}_t$ and the use of Jacobian matrices instead of given F_t and H_t. The apparition of Jacobian dreadfully increases the computational cost because they must be calculated and inverted at each time step. The proposed estimator $\hat{X}_t$ is no longer optimal. Moreover, the filter is proven to be unstable when the noise is high or when it is badly initialized (Reif et al., 1999). Conditions to have a bounded error include small noise, good starting point, and non-linearities not continuous.

$$\left\{\begin{array}{rcl} \textit{Prévision :} & & \\ \bar{X}_t & = & f_t(\hat{X}_{t-1}, 0) \\ \bar{P}_t & = & F_t \hat{P}_{t-1} F_t^T + C_t B_t C_t^T \\ \textit{Correction :} & & \\ K_t & = & \bar{P}_t H_t^T \left(H_t \bar{P}_t H_t^T + D_t R_t D_t^T \right)^{-1} \\ \hat{X}_t & = & \bar{X}_t + K_t \left(Y_t - h_t(\bar{X}_t, 0) \right) \\ \hat{P}_t & = & \bar{P}_t - K_t H_t \bar{P}_t \end{array}\right. \tag{2.22}$$

2.2.3 Unscented Kalman filter

So-called *unscented Kalman filter* has been invented by Julier and Uhlmann (1997, 2004). Its idea is to approximate the distribution of X_t rather than the non-linear functions f_t and h_t. The distribution of X_t is approximated using a sample of $2d+1$ points. The sampling point (so-called *sigma-points*) are deterministically chosen according to the method described by Julier and Uhlmann (1997): the unscented transform.

The principle of the unscented transform is to approximate the two first moments with two points per dimension plus the average (for a real-valued random variable X, such points would be $\mathbb{E}[X]$, $\mathbb{E}[X] + \sigma_X$ and $\mathbb{E}[X] - \sigma_X$). Since the average is common for all dimension, the unscented transform ends up with $2d+1$ points $\{\mathbf{X}^0, ..., \mathbf{X}^{2d+1}\}$ gathered into a matrix $\mathbf{X} \in \mathbb{R}^{2d+1 \times d}$ (each line of $\mathbf{X}$ is a state vector). For a d-dimensional random variable X, its covariance matrix $\hat{P} = cov(X, X)$ is positive-define and symmetric. Thus there exists $Q \in \mathbb{R}^{d \times d}$ such that $\hat{P} = Q^T Q$ (Q is the "square-root" of $\hat{P}$). For any $\kappa \in \mathbb{R}$, the sigma-points are given by:

$$\begin{array}{rcl} \mathbf{X}^0 & = & \hat{X} \\ \mathbf{X}^i & = & \hat{X} + \sqrt{d+\kappa} Q_{\bullet,i}\,, \quad \forall i \in \{1, ..., d\} \\ \mathbf{X}^{i+d} & = & \hat{X} - \sqrt{d+\kappa} Q_{\bullet,i} \end{array}$$

with each a weight $w \in \mathbb{R}^{2d+1}$ such that $\mathbf{w} = (\frac{\kappa}{d+\kappa}, \frac{1}{2(d+\kappa)}, ..., \frac{1}{2(d+\kappa)})^T$

After the unscented transform, the prediction step is done by apply the non-linear transform to the sigma points. The mean and covariance are computed directly from the predicted sigma points, according to their weight.

$$\bar{X}_t = \sum_{i=0}^{2d} \mathbf{w}^i \mathbf{X}^i = \mathbf{X}^T \mathbf{w}$$

$$\bar{P}_t = \sum_{i=0}^{2d} \mathbf{w}^i \left(\mathbf{X}^i - \bar{X}_t \right) \left(\mathbf{X}^i - \bar{X}_t \right)^T$$

The correction step assimilates the observation in the observation space. The observation operator is applied to the sigma points, without noise: $\mathbf{Y} = h_t(\mathbf{X}, 0)$. The given observation

Y_t is compared to the average of such ensemble.

$$\hat{X}_t = \bar{X}_t + K_t(Y_t - \mathbf{Y}^T\mathbf{w})$$

$$\hat{P}_t = \bar{P}_t - K_t P^{yy} K_t^T$$

with the covariance matrix of the ensemble in the observation space $P^{yy} = \sum_{i=0}^{2d} \mathbf{w}^i(\mathbf{Y}^i - \mathbf{Y}^T\mathbf{w})(\mathbf{Y} - \mathbf{Y}^T\mathbf{w})^T$ and the gain matrix $K_t = P^{xy}(P^{yy})^{-1}$, where $P^{yy} = \sum_{i=0}^{2d} \mathbf{w}^i(\mathbf{X}^i - \bar{X}_t)(\mathbf{Y}^i - \mathbf{Y}^T\mathbf{w})^T$ is the likelihood.

After the correction, the unscented transform create a new sample with the corrected mean and covariance. This new sample replace the previous, and the cycle starts again. In summary, the algorithm of the UKF consists in the following steps:

1. At $t = 0$, do the unscented transform with the initial mean and covariance: $\mathbf{X} = UT(x_0, P_0)$
2. Propagate the sigma points to the next time step with the non-linear dynamic: $\mathbf{X}_t = f_t(\mathbf{X}_{t-1}, 0)$.
3. Compute the prediction statistics: $\bar{X}_t = \mathbf{X}^T\mathbf{w}$, $\bar{P}_t = \sum_{i=0}^{2d} w^i \left(\mathbf{X}^i - \bar{X}_t\right)\left(\mathbf{X}^i - \bar{X}_t\right)^T$
4. Apply the observation operator to the sigma points: $\mathbf{Y} = h_t(\mathbf{X}, 0)$
5. Compute the corrected statistics: $\hat{X}_t = \bar{X}_t + K_t(Y_t - \mathbf{Y}^T\mathbf{w})$, $\hat{P}_t = \bar{P}_t - K_t P^{yy} K_t^T$
6. Do the unscented transform with the corrected statistics: $\mathbf{X} = UT(\hat{X}_t, \hat{P}_t)$
7. Repeat from step 2 until the end of the time series.

The main advantage of the UKF over the EKF is to avoid degeneration. The estimation are stable, and the estimation of covariances matrices are widely improved, as shown in (Julier and Uhlmann, 2004). In high dimension, the unscented transform suffers from non-local sampling. The UKF is limited to the two first momenta. It is a barrier for strongly non-Gaussian distributions. Still, the UKF is very popular filter, because it avoids costly Monte Carlo method and provides good results. Many application can be found, as well as improvement to shrink its drawbacks (Julier, 2002; Gustafsson and Hendeby, 2012).

2.2.4 Ensemble Kalman filter

The ensemble Kalman filter (EnKF) is based on the same idea as the UKF: it is better to approximate the probability law than the non-linear dynamic. Instead of using the unscented transform to approximate the law, the EnKF uses a Monte Carlo method. Since its creation by Evensen (1994), the EnKF encountered a large success in the Earth science community. Evensen (2003) relates the many works on the ensemble Kalman filter that have been carried out before 2003. There is no mention to the unscented Kalman filter because it can only be

used on small problems. As Evensen is focused in geosciences, the unscented transform has a prohibitive cost and thus not considered in the paper.

The ensemble Kalman filter presented here comes from Burgers et al. (1998). It assumes the observation operator is linear: $Y_t = h_t(X_t, \epsilon_t^Y) = H_t X_t + \epsilon_t^Y$ with $H_t \in \mathbb{R}^{m \times d}$ and $\epsilon_t^Y \rightsquigarrow \mathcal{N}(0, R_t)$. As for UKF, the distribution of X_t is approximated using a sample, but a random sample of N points. This sample $\mathbf{X}_t \in \mathbb{R}^{N \times d}$ is called an *ensemble*. A single state $X^i \in \mathbb{R}^d$ of the ensemble is called a *member*. Every member has the same weight in the computation of the system's statistics. Each member is propagated independently from the time t to $t+1$ with the non-linear dynamic f_t. The *a priori* statistics are computed as follows:

$$\bar{X}_t = \frac{1}{N} \sum_{i=0}^{N} \mathbf{X}_t^i$$

$$\bar{P}_t = \frac{1}{N} \sum_{i=0}^{N} \left(\mathbf{X}_t^i - \bar{X}_t\right) \left(\mathbf{X}_t^i - \bar{X}_t\right)^T$$

For the correction step, an ensemble of observations is created. That is to say a centred Gaussian noise with variance R_t is added to the provided observation Y_t: $\mathbf{Y}^i = Y_t + \epsilon^i$. The corrected ensemble (also called *analysis*) is the one obtained from the assimilation of one member in one member of prediction: $\widehat{\mathbf{X}}_t^i = \mathbf{X}^i + K_t(\mathbf{Y}^i - H_t \mathbf{X}^i)$. The gain matrix is the same as for the original Kalman filter: $K_t = \bar{P}_t H_t^T \left(H_t \bar{P}_t H_t^T + R_t\right)^{-1}$. From that *a posteriori* ensemble, the corrected statistics are computed:

$$\hat{X}_t = \frac{1}{N} \sum_{i=0}^{N} \widehat{\mathbf{X}}_t^i$$

$$\hat{P}_t = \frac{1}{N} \sum_{i=0}^{N} \left(\widehat{\mathbf{X}}_t^i - \hat{X}_t\right) \left(\widehat{\mathbf{X}}_t^i - \hat{X}_t\right)^T$$

After the correction step, the *a posteriori* ensemble is used to build the next *a priori* ensemble (by moving each member according to the non-linear dynamic). The ensemble is not reset and thus some features due to dynamics can appear. It makes the sample more representative of the system. In summary, the ensemble Kalman filter is defined by the

following system:

$$\left\{\begin{array}{rcl} \textit{Prediction :} & & \\ \mathbf{X}_t^i &=& f_t(\widehat{\mathbf{X}}_{t-1}^i, 0) \\ \bar{X}_t &=& \frac{1}{N}\sum_{i=0}^N \mathbf{X}_t^i \\ \bar{P}_t &=& \frac{1}{N}\sum_{i=0}^N \left(\mathbf{X}_t^i - \bar{X}_t\right)\left(\mathbf{X}_t^i - \bar{X}_t\right)^T \\ \textit{Correction :} & & \\ K_t &=& \bar{P}_t H_t^T \left(H_t \bar{P}_t H_t^T + R_t\right)^{-1} \\ \widehat{\mathbf{X}}_t^i &=& \mathbf{X}^i + K_t(Y_t + \epsilon_Y^i - H_t\mathbf{X}^i) \\ \hat{X}_t &=& \frac{1}{N}\sum_{i=0}^N \widehat{\mathbf{X}}_t^i \\ \hat{P}_t &=& \frac{1}{N}\sum_{i=0}^N \left(\widehat{\mathbf{X}}_t^i - \hat{X}_t\right)\left(\widehat{\mathbf{X}}_t^i - \hat{X}_t\right)^T \end{array}\right.$$

The target distribution $\hat{\eta}_t = \mathbb{P}\left(X_t | Y_{0:t}\right)$ is approached by the corrected ensemble $\widehat{\mathbf{X}}$. When the size of the ensemble N goes to infinity, it is expected that the approximation converges toward the target. Le Gland et al. (2009) studied the asymptotic probability law of the EnKF. They show the ensemble Kalman filter converges. The asymptotic law is the Bayesian filter $\hat{\eta}_t$ when the dynamic is linear and the noise are Gaussian (EnKF converges toward KF). But when the dynamic is non-linear, it is shown that EnKF converges toward the wrong limit. Nevertheless, Mandel et al. (2011) shows the L^p convergence of the 2 first moments toward the good limit, which is sufficient for most of application. Baehr et al. (2010) recommends to couple a large scale EnKF with a local particle filter for numerical weather prediction applications.

To conclude this introduction to linear and non-linear filtering with the Kalman family of filters, the table 2.1 recaps the main idea for each Kalman filter. This introduction did not meant to be exhaustive. Even restricted to one area of application, the popularity of Kalman filter is such that it become hard to be exhaustive. The 328 references given by Mahmoud and Khalid (2013) for distributed systems give a taste of it. Original Kalman filter is optimal for linear and Gaussian filtering. Extended Kalman filter can be useful for poorly non-linear dynamics, but suffers from instability and high computational cost. Unscented Kalman filter uses $2d+1$ points to estimate mean and covariance (unscented transform) with less instability and not higher computational cost. Ensemble Kalman filter uses N points to estimate mean and covariance (Monte Carlo) and adds noise to the observations. Although EnKF is the most adapted to non-linear dynamics, it does not fit for strongly non-Gaussian posterior.

Name of the filter	Short name	Description	Avantages	Shortcomings
Classical Kalman	KF	Solution of the filtering problem in the linear and Gaussian case	Optimal	Needs linearity and Gaussian independent noise
Extended Kalman	EKF	KF on the increments	Simple extension of KF to the non-linear cases	Unstable when high noise or discontinuous non-linearity, high computational cost
Kalman "un-scented"	UKF	Approximation of the law with unscented transform	Better representation of variances, suffer less from non-linearity	Only 2 first momenta are known.
Kalman d'ensemble	EnKF	Approximation of the law with Monte Carlo	Suitable to very high dimensional problems	Does not converge toward the optimal estimator

Table 2.1 – Short list of Kalman filters with pros and cons

2.3 Particles filters

2.3.1 The Bayesian filter and some admissible selection kernels

At the begin of the chapter, the filtering problem has been introduced and the two major steps (mutation and selection) have been presented. It has been said that the mutation step is more the concern of modelling than of filtering. The concern of filtering is the selection step. We recall here the equation (2.12) which define a selection kernel.

Definition 2.2 (Selection kernel)**.**

With words *A selection kernel is a stochastic kernel which gives, for any a priori state, the a posteriori probability law.*

Formally *S_t is a selection kernel if it verifies*

$$\hat{\eta}_t(d\hat{x}) = \int \eta_t(d\bar{x}) S_t(\bar{x}, d\hat{x})$$

it depends on $\bar{x}$ and $\hat{x}$, but also on the whole a priori probability law, in which case it is denoted S_{t,η_t}.

The Bayesian filter is the filter which can access for any function ϕ the distribution η_t and $\hat{\eta}_t$. Although they have been presented at the beginning of the chapter, we did not express

what Bayes' theorem says from them because it was not necessary to introduce Kalman filters. Bayes' theorem for the event $A = \{X_t \in dx | Y_{0:t-1} = y_{0:t-1}\}$ and $B = \{Y_t \in dy_t\}$ gives

$$\mathbb{P}\left(X_t \in dx | Y_{0:t} = y_{0:t}\right) = \frac{\mathbb{P}\left(Y_t \in dy_t | X_t = x, Y_{0:t-1} = y_{0:t-1}\right) \mathbb{P}\left(X_t \in dx | Y_{0:t-1} = y_{0:t-1}\right)}{\mathbb{P}\left(Y_t \in dy_t\right)} \tag{2.23}$$

On the left hand side, one can recognize the a posteriori distribution $\hat{\eta}_t(d\hat{x})$. On the right hand side, one of the term at the numerator is the a priori distribution $\eta_t(d\bar{x})$. The remaining terms can be expressed thanks to the assumption that $(Y_t)_t$ is a memoryless channel. In particular, $\mathbb{P}\left(Y_t \in dy_t | X_t = x\right) = g_t(x, y_t)\Lambda(dy_t)$ with the measure Λ which does not depend on X_t. At the denominator, the formula of total probability is used once more and the term $q_t(dy_t)$ disappears:

$$\frac{\mathbb{P}\left(Y_t \in dy_t | X_t = x, Y_{0:t-1} = y_{0:t-1}\right)}{\mathbb{P}\left(Y_t \in dy_t\right)} = \frac{g_t(x, y_t)\cancel{\Lambda(dy_t)}}{\int g_t(x, y_t)\cancel{\Lambda(dy_t)}\mathbb{P}\left(X_t \in dx\right)} = \frac{g_t(x, y_t)}{\int g_t(x, y_t)\eta_t(dx)}$$

Finally, Bayes' theorem expresses the a posteriori distribution in function of the a priori distribution and the likelihood. For now, the notation of the likelihood as a function of the a priori state is in use $G_t(x) := g_t(x, y_t)$ (the observation y_t is implied) as well as the distribution notation $\eta_t(\phi) = \int \phi(x)\eta_t(dx)$ for any function ϕ. The equation 2.23 can be rewritten

$$\hat{\eta}_t(dx) = \frac{G_t(x)}{\eta_t(G_t)}\eta_t(dx) \tag{2.24}$$

The equation 2.24 is the starting point to define several selection kernels. Indeed, any kernel satisfying the relation (2.12) can be used as selection kernel. Two candidates are shown to match the definition 2.2, but many others exists (Del Moral, 2004; Baehr, 2008).

Proposition 2.1. *The following kernels are selections kernels.*

$$S^1_{t,\eta_t}(\bar{x}, d\hat{x}) = \frac{G_t(\hat{x})}{\eta_t(G_t)}\eta_t(d\hat{x})$$

$$S^2_{t,\eta_t}(\bar{x}, d\hat{x}) = G_t(\bar{x})\delta_{\bar{x}}(d\hat{x}) + \left[1 - G_t(\bar{x})\right]\frac{G_t(\hat{x})}{\eta_t(G_t)}\eta_t(d\hat{x})$$

Proof. Both match the definition 2.1 of a stochastic kernel. The kernel S^1_{t,η_t} is a selection kernel:

$$\begin{aligned}
\int_{\bar{x}} \eta_t(d\bar{x}) S^1_{t,\eta_t}(\bar{x}, d\hat{x}) &= \int_{\bar{x}} \eta_t(d\bar{x}) \frac{G_t(\hat{x})}{\eta_t(G_t)}\eta_t(d\hat{x}) \\
&= \underbrace{\frac{G_t(\hat{x})}{\eta_t(G_t)}\eta_t(d\hat{x})}_{\hat{\eta}_t} \underbrace{\int_{\bar{x}} \eta_t(d\bar{x})}_{=1} \\
&= \hat{\eta}_t(d\hat{x})
\end{aligned}$$

Idem for the kernel S^2_{t,η_t}:

$$\begin{aligned}
\int_{\bar{x}} \eta_t(d\bar{x}) S^2_{t,\eta_t}(\bar{x}, d\hat{x}) &= \int_{\bar{x}} \eta_t(d\bar{x}) \left(G_t(\bar{x})\delta_{\bar{x}}(d\hat{x}) + [1 - G_t(\bar{x})] \frac{G_t(\hat{x})}{\eta_t(G_t)} \eta_t(d\hat{x}) \right) \\
&= \int_{\bar{x}} \eta_t(d\bar{x}) G_t(\bar{x})\delta_{\bar{x}}(d\hat{x}) + \frac{G_t(\hat{x})}{\eta_t(G_t)} \eta_t(d\hat{x}) \int_{\bar{x}} \eta_t(d\bar{x}) \left[1 - G_t(\bar{x})\right] \\
&= \hat{\eta}_t(d\hat{x}) \eta_t(G_t) + \hat{\eta}_t(d\hat{x}) \left[1 - \eta_t(G_t)\right] \\
&= \hat{\eta}_t(d\hat{x})
\end{aligned}$$

□

To interpret the differences between these kernels, we recall the expression (2.10) which holds for any of them:

$$S_{t,\eta_t}(\bar{x}, d\hat{x}) = \mathbb{P}\left(\hat{X}_t \in d\hat{x} \,|\bar{X}_t = \bar{x} \right)$$

with $\hat{X}_t \overset{\mathcal{L}}{=} X_t|Y_{0:t}$ and $\bar{X}_t \overset{\mathcal{L}}{=} X_t|Y_{0:t-1}$ and $\overset{\mathcal{L}}{=}$ standing for the equality in law.

This relation tells the general interpretation of selection kernels: given a position $\bar{x}$ of the system before it is observed, $S_{t,\eta_t}(\bar{x}, d\hat{x})$ is the probability that the system after is around $\hat{x}$ after a new observation is assimilated.

For the first kernel S^1_{t,η_t}, this probability does not depend on $\bar{x}$, but it does depend on the whole a priori distribution η_t. It means that all *a priori* state $\bar{x}$ will give the same *a posteriori* probability law. Nevertheless, the gathering of all *a priori* states has an influence on the *a posteriori* probability law. That is to say, the a priori state $\bar{x}$ has an impact on the *a posteriori* probability law as a part of the *a priori* distribution. This is a so-called *mean-field* process. For a posterior state $\hat{x}$, the higher the *a priori* probability $\eta_t(d\hat{x})$, the higher the *a posteriori* probability $\hat{\eta}_t(d\hat{x})$. Moreover, the transition probability depends on the arrival state through the term $G_t(\hat{x})$. It tells the higher is the likelihood of a posterior, the higher is its probability. Overall, the selection kernel S^1_{t,η_t} selects the posterior states both close the observation and with good *a priori* probability.

For the second kernel S^2_{t,η_t}, the transition probability does depend on the *a priori* state $\bar{x}$. The term $G_t(\bar{x})\delta_{\bar{x}}(d\hat{x})$ indicates that prior states with a high likelihood will have a larger chance to stay at the point $\bar{x}$. Conversely, the prior states with a small likelihood will have different posterior states, given according to the same rule as the kernel S^1_{t,η_t}. That is to say, with kernel 2, the prior the closest the observation are kept where they are and the others have a posterior given by the comprise between likelihood and the *a priori* probability, like with kernel 1.

These two kernels lead to different algorithm, as we will see at the end of this chapter: the kernel 1 is used in the SIS and SIR algorithms, the kernel 2 is used in the genetic selection algorithm. By default, what the expression "particle filter" stands for is the filter with the selection kernel S^1_{t,η_t}. But any other kernel can usually be used instead and lead to the same comments.

2.3.2 The particle approximation

At chapter A, we have seen that random variables and stochastic processes are manipulated in practice with samples. Unless the theory gives directly the quantities of interest[2], the estimation are made on a sample. The target distributions are approached with a Monte Carlo method. Each element of a sample is called a *particle*.

$$\eta_t^N = \frac{1}{N}\sum_{i=1}^{N} \delta_{X_t^i} \tag{2.25}$$

$$\hat{\eta}_t^N = \frac{1}{N}\sum_{i=1}^{N} \delta_{\hat{X}_t^i} \tag{2.26}$$

The particle approximation have already been made for the ensemble Kalman filter and, in a certain extend[3], for the unscented Kalman filter. It is unavoidable for particle filters. The mutation and selection kernel should be rewritten in the particle approximation. But all the interpretation can be done equally out of the particle approximation. The particle approximation is thus only presented to better describe the algorithms.

The a posteriori distribution $\hat{\eta}_t(\phi) = \int \phi(x)\hat{\eta}_t(dx)$ is approached by

$$\widehat{\eta}_t^N(\phi) = \sum_{i=1}^{N} w^i \phi(x^i) \tag{2.27}$$

with the weight associated to each particle:

$$w^i = \mathbb{P}\left(X_t^i \in dx^i | Y_{0:t}\right) = \frac{G_t(x^i)\eta_t(dx^i)}{\sum_{i=1}^N G_t(x^i)\eta_t(dx^i)} \tag{2.28}$$

The weight samples the importance distribution. It is given by Bayes' theorem. Using these weight, the selection kernel S_{t,η_t}^1 is written

$$S_{t,\eta_t}^{1,N}(\bar{x}^i, d\hat{x}^i) = \mathbb{P}\left(\hat{X}_t^i \in d\hat{x}^i | \bar{X}_t^i = \bar{x}^i\right) = w^i$$

A particle at the point $\bar{x}^i$ will be at $\hat{x}^i$ after the correction with a probability w^i. The second selection kernel is written as such

$$S_{t,\eta_t}^{2,N}(\bar{x}^i, d\hat{x}^i) = \mathbb{P}\left(\hat{X}_t^i \in d\hat{x}^i | \bar{X}_t^i = \bar{x}^i\right) = G_t(\bar{x}^i)\delta_{\bar{x}^i} + (1 - G_t(\bar{x}^i))w^i$$

[2]It is the case for the linear and Gaussian case: the theory tells us the target laws are Gaussian and gives a way to estimate recursively mean and covariance with the original Kalman filter. There is no need of a sample to solve the filtering problem.

[3]The unscented transform gives a deterministic choice of $2d+1$ points which are used to compute mean and covariance. This approach is different than a full Monte Carlo sample with statistically-based manipulations.

A particle at the point $\bar{x}^i$ has a probability $G_t(\bar{x}^i)$ to stay at $\bar{x}^i$ and a probability $(1 - G_t(\bar{x}^i))$ to move to a new point $\hat{x}^i$ which has a large weight w^i.

2.3.3 Three particles filter algorithms

These three algorithms are described in (Arulampalam et al., 2002; Maskell and Gordon, 2001; Doucet et al., 2000; van Leeuwen, 2009).

2.3.3.1 Sequential importance sampling (SIS)

In sequential importance sampling, particles are not modified by the selection kernel. Only their weight for the estimation is changed. The weight is given according to the likelihood Hence the posterior take the prior information from the position of particles and the likelihood information from their weight. The algorithm is illustred by figure 2.6. The corresponding kernel for this algorithm is

$$S^1_{t,\eta_t}(\bar{x}, d\hat{x}) = \frac{G_t(\hat{x})}{\eta_t(G_t)}\eta_t(d\hat{x})$$

This algorithm is known to degenerate (Snyder et al., 2008; Bengtsson et al., 2008; Del Moral, 2004). The particles spread away with time because of model error while the observation stays close to the real signal. It results smaller and smaller weights, which degrades the posterior estimation because only few particles remain relevant.

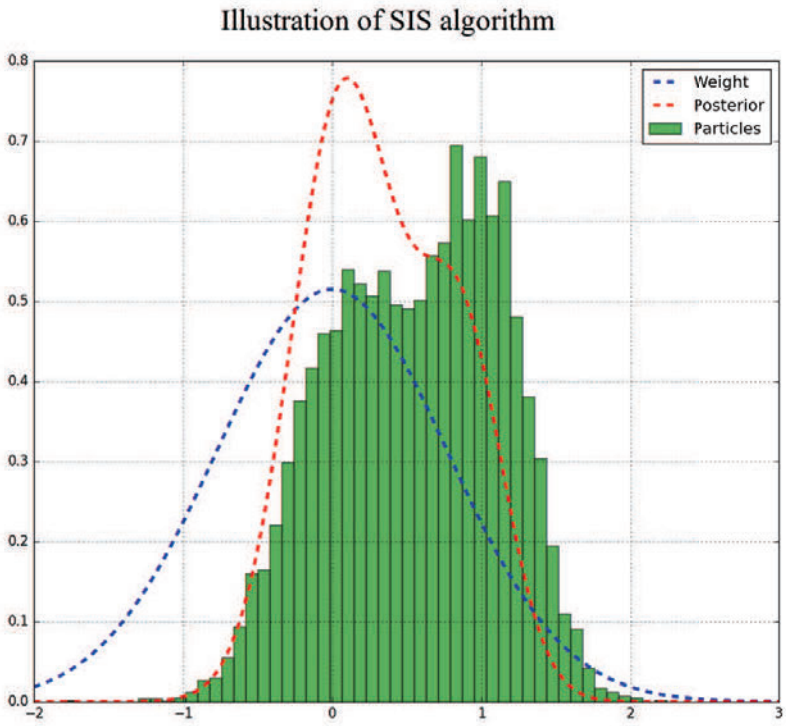


Figure 2.5 – SIS algorithm illustration. The particles a priori are represented by their histogram (in green). Their weight, given by the likelihood, is the blue dashed curve. The resulting posterior is drawn in red.

2.3.3.2 Sampling with importance resampling (SIR)

Particles are modified by the selection kernel: they are all resampled according to their likelihood. After the resampling, all particle have the same weight. Hence the distribution of particles is truly the posterior distribution. The algorithm is illustrated by figure 2.6. The corresponding kernel for this algorithm is also

$$S^1_{t,\eta_t}(\bar{x}, d\hat{x}) = \frac{G_t(\hat{x})}{\eta_t(G_t)}\eta_t(d\hat{x})$$

The resampling avoid degeneracy which can occur with the SIS algorithm (Doucet et al., 2000; van Leeuwen, 2009). However, with SIR, particles have a trend to gather on few values. Even with a good discrepancy of the initial population of particles, the discrepancy collapses. Various tricks algorithms exist to sustain variability among particles but avoid degeneracy (Del Moral, 2004; Doucet et al., 2000; van Leeuwen, 2009). We will only focus on one them: the genetic selection, which is in use for the non-linear filtering of Doppler lidar we present.

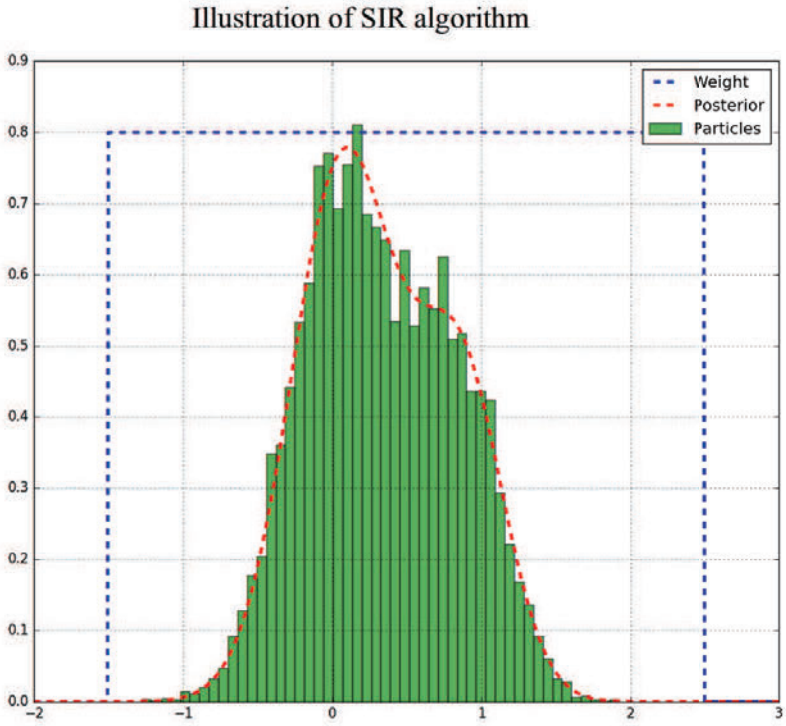


Figure 2.6 – SIR algorithm illustration. The particles a priori are represented by their histogram (in green). Their weight, given by the likelihood, is the blue dashed curve. The resulting posterior is drawn in red.

2.3.3.3 Genetic selection

The genetic algorithm include an acceptance/rejection step: particles selected are not modified, particles rejected are resampled in the same way as in SIR. The selection kernel associated to this algorithm is

$$S^2_{t,\eta_t}(\bar{x}, d\hat{x}) = G_t(\bar{x})\delta_{\bar{x}}(d\hat{x}) + [1 - G_t(\bar{x})]\,\frac{G_t(\hat{x})}{\eta_t(G_t)}\eta_t(d\hat{x})$$

The selection kernel must be read as follows. The first term has for factor $\delta_{\bar{x}}(d\hat{x})$: the arriving probability is non-zero only in $\bar{x}$. The second term has for factor $\frac{G_t(\hat{x})}{\eta_t(G_t)}\eta_t(d\hat{x})$: exactly the kernel $S^1_{t,\eta_t}(\bar{x}, d\hat{x})$, which resamples particles independently from where they come from. The coefficient of these factors are respectively $G_t(\bar{x})$ and $1 - G_t(\bar{x})$, which are likelihood and anti-likelihood of the particle at the departure point. Hence, this selection kernel denotes the following operations: first, the particles are selected randomly with a probability proportional to their likelihood at the departure point. The selected particles stay where they are. The rejected particles are resampled like with SIR.

The advantage of this method is to keep the benefit of a good prediction. It also reduces the computing cost due to resampling. The number of rejected particles is also a simple tracer of degeneracy: if particles are taking unrealistic values, they will be rejected. To give an order of magnitude, when the reconstruction method is performing well, the proportion of rejection lies between 15% and 55%, with an average value around 30%

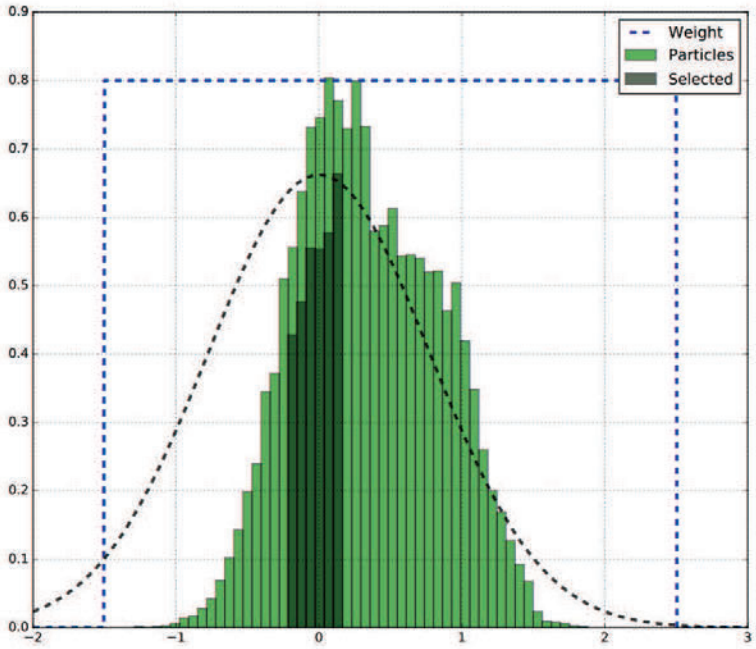


Figure 2.7 – Illustration of the genetic selection algorithm. The process is the same as SIR, but it is preceded by an acceptance/rejection step. The accepted particles are represented in grey. They are selected according to their likelihood (grey bell curve).

2.3.4 Particles filter and ensemble Kalman filter

At the first sight, particle filters and ensemble Kalman filter look similar. Indeed, they both aim to perform non-linear filtering and the Monte Carlo approximation of the distribution is a common feature. Although, they differ on two points:

- In EnKF, noise is added to observations, not in particle filters.
- In particle filters, members have different weights (or are resampled), not in EnKF.

These two main differences come from two independent approaches: EnKF coming from the original KF and keeping its "linear aspect" and particle filter coming from the direct use of Bayes' formula (Pham, 2001). Le Gland et al. (2009) show there is no difference in the

mutation step, but only in the selection step. They highlight the difference as such: the *a posteriori* distributions obtained with the ensemble Kalman filter is denoted $\widehat{\eta}^{KF}(\phi)$ and the one obtained with the Bayes' formula is denoted $\widehat{\eta}^{BF}(\phi)$. They are based on a common *a priori* distribution η_t. The Kalman filtered distribution is given by (2.29) for any function ϕ.

$$\widehat{\eta}_t^{KF}(\phi) = \int\int \phi(x + K_t(y_t - H_t x - v))q_t(v)dv\eta_t(dx) \tag{2.29}$$

with $K_t = \bar{P}_t H_t^T \left(H_t \bar{P}_t H_t^T + R_t\right)^{-1}$ the gain matrix and q_t the PDF of the added error. Most of the time, the error are assumed to be Gaussian: $q_t(v) = (2\pi)^{-1/2} e^{-v^2/2}$.

The Bayesian filtered distribution is given by (2.30).

$$\widehat{\eta}_t^{BF}(\phi) = \frac{\int \phi(x) q_t(y_t - H_t x)\eta_t(dx)}{\int q_t(y_t - H_t x)\eta_t(dx)} \tag{2.30}$$

with the likelihood $G_t(x) = q_t(y_t - H_t x)$ in equation (2.24).

When restricted to the 1-dimensional case, the definition of a the selection kernel and the change of variables $(\bar{x}, \hat{x}) \to (x, v)$, it yields to the following kernels:

$$S_{t,\eta_t}^{BF}(\bar{x}, \hat{x}) = \frac{q_t(y_t - H_t \hat{x})}{\int q_t(y_t - H_t x)\eta_t(dx)}\eta_t(d\hat{x}) \tag{2.31}$$

$$S_t^{KF}(\bar{x}, \hat{x}) = q_t\left(y_t - K_t^{-1}(\hat{x} - (1 - K_t H_t)\bar{x})\right) d\hat{x} \tag{2.32}$$

Le Gland et al. (2009) argue that the two distributions $\widehat{\eta}_t^{BF}$ and $\widehat{\eta}_t^{KF}$ are equal only when the dynamic is linear. Out of this case, the EnKF does not converge to the Bayesian filter. And indeed, the interpretation of stochastic kernels points out several differences.

The kernel S_t^{KF} does not depend on the full probability law η_t (which is no longer in subscript) and depends on the *a priori* state $\bar{x}$. It tells that the selection step is done independently among particles. As one can see in the equations of EnKF, the corrected value for a member depends only the current state of this member. The other members does not influence the correction. Conversely, in the kernel S_{t,η_t}^{BF}, one can see that the full probability law η_t appears and the *a priori* state $\bar{x}$ does not.

2.4 Conclusion

The filtering problem is to find *a priori* and *a posteriori* estimates of the system, given a model and set of observations with known variances. Bayesian filters have two steps: the mutation and the selection. They are described by stochastic kernels (Le Gland, 2009).

Kalman filter gives a solution of the filtering problem when the dynamic is linear and the noises are Gaussian. The estimator provided by the original Kalman filter is unbiased

and with a minimum variance (Welch and Bishop, 1995). However, the linear dynamic is limiting and various extensions have been invented to extend the Kalman filter to non-linear cases. Extended Kalman filter is based on a linearisation of the dynamics with Taylor's formula. It can be useful for poorly non-linear dynamics, but suffers from instability and high computational cost (Reif et al., 1999). Unscented Kalman filter uses $2d+1$ points to estimate mean and covariance (unscented transform) with less instability and not higher computational cost (Julier and Uhlmann, 1997, 2004). Ensemble Kalman filter uses N points to estimate mean and covariance (Monte Carlo) and adds noise to the observations **?**Burgers et al. (1998); Evensen (2003). Despite estimation of mean and covariance are reliable (Mandel et al., 2011), the asymptotic filter is not the Bayesian filter when dynamics are non-linear (Le Gland et al., 2009). Kalman filters are very famous and give good results when the stochastic process to estimate has a nearly Gaussian distribution. But when the dynamic is strongly non-linear, the Gaussian assumption is strong and particle filters should be be preferred. Thus, to filter the wind measurements at turbulence scale, one needs particles filters.

Particles filters overcome some limitations of the Kalman family for strongly non-linear and/or non-Gaussian problems. But they are limited by degeneracy when applied to high dimensional systems (Snyder et al., 2008; Bengtsson et al., 2008), especially in geosciences (van Leeuwen, 2009). Three algorithms have been commented: sequential importance sampling (SIS), sampling with importance resampling (SIR) and genetic selection (Arulampalam et al., 2002; Doucet et al., 2000). Particles filters are widely used in other domains with non-linear/non-Gaussian filtering problem such as target positioning and tracking (Gustafsson et al., 2002), finance (Javaheri et al., 2003; Lopes and Tsay, 2011), audio signal processing (Fong et al., 2002) and more (Smith et al., 2013). Attempts to export it in meteorology for data assimilation has not been successful yet due to the limitations of particles filters for high-dimensional systems (van Leeuwen, 2009; Snyder et al., 2008). But other experiments where promising in oceanography (van Leeuwen, 2003) or with auxiliary particle filters (Baehr et al., 2010). At smaller scale than numerical weather prediction, Bernardin et al. (2009) used particle filters to refine the wind on a larger scale. At even smaller scale, Baehr (2008) applied particle filters on fast punctual wind measurements: this is the so-called *turbulence reconstruction method*. He also gave a theoretical description of the reconstruction in (Baehr, 2010). The first application to a Doppler lidar was done by Suzat et al. (2011) and Baehr et al. (2011). Recently, Rottner (2015) extended it to real 3D Doppler lidar measurements. The current work explores the 1D vertical staring Doppler lidar.

Chapter 3

Instrument and material

Contents

Lidars for wind measurements have become popular instruments over the last decades (Köpp et al., 1983; Hall et al., 1984; Menzies and Hardesty, 1989; Frehlich and Yadlowsky, 1994; Dabas et al., 2000; Grund et al., 2001; Cariou et al., 2011; Sathe and Mann, 2013). Applications of atmospheric Doppler lidars are numerous: boundary layer meteorology (Tucker et al., 2009; Neiman et al., 1988; Smalikho et al., 2005), wind energy (Kusiak and Song, 2010; Pichugina et al., 2011; Schlipf, 2014) and airport management (Chan and Shao, 2007; Hinton et al., 2000) for the most popular. In wind energy, lidars are used to measure the wind ahead of wind turbine and thus optimise (in both intensity and regularity) the electricity production or prevent from damages due to turbulence (Wharton and Lundquist, 2012; Schlipf, 2014). They can also assess the potential of a tentative of wind farm site (Kusiak and Song, 2010; Pichugina et al., 2011; Sun et al., 2012; Banta et al., 2013). In airport management, lidars measure the wake of aircrafts to optimise the use of runways (Hinton et al., 2000), to warn about wind shear (Chan and Shao, 2007; Chan et al., 2006) or pollution (Bennett et al., 2010). According to the last ASIAS Weather-related Aviation Accident Study[1], weather was a cause or a contributing factor in 20.1% of incidents reported between 2003 and 2007. Inside these 20.1%, wind, turbulence, up/down drafts and wind shear are responsible for respectively

[1]Hyperlink to the report on ASIAS website : http://www.asias.faa.gov/ (in 2017)

51.7%, 5.1%, 3.8% and 1.7% of incidents (ASI, 2010). Doppler lidars can help to reduce the risk around these phenomena if they are able to measure both wind and turbulence.

The turbulence reconstruction method has been invented for that purpose. As it is based on the lidar, this chapter tells more about this instrument. First, it is explained how a wind lidar works. Then, it is followed by a short focus on turbulence estimation with lidar. Next, the real data on which are based this study are presented, with a particular focus on missing data issues.

3.1 Instrument

3.1.1 The principle of measurement

Radars send radio waves in the atmosphere, lidars send light. The principle of the measurement is otherwise the same, only the wavelength λ of the electromagnetic wave sent changes. But a different λ makes radars and lidars sensitive to very different things. For the lidar used in the filtering, $\lambda = 1.5\mu$m.

The air is constituted of molecules with two or three atoms. Their size is about few nanometers. It also contains aerosols (fine particles like dust, smoke, sea spray...), about a micrometer large[2]. When light goes through atmosphere, it interacts with both molecules and particles. The scattering is different for molecules and particles because of their large difference of size.

For molecules, Rayleigh scattering is at work. Electrons and nucleus form an electric dipole under the effect of the incoming light. This dipole emits its own electromagnetic wave at the same frequency as the incident light. This emission is anisotropic and represented in figure 3.1. This phenomenon explains for example the blue color of the sky.

For particles, Mie scattering is at work. Each big particle reflects some part of light. Seen at the continuum scale, these successive reflections yield to anisotropic emission represented in figure 3.1 (case where the radius of particles a is comparable to λ). This phenomenon explains for example the white color of the cloud or the snow.

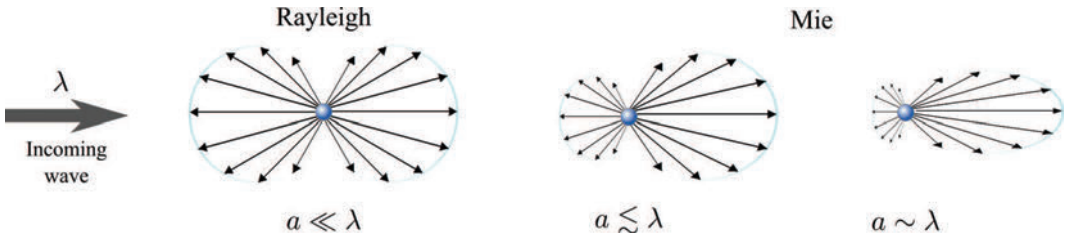


Figure 3.1 – Mie and Rayleigh directions of backscatter.

[2]The range of size is actually quite large for aerosols, with various concentration, from 0.01μm to 100μm.

In the atmosphere, each layer will add its part of scattering. Each layer will also absorb a bit of energy and let some light go through to the next layer. This is represented in figure 3.2. The lidar sends a light wave in the atmosphere and measures the backscattered part. Absorption, transmission and scattering take place on the way back too.

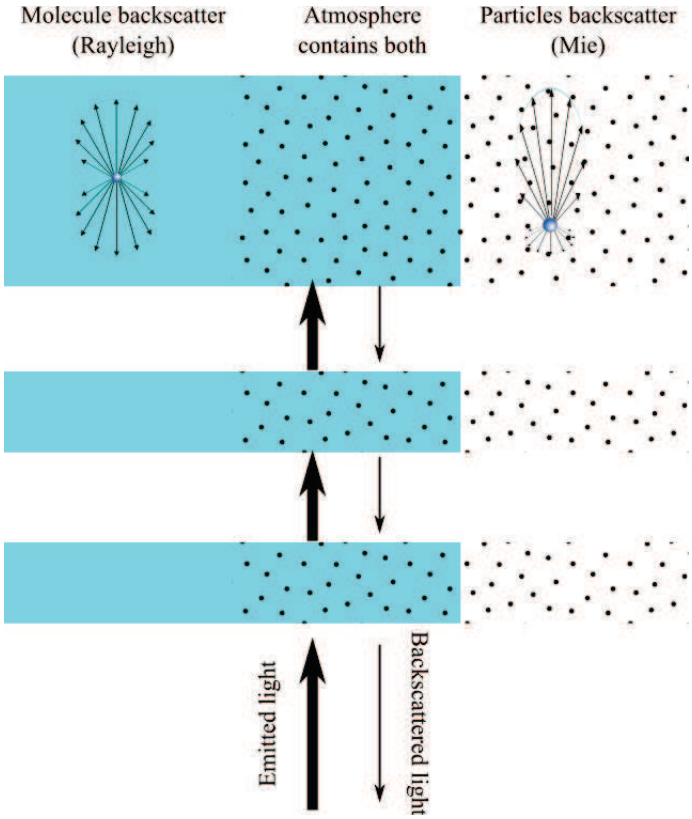


Figure 3.2 – Backscatter of light by particles and molecules in the atmosphere. Each layer transmits, absorbs and scatters the light. The amount of light coming back to the lidar is thus very small.

None of this interactions with the medium change the wavelength of the light. But if the particles responsible for the backscatter are moving, there will be a Doppler effect. In the axis of the beam (the line of sight), the wind blowing towards the lidar will increase the frequency of the wave, while it decreases the frequency when blowing away. The Doppler equation is the following:

$$\Delta\nu_d = \frac{2v_{los}}{\lambda} \tag{3.1}$$

with $\Delta\nu_d$ the Doppler frequency shift, v_{los} the wind speed component in the line of sight and λ the wavelength of the light. The Doppler shift in the frequency of the light coming back is measured and this give the wind component in the line of sight. It is very important to keep in mind that a Doppler lidar measures only the wind in the line of sight. It is a limitation known as the Cyclops dilemma: how to find a 3D wind with only one direction of measure?

This introduction is focused on Doppler lidar but there exists myriads of different lidars. Instead of atmosphere, some lidars point to hard targets. Such systems are simpler because the signal coming is stronger and not diffuse. Although, they are used for many applications (altimetry, cartography, archaeology...). Some lidars focus on the Rayleigh scattering (Rayleigh lidars) while other exploit Mie scattering. Doppler lidars are usually based on Mie scattering because its frequency response is more accurate. Some lidars measure directly the signal coming (direct lidar : handy for power measurement). Others measure the backscattered signal mixed with a local oscillator of known frequency (heterodyne lidar: handy for frequency measurement). Some lidars send pulses of light, while other send a continuous wave. All these forks are represented in figure 3.3. Since only heterodyne pulsed Doppler lidar are used in this work, from now we exclusively focus on this type of lidar.

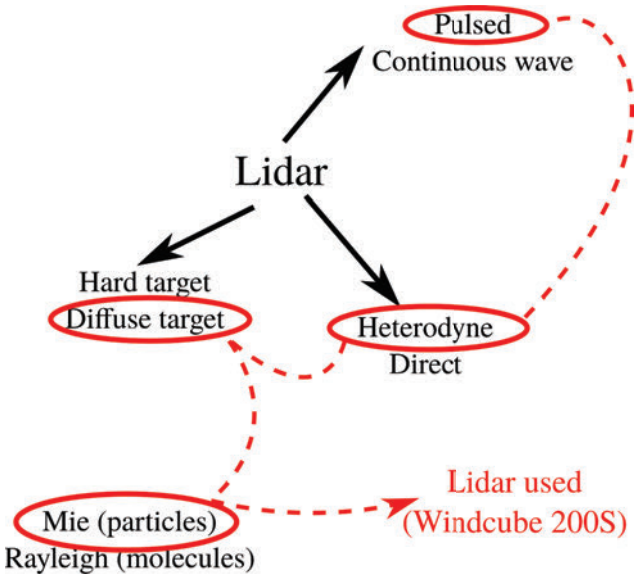


Figure 3.3 – Basic overview of major differences in lidars. Characteristics signalized in red are the ones of the lidar for which the filtering method is designed.

3.1.2 Lidar equation for pulsed emission

The lidar considered in this work is a heterodyne pulsed Doppler lidar. The light is emitted by pulse and the shape of the pulse is usually assumed Gaussian, with a standard deviation τ about few hundreds of nanoseconds (Frehlich and Yadlowsky, 1994). The pulse is emitted in the atmosphere (emission phase), then the optic channel is let free to receive the light coming back (reception phase), as shown in figure 3.4. This is repeated N times and the power coming back from the N pulses is used to get a single measurement. One pulse sent at $t = 0$ goes up to the distance z and comes back at the time t. This is repeated for all atmospheric layers. The resulting power $S(t)$ is given by the *lidar equation*:

$$S(t) = K_0 \int_0^{+\infty} I\left(t - \frac{2z}{c}\right) \frac{\beta(z)T(z)^2}{z^2} dz + \epsilon_t \tag{3.2}$$

with $\beta(z)$ the backscatter coefficient, $T(z) = \exp\left(-\int_0^z \alpha(x)dx\right)$. the extinction coefficient and ϵ_t a noise due to imperfection of sensors and underlying approximations.

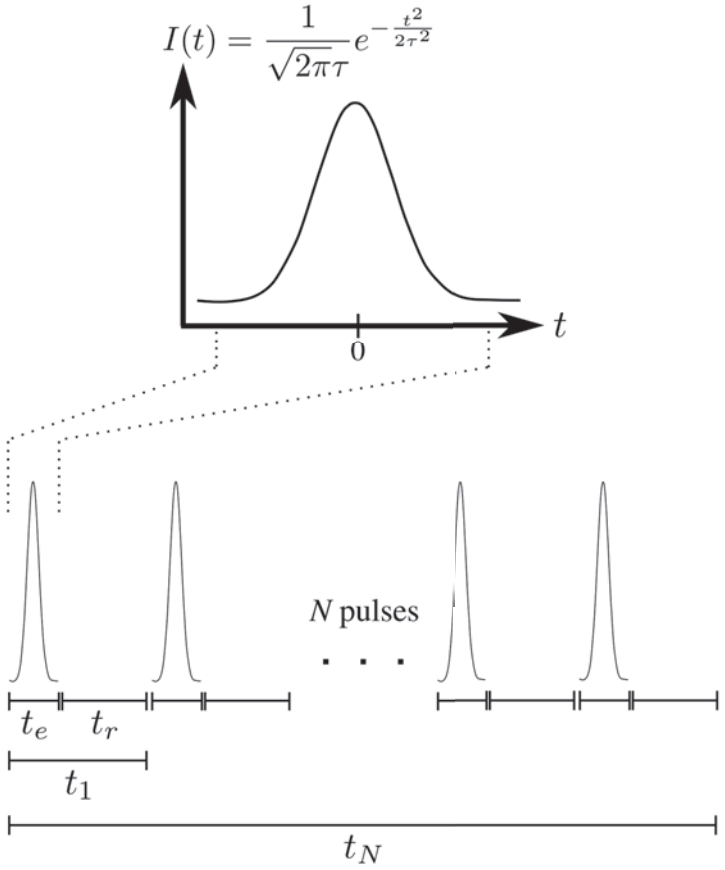


Figure 3.4 – Illustration of lidar emission: during the time t_e, the lidar sends one Gaussian pulse, then it receives during t_r. This forms the acquisition time for one pulse $t_1 = t_e + t_r$. It is repeated for N pulses.

If the pulse were a Dirac distribution, the power received at time t would be exactly the contribution of the layer at distance z from the lidar (with t and z related by $t = 2z/c$). The lidar equation written is terms of z is

$$S(z) = K_0 \frac{\beta(z)}{z^2} \exp\left(-2\int_0^z \alpha(x)dx\right) + \epsilon_z \tag{3.3}$$

The equation (3.3) has two unknowns: the backscatter coefficient $\beta(z)$ and the attenuation coefficient $\alpha(z)$. Both can be separated into a molecule and a particle contribution. This formula highlights the role of aerosols in the lidar measurement: without aerosols, no light is backscattered. But the amount of power coming back from the atmosphere is not the target quantity. It is the carrier from the target information, which is the Doppler shift.

3.1.3 Space-time ambiguity

The pulse is a Gaussian of finite duration t_e (Frehlich and Yadlowsky, 1994). Every layer of atmosphere is illuminated during t_e. Hence the signal received at time t_r sums up all the backscattered light arriving at that time. It is representative of a chunk of atmosphere of size $\Delta z = ct_e/2$. For the lidar used in this work, t_e is set to 400ns, which gives Δz about 60m.

In practice, the spatial resolution is set by the number of acquisition points considered. In the figure 3.5 is an example of signal in output of the receiver for one pulse. When the beam is going through the telescope, a lot of light is coming back directly onto the receiver because of parasite reflections. This is visible in the first part of the signal. Then the signal coming back from the atmosphere is much weaker. It is sequenced into chunks (so-called range gates), displayed in the figure. On the example of figure 3.5, vertical levels are about 30 meters. The choice of the range resolution is thus a trade-off between the quality of the spectrum estimation and the spatial resolution of the lidar. The number of points in a single gate is denoted M in the literature (Dabas, 1999).

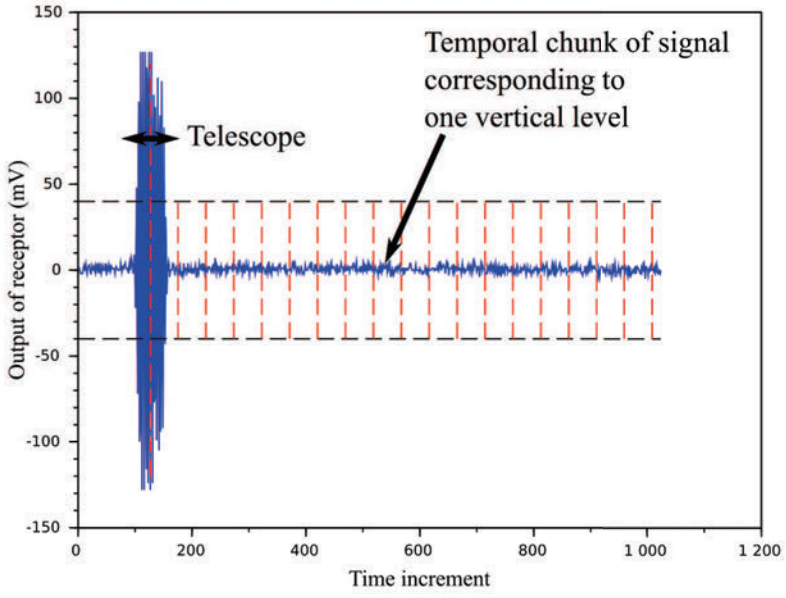


Figure 3.5 – Example of signal coming back from one pulse. The area of big signal at the beginning is the reflection on the telescope. The temporal signal is divided into chunks to isolate vertical levels. Data from lidar WindCube V2: $N = 1000$ pulses, 1024 points per pulse, $\Delta z = 30$m, $M = 49$ points per vertical level.

Figures made with real data from a WindCube V2. Courtesy of Leosphere for the summer school *Lidar for geophysics* (Aspet, 2015).

As one can see in figure 3.6 the signal measured in the i-th range gate (that is to say measured between times t_r^i and t_r^{i+1}) comes from the volume of atmosphere between z_0 and $z_0 + \Delta z$. The spatial resolution Δz is mostly driven by the size of the range date.

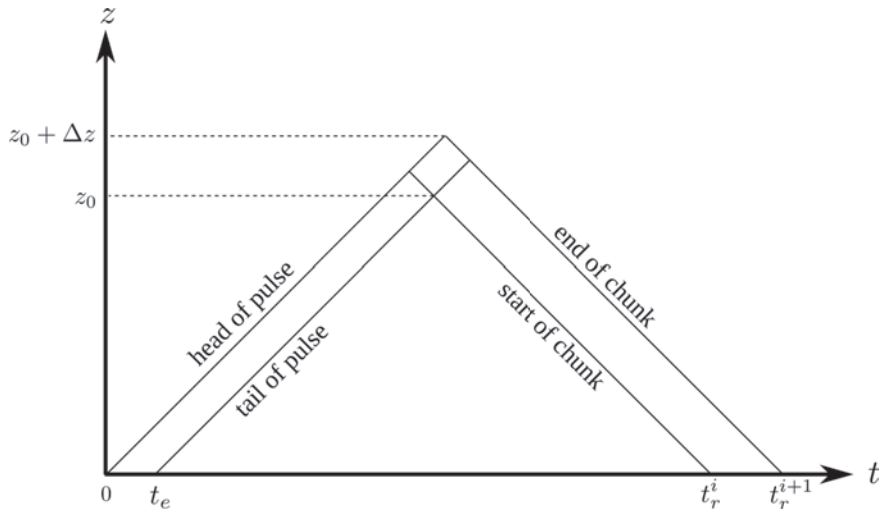


Figure 3.6 – Illustration of time-space ambiguity to due the temporal extent of the pulse and the temporal subdivision of the signal into range gates.

As a consequence, the lidar measure is not punctual. The measurement of wind at a given range is the average wind on Δz meters along the line of sight.

3.1.4 Heterodyne detection

Heterodyne detection, conversely to direct detection, does not provide intensity measurement. It intents to measure better the frequency shift of the signal. Indeed, the pulse sent in the atmosphere represented in figure 3.4 is a pulse of intensity of the electromagnetic wave. But this electromagnetic wave has a frequency given by the wavelength of the emitting laser.

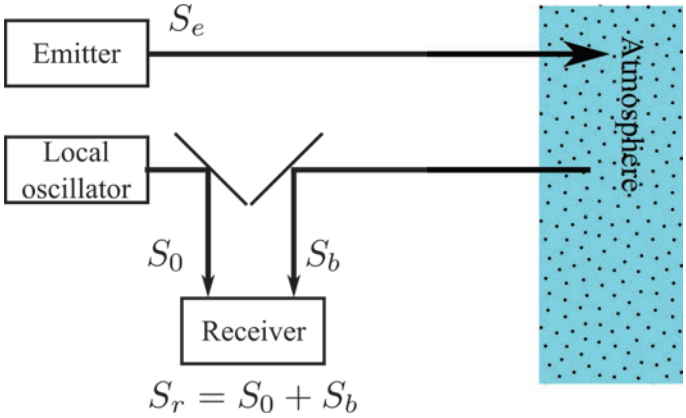


Figure 3.7 – Principle of heterodyne detection: an auxiliary wave is added on the receiver to improve frequency retrieval. The order of magnitude are $\nu_e \simeq 300$ THz, $\nu_r = \nu_e + \Delta\nu_d$, $\nu_0 = \nu_e + \Delta\nu_0$ with $\Delta\nu_0$ and $\Delta\nu_d$ about few MHz.

The emitted signal S_e is an electromagnetic wave of frequency ν_e. In complex notation (complex denoted $i = \sqrt{-1}$), it can be written as follows:

$$S_e(t) = E_e e^{-2\pi i(\nu_e t + \varphi_e)} \tag{3.4}$$

Similarly, the local oscillator and the backscattered signal are written as follow:

$$S_0(t) = E_0 e^{-2\pi i(\nu_0 t + \varphi_0)} \tag{3.5}$$

$$S_b(t) = E_b e^{-2\pi i(\nu_b t + \varphi_b)} \tag{3.6}$$

with $\nu_b = \nu_e + \Delta\nu_d$ (the frequency of the backscattered wave shifted by the Doppler effect) and $\nu_0 = \nu_e + \Delta\nu_0$ (the local oscillator has the same frequency as the emitter shifted by a known value $\Delta\nu_0$). The receiver is only sensitive to illuminance $\mathcal{E}_r(t) = |S_r(t)|^2$. The illuminance of a single wave is constant.

$$\begin{aligned} \mathcal{E}_r(t) &= |S_0(t) + S_b(t)|^2 \\ &= |S_0(t)|^2 + |S_b(t)|^2 + 2\Re\left(S_0(t)S_b(t)^*\right) \\ &= \mathcal{E}_0 + \mathcal{E}_b + 2\Re\left(S_0(t)S_b(t)^*\right) \end{aligned}$$

The two first terms vary with very low frequency. The only term left with rapid variation over time is the interference term. It is the term from which we get the Doppler shift. Indeed, it is written as follows (the superscript * denotes the complex conjugate).

$$2\Re\left(S_0(t)S_b(t)^*\right) = 2\Re\left(E_0 E_b e^{-2\pi i((\nu_0-\nu_b)t + \varphi_0 - \varphi_b)}\right) = 2E_0E_b\cos\left(2\pi\left((\nu_0-\nu_b)t + \varphi_0 - \varphi_b\right)\right)$$

Then, the signal in output of the receiver ($\mathcal{E}_r(t)$) is a single sinusoid of frequency $\nu_0 - \nu_b = \Delta\nu_0 - \Delta\nu_d$. The spectrum of $\mathcal{E}_r(t)$ is thus a Dirac at the frequency $\Delta\nu_0 - \Delta\nu_d$. Since $\Delta\nu_0$ is known, $\Delta\nu_d$ is measured as soon as one gets the position of the peak in the spectrum of $\mathcal{E}_r(t)$. The role of $\Delta\nu_0$ is to enable the lidar to distinguish the cases $\Delta\nu_d > 0$ and $\Delta\nu_d < 0$. If $\Delta\nu_0 = 0$, the sign of $\Delta\nu_d$ is lost because cosine is a pair function.

In practice, the spectrum of the output signal is not such a strong peak but rather something like in figure 3.8. There are peaks in the spectrum, but it is not clear which are due to noise and which are due to Doppler. To sort the noise out, the spectrum is averaged over many pulses. Figure 3.9 is an example of 1000 pulses averaged. The Doppler peak is clear, and can thus be estimated precisely. However, the ground level of the spectrum is noticeably high, which denotes a poor signal-to-noise ratio. The noise is due to the random arrival of photons. Photons reach the receiver at a random time, which makes the so-called shot noise. The shot noise look strong in the figure 3.9 because the signal itself is really small.

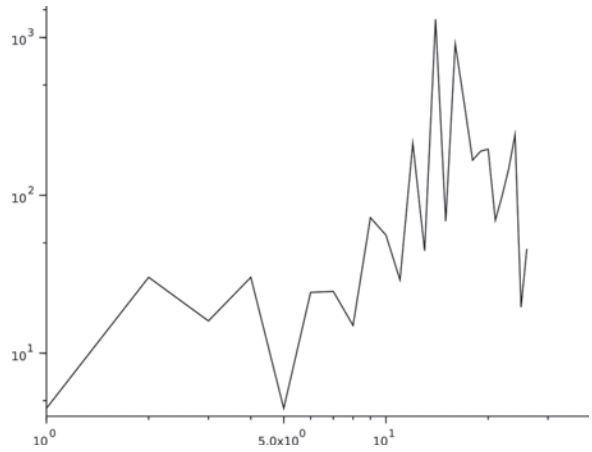


Figure 3.8 – Example of spectrum of the output signal for one pulse. The Doppler peak is drawn into noise.

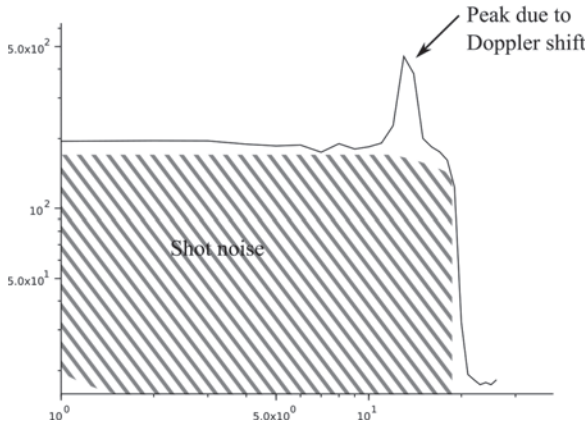


Figure 3.9 – Average of 1000 spectra, corresponding to 1000 pulses. The peak due to the Doppler shift is clearly visible, despite a strong noise base (shot noise).

Figures made with real data from a WindCube V2. Courtesy of Leosphere for the summer school *Lidar for geophysics* (Aspet, 2015).

3.1.5 Frequency analysis

To find the peak visible in figure 3.9, the technique is called adapted filtering. We explain only the technique used here but various options exist (Levin, 1965; Dabas et al., 1999, 2000). We consider a pass-band window multiplied to the spectrum. An example of window is provided in figure 3.10. It is known as Levin window. It has for equation

$$h(\nu) = \frac{1}{1 + \frac{A}{\sqrt{2\pi}\sigma} e^{-\frac{\nu^2}{2\sigma^2}}}$$

The product of $h(\nu)$ with the spectrum will have a minimum integral when the window is centred on the peak, as illustrated in figure 3.11. Hence, the convolution of the window and the spectrum (equation 3.7) will be minimum at the frequency of the peak.

$$(h * s)(\nu) = \int s(f)h(\nu - f)df \tag{3.7}$$

The Doppler shift is then obtained by taking the frequency where the convolution is minimum:

$$\Delta\nu_d = \Delta\nu_0 - \text{argmax}\left\{(h * s)(\nu)\right\}$$

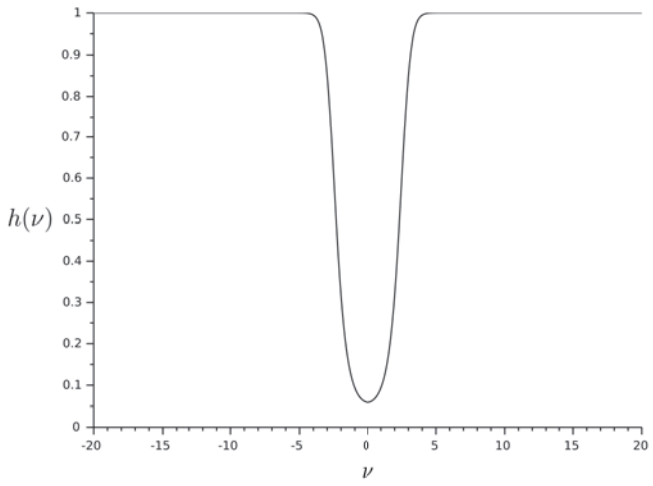


Figure 3.10 – Example of window for adapted filtering (Levin window).

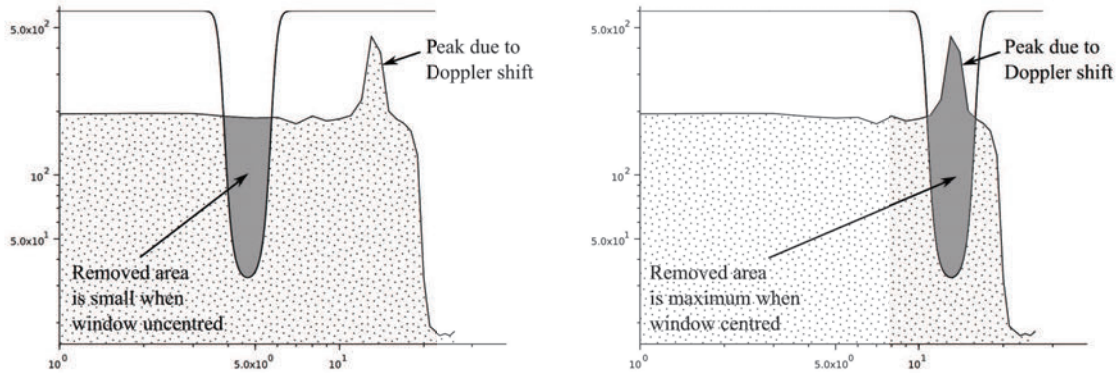


Figure 3.11 – Illustration of adapted filtering. A window is convolved to the spectrum and the peak is found at the minimum of the convolution.

3.1.6 Carrier-to-noise ratio estimation

The signal out of the receiver is poisoned by noise. The perfect signal $\mathcal{E}_r(t)$ is added with a noise n_t.

$$\mathcal{E}_s(t) = \mathcal{E}_r(t) + n_t$$

The power of the signal $\mathcal{E}_s(t)$ can be decomposed in the sum of the power of $\mathcal{E}_r(t)$ and the power of the noise.

$$\mathcal{P}_s = \mathcal{P}_r + \mathcal{P}_n$$

with $\mathcal{P}_s = \mathbb{E}\left[|\mathcal{E}_s(t)|^2\right]$, $\mathcal{P}_r = \mathbb{E}\left[|\mathcal{E}_r(t)|^2\right]$ and $\mathcal{P}_n = \mathbb{E}\left[|n_t|^2\right]$.

The signal-to-noise ratio (SNR) is defined as

$$SNR = \frac{\mathcal{P}_r}{\mathcal{P}_n}$$

The SNR compares the signal received from the atmosphere with the noise. In heterodyne detection, only the Doppler peak is valuable in the signal. The signal coming back from the atmosphere is just a carrier for the Doppler shift. That why it is more correct to talk about carrier-to-noise ratio (CNR):

$$CNR = \frac{\mathcal{P}_{Doppler}}{\mathcal{P}_n}$$

As the signal is only a peak located on the spectrum, the convolution (3.7) is assumed to remove all the signal when the window is centred on the peak. At this point, in the figure 3.11 (right panel), the grey area is assumed to be $\mathcal{P}_{Doppler}$ and the dotted area is assumed to be $\mathcal{P}_n$. Hence the CNR is estimated at the same time as the Doppler shift, with adapted filtering too.

$$\widehat{CNR} = \frac{\int s(\nu)d\nu - \min\{(h * s)(\nu)\}}{\min\{(h * s)(\nu)\}}$$

3.1.7 Uncertainty on wind measurements

As we have seen in the previous section, the wind measure is done as follows. The lidar sends N pulses in the atmosphere. For each pulse, the signal out of the receiver is sampled at the frequency $1/T_s$[3] (T_s about 10^{-6} seconds). The signal of one pulse is sequenced into slices of M points corresponding to vertical levels (see figure 3.5). On the M points of a single slice, the spectrum is computed (see figure 3.8). This is repeated for the N pulses and the N spectra are averaged (see figure 3.9). Then the Doppler shift is found by adapted filtering (see figure 3.11).

To qualify the uncertainty on the wind measurement, it is common to distinguish the

[3]In reality, the signal out of the receiver is analog, but it is manipulated and stored on computers after analog-to-digital conversion.

reliability and the accuracy. Empirically, the distribution of wind measurements appear to be composed of a fraction b of uniform-distributed outliers and a fraction $1-b$ of normal-distributed measurements centred on the good value and with a standard deviation g. The parameter $1-b$ is the *reliability* and the parameter $1/g$ is the *accuracy* (Frehlich, 1997, 2001).

These two parameters are function the CNR (see figure 4 of Frehlich and Yadlowsky (1994)). For low CNR, the reliability collapses: the fraction b of outliers increases significantly. In the meanwhile, the accuracy is degraded too. For high CNR, b is almost zero, g is the smallest and no longer depends on CNR.

Several studies have been carried out to master the lidar error. Dabas (1999) gives a semi-empirical model for the reliability as a function of the lidar parameters. Dabas et al. (2000) study the bias induced by asymmetry of the pulse. Dabas et al. (1998) study the bias due to chirp effect. O'Connor et al. (2010) give an expression of the error against the CNR (equation 7). For the lidar used in the filtering, the order of magnitude for uncertainty is about 0.1 m/s (according to the constructor paper Cariou et al. (2011), figure 6).

The error of lidar can be assumed Gaussian when the CNR is high enough (b is almost zero in that case). Several theoretical expressions have been found, as well as efficient ways of estimations. For the non-linear filtering invented by Baehr (2010), the measurement error has to be Gaussian and a guess of its variance must be provided. In this work, the true variance of the measurement noise is denoted σ^{add} and the guess provided to the filter is denoted σ^{obs}.

3.1.8 TKE estimation

As lidars measure the wind in the boundary layer with fine time and spatial resolution, they are good candidates for measurement of turbulence. In the previous chapter, we have seen the turbulent kinetic energy (TKE) is a desired parameter. Because the lidar measures only the component of the speed along the line of sight, the estimation of TKE is depends strongly on the scanning geometry. The figure 3.12 shows the most common scanning strategy.

The lidar used by Rottner (2015) is a WindCube V2. It does Doppler Beam Swinging (DBS) scans every 4 seconds. For such type of lidar, the TKE estimation is described in (Kouznetsov et al., 2004) (the instrument is a sodar in the paper, but the retrieval is the same). The lidar measures along 5 beams pointing respectively to north, west, south, east with the same elevation angle φ. Each beam measures the wind along its line of sight: v_n, v_w, v_s, v_e, v_z. For each of these estimates, it is possible to compute a time variance: σ_n^2, σ_w^2, σ_s^2, σ_e^2, σ_z^2. Assuming the wind is homogeneous in the whole volume covered by the scan, the estimated TKE is given by:

$$\widehat{k} = \frac{\sigma_n^2 + \sigma_w^2 + \sigma_s^2 + \sigma_e^2}{4\sin^2\varphi} - \sigma_z^2\left(\frac{1}{\tan^2\varphi} - \frac{1}{2}\right) \tag{3.8}$$

The main attractive feature of turbulence reconstruction over such estimation is to relax

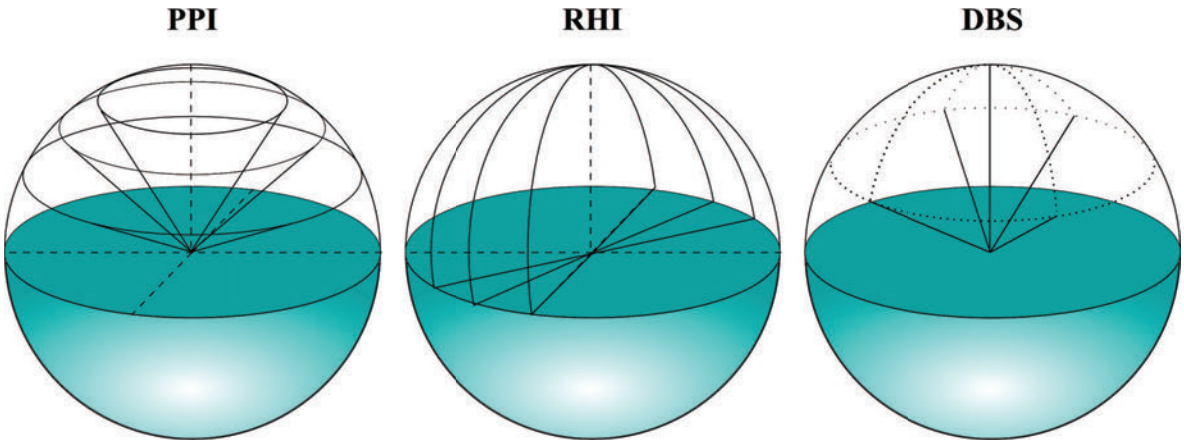


Figure 3.12 – Three common scanning geometries : PPI (Plan Position Indicator), RHI (Range Height Indicator) et DBS (Doppler Beam Swinging).

the assumption of homogeneity. Full homogeneity is needed to get an estimation only from geometric considerations. The reconstruction, helped by the Lagrangian model, is able to propagate information inside the volume delimited by the beams. The strict homogeneity is replaced by local homogeneity of the acquisition process (Baehr (2010), definition 2.1). Moreover, the computation of the variance for each line of sight requires many lidar signals. Typical integration time are a few tens of minutes. With the reconstruction, particles are spatially spread, thus a spatial variance can be calculated at each time step, that is to say every 4 seconds for this lidar.

The filtering method is tested in the 1-dimensional case. Hence, only vertical staring Doppler lidar data are used. Since only one component of the wind is available, the proper TKE is not reachable because the two horizontal components of the wind are missing. Nevertheless, we will still do the misuse of language to call TKE the half variance of the vertical component.

$$k_{z,t} = \frac{1}{2}\mathbb{E}\left[\left(V_{z,t} - \mathbb{E}\left[V_{z,t}\right]\right)^2\right]$$

Without particles, it is estimated with a time variance

$$k^T_{z,t'} = \frac{1}{2}\overline{\left(V_{z,t} - \overline{V_{z,t}}^{\tau}\right)^2}^{\tau}$$

With particle, it is possible to calculate a space variance, which has much higher temporal resolution.

$$k^S_{z,t} = \frac{1}{2}\left\langle\left(V_{z,t} - \langle V_{z,t}\rangle\right)^2\right\rangle$$

with t the "fast" time counter (every 4s) and t' the "slow" time counter ($\sim$ 10 minutes). Such TKE estimates are presented in the chapter 1 with comparison and comments (see page 27).

This section presented the principle of measurement of the lidar instrument for curious novice. The key points that will have consequences in the following are that the lidar measures

only the wind component in the line of sight. Since our lidar points vertically, only the vertical velocity is measured. No information about the horizontal wind can be found with such scanning geometry.

3.2 Material

3.2.1 Data and field experiment

The filtering method have been applied to real lidar data coming from a field campaign called BLLAST. Briefly, the BLLAST experiment (for Boundary Layer Late Afternoon and Sunset Turbulence) is a research program accompanied with a field measurement campaign gathering many actors (up to 26 institutes in 9 countries) that aims to describe the late afternoon transition of the atmospheric boundary layer by using large eddy simulations and a wide range of instruments (Lothon et al., 2014). The Doppler lidar used during BLLAST is a WindCube 200 (from Leosphere). It is a pulsed laser emitting at a 1.5μm wavelength. It points vertically only, providing vertical velocity measurements with a 50m spatial resolution. These informations are summarized in the table 3.1 and further information are in Cariou et al. (2011).

Wavelength	1.5μm
Measurement frequency	3.9s (average)
Spatial resolution	50m
Minimum range	100m
Maximum range used	800m

Table 3.1 – Basic characteristics of the Doppler lidar that provided the data.

As described in (Lothon et al., 2014), intensive observation periods (IOP) took place during fair weather episodes, because the influence of solar radiation is enhanced. The measurements period was from 14 June to 8 July 2011 with 11 IOPs over complex terrain near Lannemezan (southern France) during fair weather episodes. The lidar was active during several IOPs. The lidar measures up to 1800m, but it is often limited by the top of the boundary layer (lack of aerosol or cloud base). Thus, we limited our study to the vertical portion between 100m and 800m of altitude (where still remain missing data).

The measurements used for this work must be of very good quality because there used as a reference in the experiments (see section 5.4.1 for the description of the experiments). The selected measurement are during the IOP 8 (30^{th} of June 2011), from 14:42 to 16:45 local time (12:42 to 14:45 UTC). The figure 3.13 shows the entire IOP 8, with a shaded rectangle to highlight the selected measurements. The meteorological conditions are fair, with intermittent cloud cover (fair weather cumuli) and weak westerly wind. This set of nearly two hours of data have been chosen to be in well developed turbulence, because the Lagrangian model have been built for such conditions. Other dataset have been used occasionally (IOP 9 and 10) to test the filtering method after the development phase, but they are not included in this manuscript. Only the missing data processing developed at this occasion is presented (next section).

The measurement frequency is actually not constant, according to meta-data of the

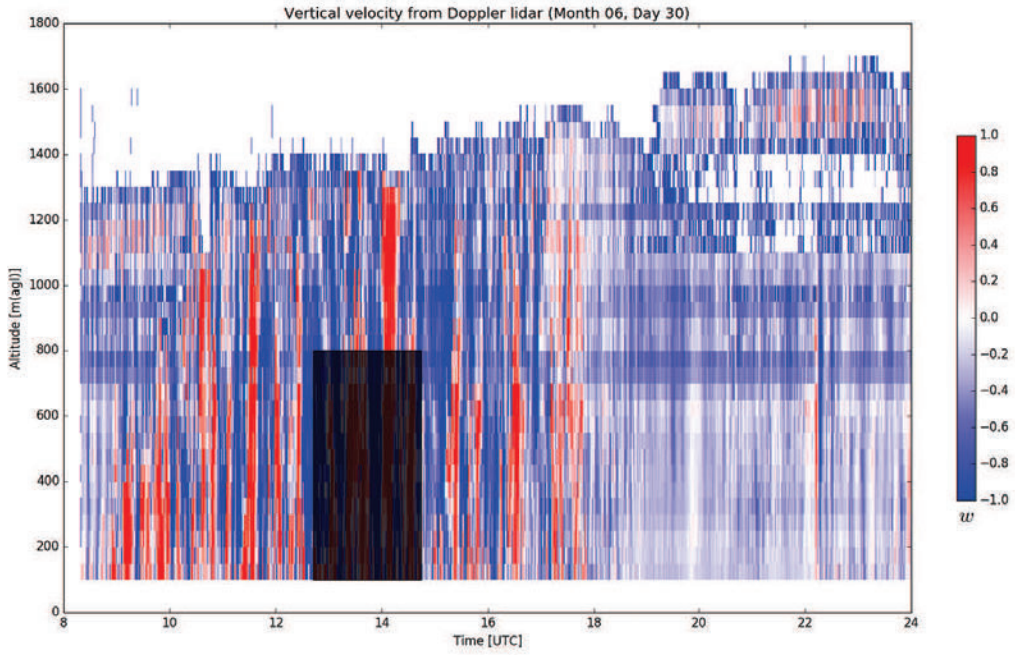


Figure 3.13 – Measurements during the IOP 8. Measurements used as reference are in the shaded area, from 12:42 to 14:45 UTC (14:42 to 16:45 local time) and from 100m to 800m (14 vertical levels) without missing data. Fully developed turbulence conditions.

Doppler lidar. Most of the time, it is very close to 3.9s, but sometimes the lidar stops without notice, making time intervals very large (up to 1 hour). The reason of these stops is the temperature regulation of the laser: under the heat of July, the laser needed to be stopped regularly to cool down. In figure 3.14 is shown a more precise description of the time step variation. The actual measurement time is drawn against the time step number. If the instrument measured continuously and regularly, it should be a straight line with slope the measurement frequency. One can see that it is not the case. More, a zoom on the small time steps shows that they are not constant but varying around the 3.9s seconds value.

Thus, it is not correct to assume the time step as constant and some processing has to be done to fill the large gaps between measurements with missing values. This correction will be described in the next paragraph, as the *irregular measurements* issue.

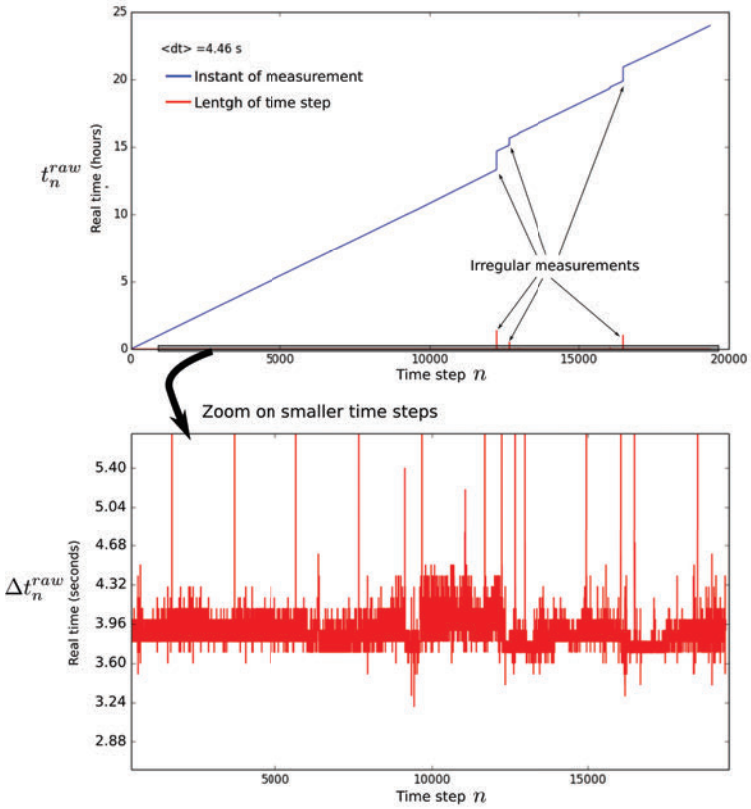


Figure 3.14 – Measurement times are not regular. The blue line is the actual measurement time, drawn against the time step number, the time step length is in red. A zoom on the small time steps shows that they are not constant. Some jumps have to be filled with missing values (NaN).

3.2.2 Missing data processing

Working with real measurements raises the question of how to deal with missing data? Although the final dataset (figure 3.13) is free of missing data, the filtering algorithm have been tested on others datasets (the entire IOP 8, 9 and 10) and these IOP suffer from missing data. The following process has been built to apply the reconstruction method to this datasets. Since this process is specific to the reconstruction system, some elements evoked here might be presented together later ("selection step", "estimated quantities A_t and ε_t"), at the chapter 5. They have been presented separately in the previous chapters. The missing data processing presented here is included in the patent Baehr et al. (2016).

Missing data can disturb a code computation, or even make it crash. Even if no trouble occur in the computer experiment, missing data can still introduce statistical malfunctions (e.g bias, misinterpretation...). The question to solve in the presence of missing data is then double:

- How to overcome missing data issues in the computer experiment?
- What is the effect of missing data and their processing on the output?

The first question will be answered in this section, for this specific computer experiment. First, issues are classified and an idea of the processing is given. Then, the complete processing is illustrated on an example.

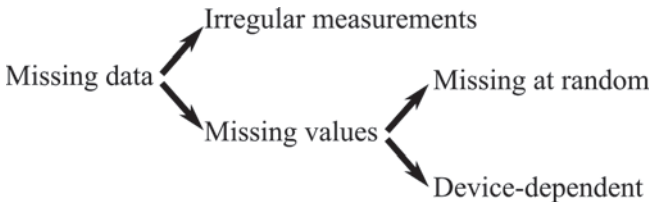
The second question is expected to be "none". But a devoted experimental setup is needed to provide such an answer, and this work has not been carried out here. However, the Bayesian filtering uses two sources of information (observation and model). So we can think that when the observation is missing, the model can complete the information in a statistically correct way.

The missing data community (Schafer and Graham, 2002; Rubin, 1976) is mostly oriented toward statistics, medicine or social sciences practitioners. They also use the Bayesian framework, mostly to complete the data set with non poisoning values (so-called data imputation).

3.2.2.1 Classification and idea of processing

In this work, we distinguished two types of missing data issues, shown on the example of figure 3.15.

- *Missing values*: measurement value is not present but signalized as such (NaN).
- *Irregular measurements*: the time between two consecutive measurements is unusually long.



Missing values can be of two kinds : missing at random or device-dependent. The data missing at random are due to punctual malfunction affecting every measurement device. The device-dependent missing data are due to the limitations of the measurement device. For Doppler lidar, the most common sources of device-dependent missing data are range limitations due to lack of aerosols and obstruction by rain or cloud droplets. The reconstruction system should not be affected by few missing values. But when there are too many missing values, there is no longer sense to filter unavailable data.

Outliers are detected with a simple thresholding (threshold value : 12 $\mathrm{m{\cdot}s^{-1}}$ for vertical velocity without deep convection) and marked as missing. Then, they are treated like the other missing values.

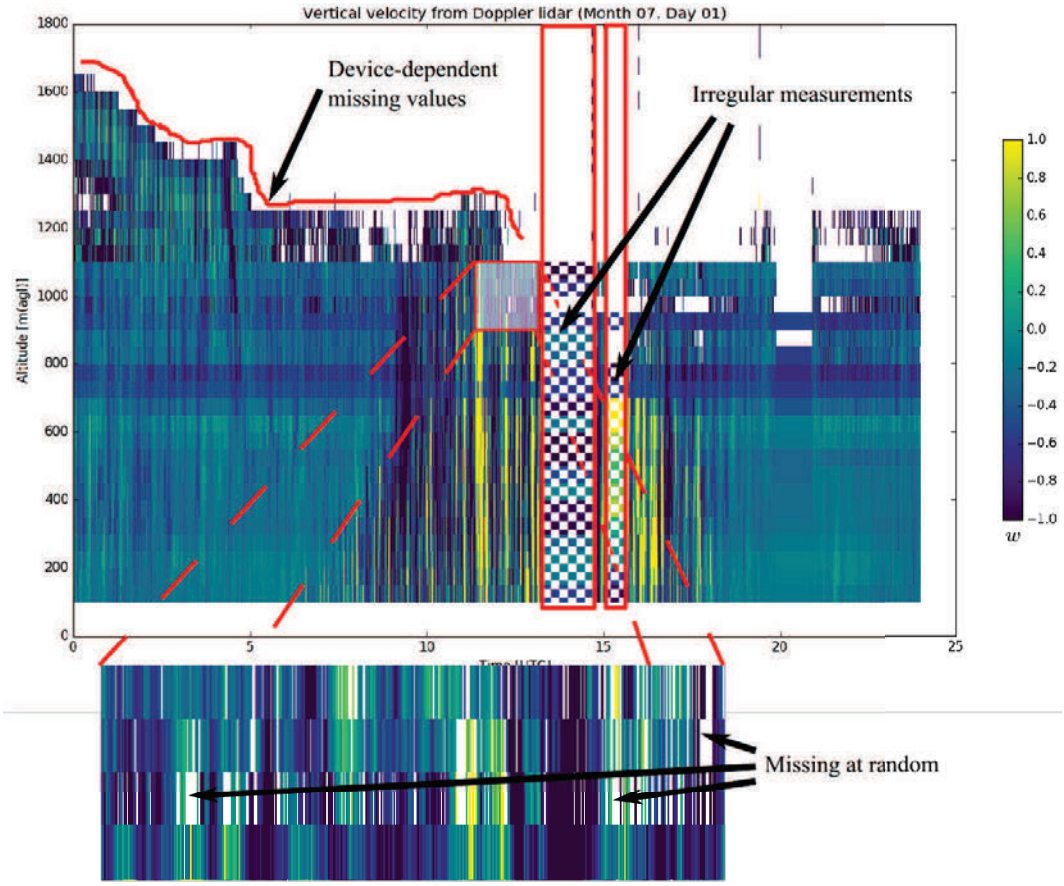


Figure 3.15 – Illustration of the different categories of missing data on the data from IOP 9.

3.2.2.2 Irregular measurements processing

We have seen in figure 3.14 that the time is not constant, with sometimes a long time (up to 1 hour) between two consecutive measurements. They might be due to external events, easily detectable, but rarely recorded (power cut, temperature regulation, wiper action, operator intervention...). To solve this issue, we have chosen to fill the time interval without measure with missing values, and then we are back to the missing values problem.

This process is illustrated by the figure 3.16. Irregular measurements are detected by looking at the real time difference between the measurements. When it exceeds a given threshold, a gap is detected. This threshold has been chosen as

$$\Delta t_{gap\ detection} = 2 \min \left\{ \Delta t_n^{raw},\ n \in [\![1, N_t^{raw}]\!] \right\} \tag{3.9}$$

where $\Delta t_n^{raw} = t_n^{raw} - t_{n-1}^{raw}$ is the time difference between two consecutive measurements, as given by the instrument. This choice ensures that the irregularity of the measurement time does not cause aliasing.

When a gap is detected, extra measurements instants are introduced with missing values.

The number of virtual measurements N^{extra} is chosen in order to ensure that the resulting time step stay close to a characteristic time step, denoted $\overline{\Delta t}$.

$$N^{extra} = \left\lfloor \frac{\Delta t_n^{raw}}{\overline{\Delta t}} \right\rfloor - 1 \tag{3.10}$$

where $\lfloor \cdot \rfloor$ denotes the floor function.

The virtual measurements are distributed regularly, each Δt^{loc}, within the gap. It is possible to demonstrate that Δt^{loc} is bounded, with extremum boundaries given by inequalities:

$$\overline{\Delta t} \leqslant \Delta t^{loc} \leqslant 2\overline{\Delta t} \tag{3.11}$$

The characteristic time step must be representative of the time steps without irregular measurements. We took the median of the raw time steps which do not exceed the gap threshold.

$$\overline{\Delta t} = \text{median}\left\{\Delta t_n^{raw},\ \Delta t_n^{raw} \leqslant \Delta t_{gap\ detection},\ n \in [\![1, N_t^{raw}]\!]\right\}$$

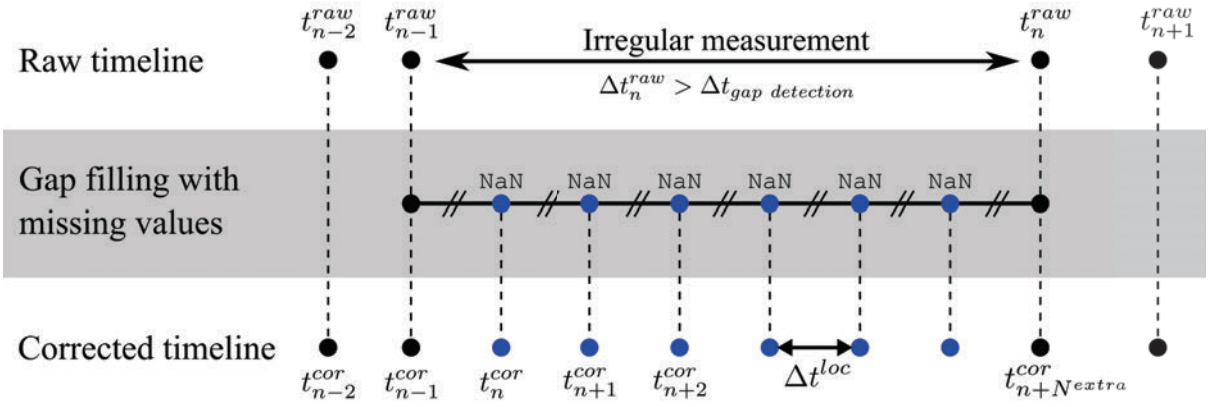


Figure 3.16 – Illustration of irregular measurement issues: the gap is filled with missing values. The N^{extra} new points are spread regularly. N^{extra} is chosen such that the corrected time step is close to the median time step.

Irregular measurements are then treated as missing data. The missing data processing relies on the assumption that missing data are noticed, which is not the case for irregular measurment. Thus, irregular measurement processing is done prior than missing data processing.

3.2.2.3 Missing values processing

When measurements are missing, different processes are made according to the length of the series of consecutive missing values.

In the case of long series of missing observation, the system is stopped and restarted at the next available observation. Without control of observation, the model degenerates quickly. The purpose of reconstruction, which is information retrieval from observation, no longer makes sense. The threshold in use to discriminate tolerable length of missing observation series is

$$\Delta t^{restart} = 8\overline{\Delta t} \tag{3.12}$$

For tolerable length of missing data, we trust the Lagrangian model to give reliable information. But the management of the particles changes when there is information to check them. The following changes are done

- Selection step is skipped.
- No update of estimated quantities (A_t and ε_t)
- Modulation of turbulent frequency by a "forget coefficient", decreasing as a Gaussian.

These changes are theoretically coherent with the system and enables it to run correctly. But more purpose-oriented experiments are needed to assess precisely the effect of these choices on the results.

3.2.2.4 Summary and illustration

Missing data processing is illustrated on an example. In the figure 3.17, three timelines of an toy data set are chosen. Missing value are represented as grey rectangles.

- (A) : meta-data from the instrument (location of known missing data).
- (B) : (A) after irregular measurements processing.
- (C) : (B) after removing missing observation series where a restart is required.

On the C timeline, only the tolerable missing observation series remain. Too large series of missing data have been removed by re-initializing the reconstruction. They are treated as mentioned before: selection skipped, "forget coefficient". To illustrate the processing between A and C, we focus on 4 particular cases (pointed by numbers in parenthesis on the figure 3.17).

1. Isolated missing values, not necessitating a restart. The processing does not affect them.
2. Wide irregular measurement: a large time between 2 consecutive measurements has been filled with missing values (see timeline B). The gap is large enough to necessitate a restart (see timeline C).

3. The gap between the consecutive measurements is filled with missing values (timeline B). Added to already present missing values, it requires a restart.

4. Small irregular measurement: the gap filled by missing values is small and does not necessitate a restart (timeline C).

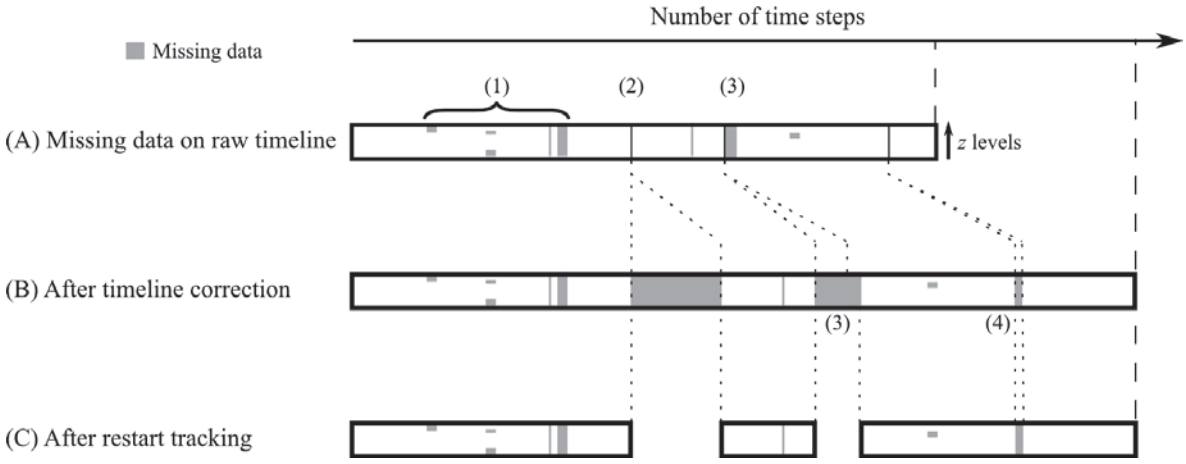


Figure 3.17 – Illustration of missing data processing.

3.3 Conclusion

The heterodyne pulsed Doppler lidar has been presented. It emits light pulses and between two emissions, it detects their echo from the atmosphere. The signal comes back because of particles backscatter. The movement of particles with wind induces some Doppler shift on the light coming back. This Doppler shift is measured thanks to heterodyne detection and adapted filtering. It has been underlined that the lidar measures only the component of wind along the line of sight. The measure is not punctual but averaged on a given range gate.

The uncertainty of the measurement is an subject of intensive research. Some references have been given and commented. The reconstruction intends to improve the lidar measurements, especially when they are noisy. Turbulence estimations depend on the scanning strategy, but usually assume frozen wind field in the whole volume analysed by the lidar. The reconstruction needs weaker assumptions to provide fast spatial variance of wind. But such spatial variance must be compared to TKE.

The data used for this work come from the BLLAST field experiment. They have been measured the 30^{th} of June 2011 by a WindCube 200. Several days suffer from missing data. We distinguish irregular measurements, device-dependent missing values and values missing at random. Irregular measurements are filled with missing values. For large areas of missing values, it does not make sense to apply the reconstruction, hence the system is switch off until the next available measurement. For small areas of missing values, a stratagem has been set up to avoid the system to crash and side effect on the estimations. The described stratagem

is included in the patent Baehr et al. (2016). Since the effect of such stratagem has been assessed, it was not applied in this work (the data set was chosen free of missing data).

Chapter 4

Sensitivity analysis theory

Contents

4.1 Introduction

Sensitivity analysis is the quantitative study of the influence of inputs on outputs. A computer code usually depends on many settings to run. When it comes to assess the way these settings impact the output of the code, it comes to sensitivity analysis. Jacques (2005) sums up the added value of sensitivity analysis in 4 goals:

Check. For instance, when a parameter overwhelms the other, it may point a malfunction in the model (an error of units, a bypass of some function...).

Simplify. Parameters weakly influencing the code can be set to a fixed value.

Error reduction. Conversely, once the most influential inputs are known, it is easier to set a method to reduce the error induced by these settings.

Understand. Sensitivity analysis highlights objectively the effect of the settings and the interactions among them. It also requires a rationalisation of the system (to list inputs and outputs, to question their relevance). All of this brings a better understanding of the system.

Several approaches are possible, depending on the system and especially on the cost to run the code. For a review of techniques before 1994, see Hamby (1994). For a more recent review, see Iooss and Lemaître (2015).

The first idea coming in mind is usually to make the inputs move one at a time. By doing this, one is sure that the output variation is not polluted by the move of another input which cannot be attributed. Such methods are called OAT (One At a Time) (Saltelli and Annoni, 2010). They suffer from many drawbacks, the main one being the choose of the value at which the non-moving inputs are set. For example, considering the following model

$$Y = X_1^2 \sin(2\pi X_2)$$

As long as X_2 is set to an integer value, the output is 0 whatever is the value of X_1. Thus the study will conclude to a null influence of X_1. This example highlights two majors points in sensitivity analysis:

- Inputs must move all together.
- Inputs values must be non-regular and cover all the space.

Variance-based sensitivity analysis starts with the definition of Sobol indices Sobol' (1990); Sobol (1993). It is very useful in modelling (Saltelli et al., 2000), to rank inputs by order of importance (Saltelli, 2002b). When the computer code is heavy to run, a common strategy is to create a surrogate model (Marrel et al., 2009). When the code is stochastic (when some inputs are uncontrollable, for instance), the variance due to the random seed must be taken apart (Marrel et al., 2012). When the code is providing several outputs (vector output, or functional output), Sobol indices have a different form (Gamboa et al., 2014). When the inputs are not independent, generalized Sobol indices exist, but they are complicated to estimate (Chastaing et al., 2015).

Variance-based is not the only approach to perform a sensitivity analysis. The whole probability law can be considered in the importance measure. It results dependence measures

based on distribution distances (Borgonovo, 2007; Da Veiga, 2015). Their advantage is to account all effects of inputs changes, not only effects on variances.

Screening methods provide sensitivity indices inspired from the derivative of the function. They include previously evoked OAT techniques, the Morris method (Morris, 1991), the sequential bifurcation method (Bettonvil and Kleijnen, 1997). They are more adapted when the computer code has a large number of inputs.

When the model is linear, many indices based on correlation exist: Pearson correlation coefficient, standard regression coefficient (SRC), partial correlation coefficient (PCC). To get rid of the linear assumption, the indices SRC and PCC can be estimated on the ranks of the runs[1]. The model no longer has to be linear, monotonic is enough. The resulting score are standard regression rank coefficient (SRRC), partial rank correlation coefficient (PRCC) (Iooss and Lemaître, 2015; Jacques, 2005).

The model can be approached by a projection onto a basis of functions such that the sensitivity indices are given analytically. The Fourier Amplitude Sensitivity Test (FAST) gives analytical sensitivity indices from the Fourier coefficients of the model transformed by a change of variable (Cukier et al., 1978). The polynomial of chaos gives gives analytical sensitivity indices from the coefficients of the polynomials fitting the best with the model responses (Sudret, 2008). The analytical formulae of such methods reduce the amount of calculus for the sensitivity indices estimation, but they rely on a good approximation of the model with the chosen basis of functions.

The sensitivity analysis carried out in this work is variance-based. It consists in the estimation and the interpretation of Sobol indices of first and second order. This chapter gives a state-of-the-art of the variance-based sensitivity analysis. First, it is presented from a theoretical point of view (variance decomposition, definitions of Sobol indices). Then, the way of estimation is stated (three estimators of Sobol indices, uncertainty of the estimation). Eventually, the meta-modelling of the system is presented.

4.2 Variance-based sensitivity analysis

4.2.1 Notations

The computer code takes p parameters in input and deliver one output. Hence the computer code is seen as a function from $\mathbb{R}^p$ to $\mathbb{R}$

$$f : \begin{array}{ccc} \mathbb{R}^p & \to & \mathbb{R} \\ X & \mapsto & Y = f(X) \end{array}$$

[1]Given a sample of N runs of the computer code, the output vector contains N values. When this vector is sorted increasingly, the rank of a run is its place in the sorted vector.

In this section, variables X and Y will be denoted in lower case when they are deterministic and in upper case when they are random. For the indices, we stick to the notation used in Chastaing (2013) that are recalled here:

- $[\![1, p]\!] = \{1, \dots, p\}$
- I is the collection of all subset of $[\![1, p]\!]$ (thus of cardinal 2^p).
- $\mathbf{u}$ is an element of I.
- $|\cdot|$ the cardinal of the set "$\cdot$". For example, $|\mathbf{u}|$ is the number of indices in $\mathbf{u}$; and $|I|$ is the number of groups of indices in I ($|I| = 2^p$).
- For all $\mathbf{u} \in I$, it is denoted $\bar{\mathbf{u}} = [\![1, p]\!] \setminus \mathbf{u} = \{i \in [\![1, p]\!], i \notin \mathbf{u}\}$
- For all $\mathbf{u} \in I$, it is denoted $X_{\mathbf{u}} = (X_i,\, i \in \mathbf{u})$ the vector extracted from X.
- When it exists (assumed true by default), probability density function of $X_{\mathbf{u}}$ is denoted $p_{X_{\mathbf{u}}}$, for all $\mathbf{u} \in I$.

To order the elements in I, the convention is to use the lexicographic order from left to right (order of the dictionary). For instance, for $p = 3$:

$$I = \{\emptyset, \{1\}, \{2\}, \{3\}, \{2, 1\}, \{3, 1\}, \{3, 2\}, \{1, 2, 3\}\}$$

The proofs are postponed to the appendix B.2. They are adapted from (Chastaing, 2013).

4.2.2 Deterministic case (ANOVA functional decomposition)

We consider a deterministic function f with arguments range in $[0, 1]$ (it is equivalent to consider that they lie in a bounded interval, by means of a normalization).

Theorem 4.1 (ANOVA decomposition)**.** *Let $f : [0, 1]^p \to \mathbb{R}$ be an integrable function. Previous notations hold.*
Then, there exists a unique decomposition of f :

$$\begin{aligned} f(x) &= \sum_{\mathbf{u} \in I} f_{\mathbf{u}}(x_{\mathbf{u}}) \\ &= f_\emptyset + \sum_{i=1}^{p} f_i(x_i) + \sum_{1 \leqslant i < j \leqslant p} f_{ij}(x_i, x_j) + \cdots + f_{1,\dots,p}(x_1, \dots, x_p) \end{aligned} \tag{4.1}$$

such that

$$\forall \mathbf{u} \in I,\, \forall i \in \mathbf{u},\, \int_0^1 f_{\mathbf{u}}(x_{\mathbf{u}}) dx_i = 0 \tag{4.2}$$

The proof of this theorem has been first made by Sobol in Sobol (1976), using Fourier-Haar decompositions. Then a simpler version is in Sobol (1993). A proof adapted to our notations is the appendix B.2.1.

One can show the functions in the decomposition are orthogonal (proof of proposition in the appendix B.2.2).

Proposition 4.1 (Orthogonality in ANOVA)**.** *Let $f : [0,1]^p \to \mathbb{R}$ be integrable and $(f_{\mathbf{u}})_{\mathbf{u}\in I}$ the functions of its ANOVA decomposition. Then,*

$$\forall \mathbf{u}, \mathbf{v} \in I,\ \mathbf{u} \neq \mathbf{v},\ \int_{[0,1]^p} f_{\mathbf{u}}(x_{\mathbf{u}}) f_{\mathbf{v}}(x_{\mathbf{v}}) dx = 0 \tag{4.3}$$

The decomposition which has been proven is called ANOVA (ANalysis Of VAriance). It is the basis for the definition of sensitivity indices. Each terms of the decomposition is function of one group of inputs only. The effect of other inputs or other group of inputs manifests through another term in the decomposition. The uniqueness of the decomposition is of first importance. Without uniqueness, it is not possible to attribute the variance to a term of the decomposition.

4.2.3 Variance decomposition

The ANOVA decomposition is extended to the stochastic case. Let X be a random vector of law η_X. The components of X are assumed independent. Thus

$$\eta_X = \eta_{X_1} \otimes \cdots \otimes \eta_{X_p}$$

The output is the random variable $Y = f(X)$. The aim is to find a set of functions that gives the same decomposition as in the deterministic case. Such functions must have several properties, which define the space they belong. These space are Hoeffding spaces.

Definition 4.1 (Hoeffding spaces)**.**

With words *$H_{\mathbf{u}}$ contains functions of $X_{\mathbf{u}}$ for which the resulting random variable has finite variance. $H^0_{\mathbf{u}}$ is the subset of $H_{\mathbf{u}}$ for which the expectation of the resulting random variable conditioned to $X_{\mathbf{v}}$ is zero for any $\mathbf{v}$ with a cardinal smaller than $|\mathbf{u}|$.*

Formally $\forall \mathbf{u} \in I$

$$H_{\mathbf{u}} = \left\{ h_{\mathbf{u}}(X_{\mathbf{u}}),\ \mathbb{E}\left[h_{\mathbf{u}}(X_{\mathbf{u}})^2\right] < +\infty \right\} \quad \subset L^2(\mathbb{R}, \eta_{X_{\mathbf{u}}})$$

$$H^0_{\mathbf{u}} = \{ h_{\mathbf{u}}(X_{\mathbf{u}}) \in H_{\mathbf{u}},\ \forall \mathbf{v} \in I, |\mathbf{v}| < |\mathbf{u}|,\ \mathbb{E}\left[h_{\mathbf{u}}(X_{\mathbf{u}}) | X_{\mathbf{v}}\right] = 0 \}$$

By convention, $H_\emptyset$ and $H^0_\emptyset$ stand for the set of constant functions, and $\mathbb{E}\left[h_{\mathbf{u}}(X_{\mathbf{u}})|\emptyset\right] = \mathbb{E}\left[h_{\mathbf{u}}(X_{\mathbf{u}})\right]$.

The spaces $H^0_{\mathbf{u}}$, $\mathbf{u} \in I$ are orthogonal (proof in the appendix B.2.3).

Proposition 4.2 (Orthogonality of Hoeffding spaces)**.**

$$\forall\, \mathbf{u}, \mathbf{v} \in I, \mathbf{u} \neq \mathbf{v}, \quad H^0_{\mathbf{u}} \perp H^0_{\mathbf{v}} \tag{4.4}$$

With projectors, any finite variance real-valued random variable can be projected onto the spaces $H^0_{\mathbf{u}}$. The ensuing decomposition can be expressed thanks to orthogonality.

Lemme 4.1 (Hoeffding projection)**.** *Considering*

- $X_1, ..., X_p$ *independent random variables.*
- $T \in L^2(\Omega, \mathbb{R})$ *(real random variable of finite variance* $\mathbb{E}\left[T^2\right] < +\infty$*).*

Then, for all $\mathbf{u} \in I$*, the orthogonal projection of* T *in* $H^0_{\mathbf{u}}$ *in written*

$$\pi_{H^0_{\mathbf{u}}}(T) = \sum_{\mathbf{v} \subset \mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} \mathbb{E}\left[T | X_{\mathbf{v}}\right] \tag{4.5}$$

The proof is in the appendix B.2.4. This lemma allows a functional decomposition of f of the same form of ANOVA.

Theorem 4.2 (Hoeffding decomposition)**.** *Let* $Y : (\Omega, \mathcal{F}, \mathbb{P}) \to (\mathbb{R}, \mathcal{B}(\mathbb{R}))$ *et* $X : (\Omega, \mathcal{F}, \mathbb{P}) \to (\mathbb{R}^p, \mathcal{B}(\mathbb{R}^p))$ *such that* $Y = f(X)$ *with* $f : (\mathbb{R}^p, \mathcal{B}(\mathbb{R}^p)) \to (\mathbb{R}, \mathcal{B}(\mathbb{R}))$*, a measurable function. Previous notations hold.*
Under the following assumptions :

1. Y *has a finite variance (i.e.* $\mathbb{E}\left[Y^2\right] < +\infty$*).*
2. *The inputs* X_i, $i \in [\![1, p]\!]$ *are independent :* $p_X(x) = \prod_{i=1}^p p_{X_i}(x_i)$*.*

Then it exists an unique decomposition of Y *with respect to* $(X_i)_i$ *:*

$$\begin{aligned} Y &= \sum_{\mathbf{u} \in I} f_{\mathbf{u}}(X_{\mathbf{u}}) \\ &= f_\emptyset + \sum_{i=1}^p f_i(X_i) + \sum_{1 \leqslant i < j \leqslant p} f_{ij}(X_i, X_j) + \cdots + f_{1,\ldots,p}(X_1, \ldots, X_p) \end{aligned} \tag{4.6}$$

such that

$$\forall \mathbf{u} \in I,\ f_{\mathbf{u}}(X_{\mathbf{u}}) = \sum_{\mathbf{v} \subset \mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} \mathbb{E}\left[Y | X_{\mathbf{v}}\right] \tag{4.7}$$

The proof is in the appendix B.2.5. The expression of the decomposition is the same as in theorem 4.1 with the difference that the hypothesis on which it relies are weaker and suitable

to random variables. It also gives an expression of the terms in the decomposition, which is a valuable result in order to find numerical estimations.

This decomposition is used for the definition of sensitivity indices. The sensitivity of Y with respect to the inputs $(X_i)_i$ is measured by its variance. By taking the variance of the decomposition, one can have a decomposition of the variance of Y into terms which can be attributed to a group of inputs. From such decomposition, one gets the Sobol indices.

Corollary 4.1. *Under the same assumptions than the theorem 4.2,*

$$V(Y) = \sum_{\mathbf{u} \in I} V(f_{\mathbf{u}}(X_{\mathbf{u}})) = \sum_{\mathbf{u} \in I} \left(V(\mathbb{E}[Y|X_{\mathbf{u}}]) + \sum_{\mathbf{v} \subset \mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} V(\mathbb{E}[Y|X_{\mathbf{v}}]) \right) \tag{4.8}$$

The proof is in the appendix B.2.6. Given a group of inputs $\mathbf{u}$, the proportion of variance attributed to this group will be its sensitivity index. The proportion of variance attributed to a group is the corresponding term in the ANOVA decomposition.

4.2.4 Sobol indices

The most general form of Sobol indices results from the Hoeffding decomposition (theorem 4.2). As the variance $V(Y)$ can be estimated by many efficient ways, it is sometimes clearer to use only the unnormed Sobol indices. The cardinal of the group $\mathbf{u}$ is called the order of the indices. The indices ensuing directly from the corollary 4.1 (so-called "simple" or "main effect" Sobol indices) are the basis to build others, with different interpretations.

Definition 4.2 (Simple Sobol indices)**.**

With words *The simple Sobol index (or main effect Sobol index) of a group of inputs* $\mathbf{u}$ *is the proportion of the output variance attributed to this group.*

Formally $\forall \mathbf{u} \in I$, $\mathbf{u} \neq \emptyset$,

$$S_{\mathbf{u}} = \frac{V(f_{\mathbf{u}}(X_{\mathbf{u}}))}{V(Y)}$$

$$D_{\mathbf{u}} = V(f_{\mathbf{u}}(X_{\mathbf{u}}))$$

Total Sobol index is a score for the influence of the group $\mathbf{u}$ including all order of interaction. They have been introduced by Homma and Saltelli (1996).

Definition 4.3 (Total Sobol indices)**.**

With words *The total Sobol index of a group of inputs* $\mathbf{u}$ *is the proportion of the output variance attributed to this group and to all the groups in which its members are involved.*

Formally $\forall \mathbf{u} \in I,\ \mathbf{u} \neq \emptyset$,

$$S_{\mathbf{u}}^T = \sum_{\substack{\mathbf{v} \in I \\ \mathbf{v} \cap \mathbf{u} \neq \emptyset}} S_{\mathbf{v}}$$

$$D_{\mathbf{u}}^T = \sum_{\substack{\mathbf{v} \in I \\ \mathbf{v} \cap \mathbf{u} \neq \emptyset}} D_{\mathbf{v}} = V(Y)\, S_{\mathbf{u}}^T$$

Complete Sobol index is a score for the influence of the group $\mathbf{u}$ up to its maximal order of interaction.

Definition 4.4 (Complete Sobol indices)**.**

With words *The complete Sobol index of a group of inputs* $\mathbf{u}$ *is the proportion of the output variance attributed to this group and to all the subgroups in* $\mathbf{u}$*.*

Formally $\forall \mathbf{u} \in I,\ \mathbf{u} \neq \emptyset$,

$$S_{\mathbf{u}}^C = \sum_{\substack{\mathbf{v} \in I \\ \mathbf{v} \subseteq \mathbf{u}}} S_{\mathbf{v}}$$

$$D_{\mathbf{u}}^C = \sum_{\substack{\mathbf{v} \in I \\ \mathbf{v} \subseteq \mathbf{u}}} D_{\mathbf{v}} = V(Y)\, S_{\mathbf{u}}^C$$

For example, if $p = 3$ and $\mathbf{u} = \{1\}$,

$$D_1 = V(f_1(X_1)) = V(\mathbb{E}\,[Y|X_1])$$

$$D_1^T = D_1 + D_{1,2} + D_{1,3} + D_{1,2,3}$$

$$D_1^C = D_1$$

If $p = 3$ and $\mathbf{u} = \{1, 2\}$,

$$D_{1,2} = V(f_{1,2}(X_1, X_2)) = V(\mathbb{E}\,[Y|X_1, X_2]) - V(\mathbb{E}\,[Y|X_1]) - V(\mathbb{E}\,[Y|X_2])$$

$$D_{1,2}^T = D_1 + D_2 + D_{1,2} + D_{1,3} + D_{2,3} + D_{1,2,3} = V(Y) - D_3$$

$$D_{1,2}^C = D_1 + D_2 + D_{1,2} = V(\mathbb{E}\,[Y|X_1, X_2])$$

Using the analytical expression of the decomposition functions (theorem 4.2), one can write analytical expressions of such indices (proof in appendix B.2.7).

Proposition 4.3. *Let $\mathbf{u} \in I$. With the notation introduced above, simple, total and complete Sobol indices have the following expression :*

$$D_{\mathbf{u}} = V(\mathbb{E}[Y|X_{\mathbf{u}}]) + \sum_{\substack{\mathbf{v} \subset \mathbf{u} \\ \mathbf{v} \neq \mathbf{u}}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} V(\mathbb{E}[Y|X_{\mathbf{v}}]) \tag{4.9}$$

$$D^C_{\mathbf{u}} = V(\mathbb{E}[Y|X_{\mathbf{u}}]) \tag{4.10}$$

$$D^T_{\mathbf{u}} = V(Y) - V(\mathbb{E}[Y|X_{\bar{\mathbf{u}}}]) \tag{4.11}$$

For first order Sobol indices, the formula of the simple Sobol indices is much simpler since there is no second term.

$$S_i = \frac{V(\mathbb{E}[Y|X_i])}{V(Y)} \tag{4.12}$$

First order simple and complete Sobol indices are equal. Total Sobol indices reckon with all the effects of the input, including all order interactions. Thanks to proposition 4.3, they are written as follow.

$$S^T_i = 1 - \frac{V\left(\mathbb{E}\left[Y|X_{[\![1,p]\!]\setminus i}\right]\right)}{V(Y)} \tag{4.13}$$

Beside their interpretations, some properties ensue from the definitions of Sobol indices. They help to check the quality of the numerical estimations.

Proposition 4.4 (Basic properties of Sobol indices)**.**

$$\forall \mathbf{u} \in I,\ S_{\mathbf{u}} \geqslant 0 \qquad \sum_{\mathbf{u} \in I} S_{\mathbf{u}} = 1 \qquad \sum_{i=1}^{p} S_i \leqslant 1 \qquad \sum_{i=1}^{p} S^T_i = \sum_{\mathbf{u} \in I} |\mathbf{u}| S_{\mathbf{u}} \geqslant 1$$

4.2.5 When inputs are not independent

The previous theorems and definition are based on the assumptions that inputs are independent. Since in most cases the inputs values are given by pseudo-random number generator, it is reasonable to assume them independent. Nevertheless, one can imagine a computer code for which the inputs are known to be dependent. In this case, either grouping some variables can make the inputs independent (as suggested by Jacques et al. (2006)), either it is necessary to get rid of the independence assumption. But there is nothing left in the previous results when the inputs are not independent. Keeping the independence assumption is convenient, at the risk of deliberately ignore the dependence. For those who cannot take that risk, there exist solutions, more complex than the previous, that we will quote without the proofs.

In the works of Gaëlle Chastaing (Chastaing, 2013; Chastaing et al., 2012, 2015) are defined generalized Sobol indices. These indices can be interpreted even with dependent inputs and they stick to usual Sobol indices when the inputs are independent. In order to

complete this state-of-the-art, we give here the necessary hints to come out on generalized Sobol indices. The complete proofs are in the aforementioned papers.

The independence assumption is replaced by the two next assumptions:

(H1) $\eta_X \ll \Lambda$, where Λ stands for Lebesgue measure.

(H2) $\exists M \in]0,1], \forall \mathbf{u} \in [\![1,p]\!],\ p_X \geqslant M p_{X_{\mathbf{u}}} p_{X_{\bar{\mathbf{u}}}}$

The construction of generalized sensitivity indices rely on a functional decomposition hierarchically orthogonal. *Hierarchically orthogonal* signifies the orthogonality holds for some groups of functions organized hierarchically. The adjective hierarchical refers to the organisation of indices.

Definition 4.5 (Hierarchical collection of indices)**.**

With words *A collection of indices $T \subset I$ is said hierarchical when all sub-groups of indices taken in an element of T is in T.*

Formally

$$\forall \mathbf{u} \in T, \forall \mathbf{v} \subset \mathbf{u}, \mathbf{v} \in T$$

For example, if the code has $p = 3$ inputs. We have seen that

$$I = \{\emptyset, \{1\}, \{2\}, \{3\}, \{2,1\}, \{3,1\}, \{3,2\}, \{1,2,3\}\}$$

Then the collection of indices $T = \{\{1\}, \{2\}, \{2,1\}\}$ is hierarchical. Ideed, if one takes $\mathbf{u} = \{1,2\}$, then $\{1\}, \{2\}, \{2,1\} \subset \mathbf{u}$, and $\{1\}, \{2\}, \{2,1\} \in T$.
Similarly, if one takes $\mathbf{u} = \{1\}$, then only $\{1\} \subset \mathbf{u}$, and $\{1\} \in T$

Lemme 4.2. *Let $T \subset I$ be a hierarchical collection of indices. Under the hypothesis (H1) et (H2).*
It is denoted $\delta = 1 - \sqrt{1-M} \in]0,1]$. Then, $\forall \mathbf{u} \in T$, $\forall h_{\mathbf{u}} \in H^0_{\mathbf{u}}$,

$$\mathbb{E}\left[\left(\sum_{\mathbf{u}\in T} h_{\mathbf{u}}(X)\right)^2\right] \geqslant \delta^{|T|-1} \sum_{\mathbf{u}\in T} \mathbb{E}\left[h_{\mathbf{u}}(X)^2\right] \tag{4.14}$$

Theorem 4.3 (Stone decomposition)**.** *Let f be a function in $L^2(\mathbb{R}^p, \eta_X)$. Under the hypothesis (H1) et (H2).*
Then, it exists a unique family of functions $(f_{\mathbf{u}})_{\mathbf{u}\in I}$ such that $\forall \mathbf{u} \in I$, $f_{\mathbf{u}} \in H^0_{\mathbf{u}}$ and which verifies

$$\begin{aligned} f(X) &= \sum_{\mathbf{u}\in I} f_{\mathbf{u}}(X_{\mathbf{u}}) \\ &= f_{\emptyset} + \sum_{i=1}^{p} f_i(X_i) + \sum_{1\leqslant i<j\leqslant p} f_{ij}(X_i, X_j) + \cdots + f_{1,\ldots,p}(X_1, \ldots, X_p) \end{aligned} \tag{4.15}$$

Definition 4.6 (Generalized Sobol indices)**.**

With words *The generalized Sobol index of a group of inputs $\mathbf{u}$ is the proportion of variance attributed to this group of inputs and to the inputs on which they are dependent.*

Formally

$$\forall \mathbf{u} \in I,\ S_{\mathbf{u}} = \frac{1}{V(Y)}\left(V(f_{\mathbf{u}}(X_{\mathbf{u}})) + \sum_{\substack{\mathbf{u},\mathbf{v}\in I \\ \mathbf{u}\cap\mathbf{v}\neq\mathbf{u},\mathbf{v}}} \operatorname{cov}(f_{\mathbf{u}}(X_{\mathbf{u}}), f_{\mathbf{v}}(X_{\mathbf{v}})) \right)$$

One can notice that despite the formula has not changed, the theorem no longer gives an expression for the decomposition functions $f_{\mathbf{u}}$. It only states their existence and uniqueness (which is already a good advance). It has for consequence that the Sobol indices can no longer be estimated by approaching conditional expectations.

$$S_{\mathbf{u}} = \underbrace{\frac{V(f_{\mathbf{u}}(X_{\mathbf{u}}))}{V(Y)}}_{VS_{\mathbf{u}}} + \underbrace{\frac{\sum_{\substack{\mathbf{u},\mathbf{v}\in I \\ \mathbf{u}\cap\mathbf{v}\neq\mathbf{u},\mathbf{v}}} \operatorname{cov}(f_{\mathbf{u}}(X_{\mathbf{u}}), f_{\mathbf{v}}(X_{\mathbf{v}}))}{V(Y)}}_{CoS_{\mathbf{u}}} \tag{4.16}$$

The generalized indices defined by equation (4.16) are composed of two terms with different meanings: on one hand is the variance attributed to the group $\mathbf{u}$ ($VS_{\mathbf{u}}$), on the other hand is the variance attributed to the inputs depending on the inputs in the group $\mathbf{u}$ ($CoS_{\mathbf{u}}$).

4.3 Estimation of sensitivity indices

4.3.1 Notations

In this section we will focus on the estimation of the Sobol index of any group of indices $\mathbf{u} \in I$ (same notations as previously hold). In addition, the notations relative to estimation (samples) are used for this section :

- $X = (X_1, ..., X_p)$ is the vector of random inputs (inputs are assumed independent).
- $Z = (Z_1, ..., Z_p)$ is an independent copy of X.
- $Y = f(X) = f(X_1, ..., X_p)$ is the output (also random).
- $X_{\mathbf{u}} = (X_i)_{i \in \mathbf{u}}$ is the vector of random inputs in $\mathbf{u}$.
- $Z_{\mathbf{u}} = (Z_i)_{i \in \mathbf{u}}$ is an independent copy of $X_{\mathbf{u}}$.
- $Y_{\mathbf{u}} = f(Z_{\mathbf{u}}, X_{\bar{\mathbf{u}}})$ is the output when the inputs in $\mathbf{u}$ are taken from another realisation. For example if $\mathbf{u} = \{i\}$ (indices in $\mathbf{u}$ highlighted in blue):

$$Y_i = f(X_1, ..., Z_i, ..., X_p)$$

  For example if $\mathbf{u} = \{1, 2, p\}$ (indices in $\mathbf{u}$ highlighted in blue):

$$Y_{\{1,2,p\}} = f(Z_1, Z_2, X_3, ..., X_{p-1}, Z_p)$$

When it comes to estimation, random variables are replaced by a sample. The aforementioned notations are slighly modified to separate exact quantities and approached quantities.

- m is the size of the sample. $e \in [\![1, m]\!]$ is the counter for the elements in the sample.
- $\mathbf{X}$ is the sample of m input vectors.

$$\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_p) = \begin{pmatrix} \mathbf{x}^1 \\ \vdots \\ \mathbf{x}^m \end{pmatrix} = \begin{pmatrix} x_1^1 & \dots & x_p^1 \\ \vdots & & \vdots \\ x_1^m & \dots & x_p^m \end{pmatrix}$$

- $\mathbf{Z}$ is an independent copy of $\mathbf{X}$.
- $y^e = f(\mathbf{x}^e)$ is the output of the experiment e.
- $\mathbf{Y} = f(\mathbf{X})$ is the output vector for all experiments.
- $\mathbf{X}_{\mathbf{u}} = (\mathbf{x}_i)_{i \in \mathbf{u}}$ is the sub-matrix of $\mathbf{X}$ with only the inputs in $\mathbf{u}$. Same for $\mathbf{Z}_{\mathbf{u}}$.
- $\mathbf{Y}_{\mathbf{u}} = f(\mathbf{Z}_{\mathbf{u}}, \mathbf{X}_{\bar{\mathbf{u}}})$ is the output vector resulting from the m experiments with the inputs in $\mathbf{u}$ are taken from another sample. For example if $\mathbf{u} = \{i\}$ and if $\mathbf{u} = \{i, j\}$ (indices in $\mathbf{u}$ highlighted in blue):

$$\mathbf{Y}_i = \begin{pmatrix} f(x_1^1, ..., z_i^1, ..., x_p^1) \\ \vdots \\ f(x_1^m, ..., z_i^m, ..., x_p^m) \end{pmatrix} \qquad \mathbf{Y}_{i,j} = \begin{pmatrix} f(x_1^1, ..., z_i^1, ..., z_j^1, ..., x_p^1) \\ \vdots \\ f(x_1^m, ..., z_i^m, ..., z_j^m, ..., x_p^m) \end{pmatrix}$$

First, Sobol indices estimators are written theoretically (in terms of random variables). Then, practical estimators are derived from them.

4.3.2 Some estimators for Sobol indices

In the previous section, the theoretical definition of Sobol indices were given (simple, total and complete Sobol indices). Now we focus on the numerical estimation of them. Only the complete Sobol index $D_{\mathbf{u}}^C$ is considered because the others can be derived from them, thanks to the proposition 4.3. From Janon et al. (2014) (lemma 1.2) it is known that:

$$V(\mathbb{E}[Y|X_{\mathbf{u}}]) = \text{cov}(Y, Y_{\mathbf{u}}) \tag{4.17}$$

It is important to notice that Y and $Y_{\mathbf{u}}$ have the same law. Moreover, since the covariance is symmetric, they play exchangeable roles. From the definition of the covariance, one can derive expressions suitable to estimation.

$$D_{\mathbf{u}}^1 = \mathbb{E}[f(X)f(Z_{\mathbf{u}}, X_{\bar{\mathbf{u}}})] - \mathbb{E}[f(X)]^2 \tag{4.18}$$

$$D_{\mathbf{u}}^2 = \mathbb{E}\left[(f(X) - f(Z_{\mathbf{u}}, X_{\bar{\mathbf{u}}}))^2\right] \tag{4.19}$$

$$D_{\mathbf{u}}^3 = \mathbb{E}[f(X)(f(Z_{\mathbf{u}}, X_{\bar{\mathbf{u}}}) - f(Z))] \tag{4.20}$$

Sobol (2001) compares the two estimators (4.21) and (4.22), based respectively on the first writtings of the covariance $D_{\mathbf{u}}^1$ and $D_{\mathbf{u}}^2$. He shows that $\widehat{D_{\mathbf{u}}}^2$ has a smaller variance to estimate total Sobol indices, while $\widehat{D_{\mathbf{u}}}^1$ has a smaller variance to estimate main effect Sobol indices. Moreover, $\widehat{D_{\mathbf{u}}}^2$ is always positive, which avoid the estimation to be negative when indices are small. Saltelli et al. (2010) makes a broader comparison focused on the estimation of first order Sobol indices, and proposes the Monte Carlo estimator (4.23). Owen (2013) also makes a comparison of several estimation strategies and proposes a new estimator with the concern of improving the estimation of small Sobol indices. The novelty in Owen's estimator is to use a third copy of X.

$$\widehat{D_{\mathbf{u}}}^1 = \frac{1}{m}\sum_{e=1}^{m} f(\mathbf{x}^e) f(\mathbf{z}_{\mathbf{u}}^e, \mathbf{x}_{\bar{\mathbf{u}}}) - \left(\frac{1}{m}\sum_{e=1}^{m} f(\mathbf{x}^e)\right)^2 \tag{4.21}$$

$$\widehat{D_{\mathbf{u}}}^2 = \frac{1}{m}\sum_{e=1}^{m} (f(\mathbf{x}^e) - f(\mathbf{z}_{\mathbf{u}}^e, \mathbf{x}_{\bar{\mathbf{u}}}^e))^2 \tag{4.22}$$

$$\widehat{D_{\mathbf{u}}}^3 = \frac{1}{m}\sum_{e=1}^{m} f(\mathbf{x}^e)(f(\mathbf{z}_{\mathbf{u}}^e, \mathbf{x}_{\bar{\mathbf{u}}}^e) - f(\mathbf{z}^e)) \tag{4.23}$$

For this work, Sobol indices have been computed with the opensource Python package `SALib`. Only first order (main and total effect) and second order Sobol indices are calculated, because it is enough to describe direct, pairwise and remaining influence. It uses the estimator from (Saltelli et al., 2010) for the first order indices (main and total effect). For the second order Sobol indices, the estimator from (Saltelli, 2002a) is used. A summary of estimators

Estimator formula	Reference
$\widehat{D_i} = \frac{1}{m}\sum_{e=1}^{m} f(\mathbf{x}^e)(f(\mathbf{z}_i^e, \mathbf{x}_{\tilde{i}}^e) - f(\mathbf{z}^e))$	(Saltelli et al., 2010) eq. (16)
$\widehat{D_i^T} = \frac{1}{2m}\sum_{e=1}^{m} (f(\mathbf{z}_i^e, \mathbf{x}_{\tilde{i}}^e) - f(\mathbf{x}^e))^2$	(Saltelli et al., 2010) eq. (19)
$\widehat{D_{ij}} = \frac{1}{m}\sum_{e=1}^{m} \left(f(\mathbf{z}_i^e, \mathbf{x}_{\tilde{i}}^e) f(\mathbf{z}_j^e, \mathbf{x}_{\tilde{j}}^e) - f(\mathbf{x}^e) f(\mathbf{z}^e)\right) - \widehat{D_i} - \widehat{D_j}$	(Saltelli, 2002a)

Table 4.1 – Recap of Sobol index estimates calculated in this work.

giving the results is in the rable 4.1.

4.3.3 Uncertainty on the estimation

From the definition of Sobol indices estimation, the combination of the central limit theorem and the Delta method gives asymptotic normality (see Janon (2012), chapter 3 and the associated paper Janon et al. (2014)).

Proposition 4.5 (Uncertainty on Sobol indices estimation)**.** *If $S_{\mathbf{u}}$ is the Sobol index related to the group* $\mathbf{u}$ *(definition* **??***) and $\hat{S}_{\mathbf{u}}$ is its estimator (equation 4.21), then*

$$\sqrt{m}(S_{\mathbf{u}} - \widehat{S}_{\mathbf{u}}^1) \xrightarrow[m\to\infty]{\mathcal{L}} \mathcal{N}(0, \sigma_S^2) \tag{4.24}$$

with

$$\sigma_S^2 = \frac{V\big((Y - \mathbb{E}[Y])\big[(Y^{\mathbf{u}} - \mathbb{E}[Y]) - S_{\mathbf{u}}(Y - \mathbb{E}[Y])\big]\big)}{V(Y)^2} \tag{4.25}$$

Although normality is a handy property, it has its drawbacks here. The property $S_{\mathbf{u}} \geqslant 0$ does not hold for the estimator $\widehat{S_{\mathbf{u}}}^1$. For small Sobol indices especially, the estimation can be negative, which weakens the interpretation. For example, $\widehat{S_{\mathbf{u}}}^2$ will not give negative estimation, but will not have such a theorem to describe its uncertainty.

In the results (see chapter 6) the confidence intervals are obtained by brute force estimation, like Jacques (2005) page 80. The estimation is repeated 100 times, which gives a sample of estimates. Such sample is assumed Gaussian and its variance is used to calculate the 95% confidence interval displayed in the figures of the chapter 6.

4.3.4 Few words about generalized Sobol indices

In the case of p dependent input random variables, the theorem 4.3 gives existence and uniqueness of 2^p functions belonging to hierarchically orthogonal spaces that allow the decomposition of the output variance into input-related terms. However, the theorem does not provide any expression for such functions. The only things known is that they exist, they are unique and they live in Hoeffding spaces. But generalized Sobol indices are expressed with these functions. To estimate them, a different strategy than previously is necessary.

In the article (Chastaing et al., 2015) a method is suggested for the numerical estimation of generalized Sobol indices. This method consist in approaching the Hoeffding spaces by finite dimensional spaces. To achieve this, we start from a basis of parametric functions (polynoms, splines, harmonics...) which is orthogonalized with a variant of Gram-Schmidt method suitable to hierarchical orthogonality. Once this basis is built, the decomposition functions are expressed as a linear combination of the basis functions. Such fitting can be done with least squares minimization. Next, generalized Sobol indices are estimated using the formulae which define them. A smart choice for the basis functions would be some functions for which the Sobol indices are given analytically from their parameters, like with polynomials of chaos.

4.4 Meta-modelling of the system

Meta-models or surrogate models are emulators of the system. Given a sample of responses, they mimic the system. Instead of using a full and complex system, often costly to run, meta-models need only a set of couple inputs-outputs calculated once and for all. Thus meta-models are a way to by-pass high computation costs. To estimate Sobol indices, computation cost is often crippling. A common strategy in sensitivity analysis is to build first a meta-model and then to use the meta-model to estimate the Sobol indices. The cost of computation of the complete system is replaced by the cost of computation of the meta-model, much cheaper. This is what has been done for this work. The drawback of meta-modelling is to introduce one more layer of approximation in the estimation. This section aims to present how the meta-modelling has been carried out: by simple kriging with Gaussian isotropic variogram and small observation noise. The meta-model was created with the meta-model `sklearn` (Pedregosa et al., 2011).

4.4.1 Simple kriging

The surrogate model is obtained by kriging with Gaussian process. This technique is often used for spatial analysis. The aim is to approach a function of space f with a Gaussian stochastic process (see def. A.14). Only intrinsic processes (see def. A.17) are considered. The target function f is written as:

$$f : \begin{array}{ccc} \mathbb{R}^p & \to & \mathbb{R} \\ \mathbf{x} & \mapsto & f(\mathbf{x}) \end{array} \tag{4.26}$$

This target function is assumed to be an intrinsic stochastic process $Y_{\mathbf{x}} = f(\mathbf{x})$ and it is known only on several sites: $f(\mathbf{x}^1), ..., f(\mathbf{x}^m)$. The point where the target function is known are gathered in a matrix $\mathbf{X}$ where each line is a single observation site (numbered with superscripts) and each column is a given coordinate of space for all sites (numbered with underscripts):

$$\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_p) = \begin{pmatrix} \mathbf{x}^1 \\ \vdots \\ \mathbf{x}^m \end{pmatrix} = \begin{pmatrix} x_1^1 & \dots & x_p^1 \\ \vdots & & \vdots \\ x_1^m & \dots & x_p^m \end{pmatrix} \tag{4.27}$$

We look for a linear estimator $\hat{Y}_{\mathbf{x}}$ with features fitting the best with the data. The linearity implies the following form:

$$\hat{Y}_{\mathbf{x}} = \lambda_0 + \sum_{j=1}^{m} \lambda_j(\mathbf{x}) f(\mathbf{x}^j) \tag{4.28}$$

As for the Gaussian linear model, we impose conditions on the residuals: it must be of mean zero and of minimal variance. But conversely to the Gaussian linear model, the residuals are not supposed independent: assumption are made on the target stochastic process instead.

The target stochastic process is at least intrinsic an two of its features are taken *a priori*: the mean function and the variogram. Then, the coefficients $\lambda_0, ..., \lambda_m$ are chosen to ensure that

1. $\mathbb{E}\left[\hat{Y}_{\mathbf{x}} - Y_{\mathbf{x}}\right] = 0$
2. $V\left(\hat{Y}_{\mathbf{x}} - Y_{\mathbf{x}}\right)$ is minimum.

The first feature to choose is the mean function because it sets the type of kriging:

- $\mathbb{E}[Y_{\mathbf{x}}] = 0$ implies simple kriging.
- $\mathbb{E}[Y_{\mathbf{x}}] = a$, with a an unknown constant, implies ordinary kriging.
- $\mathbb{E}[Y_{\mathbf{x}}] = \sum_k \alpha_k g_k(\mathbf{x})$, with g_k parametric functions and α_k unknown constants, implies universal kriging.

Simple kriging is much simpler than other method and still have good performances even when the target function is not centred on zero. In simple kriging, the minimization of the variance is without constraint. It is worth to try the simple kriging in a first place. In this work we sticked to simple kriging.

As a result of the variance minimisation, the simple kriging estimator is written:

$$\hat{Y}_{\mathbf{x}} = \mathbf{c}(\mathbf{x})^T \mathbf{C}(\mathbf{X})^{-1} \mathbf{Y} \tag{4.29}$$

with $\mathbf{c}(\mathbf{x}) = \left(\text{cov}(f(\mathbf{x}), f(\mathbf{x}^1)), ..., \text{cov}(f(\mathbf{x}), f(\mathbf{x}^m))\right)^T$,
$\mathbf{C}(\mathbf{X}) = \begin{pmatrix} \text{cov}(f(\mathbf{x}^1), f(\mathbf{x}^1)) & ... & \text{cov}(f(\mathbf{x}^1), f(\mathbf{x}^m)) \\ \vdots & & \vdots \\ \text{cov}(f(\mathbf{x}^m), f(\mathbf{x}^1)) & ... & \text{cov}(f(\mathbf{x}^m), f(\mathbf{x}^m)) \end{pmatrix}$
and $\mathbf{Y} = \left(f(\mathbf{x}^1), ..., f(\mathbf{x}^m)\right)^T$.

4.4.2 Gaussian isotropic variogram

The second feature to optimize is the variogram. Indeed, as $Y_{\mathbf{x}}$ is an intrinsic process, for any couple of points $(\mathbf{x}, \mathbf{y})$, the variogram is defined as

$$\gamma(\mathbf{y} - \mathbf{x}) = \frac{1}{2}\mathbb{E}\left[(Y_{\mathbf{x}} - Y_{\mathbf{y}})^2\right] \tag{4.30}$$

The type of the variogram is specified and its parameters can be estimated to fit with the data. It will be said **isotropic** when it depends only on $\|\mathbf{y} - \mathbf{x}\|$ instead of the full vector. The variogram is then given by a function of the real number $s = \|\mathbf{y} - \mathbf{x}\|$. For example, a Gaussian variogram would be of the form

$$\gamma(s) = \sigma^2 \left(1 - e^{-\frac{s^2}{2l^2}}\right) \tag{4.31}$$

with parameters σ and l subject to fitting.

The different components of $\mathbf{x}$ might be not comparable (because of units for example). In this case, the length l might not be the same for all components (we assume the sill variance σ^2 is). If we denote $\mathbf{h} = \mathbf{y} - \mathbf{x}$ and $\mathbf{k} = (1/l_1, ..., 1/l_p)$, the Gaussian variogram might be written $\gamma(\mathbf{h}) = \sigma^2 \left(1 - e^{-\frac{1}{2}\mathbf{k}\cdot\mathbf{h}}\right)$. Such kernel is **anisotropic**. In this case, there are p length scales to estimate instead of one.

In this work, the data have been normalized before doing the kriging. Each coordinate i has been modified as follows:

$$\tilde{\mathbf{x}}_i = \frac{\mathbf{x}_i - \min_j(x_i^j)}{\max_j(x_i^j) - \min_j(x_i^j)} \tag{4.32}$$

The matrix $\tilde{\mathbf{X}} = (\tilde{\mathbf{x}}_1, ..., \tilde{\mathbf{x}}_p)$ is used instead of $\mathbf{X}$. Then the kriging performed here is isotropic.

The other advantage to make such transform is to enable a visualisation of the variogram. The variographic swarm method (Ginsbourger (2009) section 3.2.3) aims to estimate the variogram with the help of the set $\mathcal{V}(\mathbf{X})$:

$$\mathcal{V}(\mathbf{X}) = \left\{ \left(\|\mathbf{x}^i - \mathbf{x}^j\|, \frac{1}{2}(f(\mathbf{x}^i) - f(\mathbf{x}^j))^2 \right), (i,j) \in [\![1, m]\!]^2 \right\} \tag{4.33}$$

The aim is to provide an estimation of $\frac{1}{2}\mathbb{E}\left[(Y_{\mathbf{x}} - Y_{\mathbf{y}})^2\right]$ for several values of $\|\mathbf{y} - \mathbf{x}\|$. The sites where the target is known are gathered pairwise: it gives $m(m-1)/2$ distinct pairs, therefore $m(m-1)/2$ values of gap $s = \|\mathbf{y} - \mathbf{x}\|$ and $m(m-1)/2$ values of squared increments $(f(\mathbf{x}^i) - f(\mathbf{x}^j))^2$. Considering a regular subdivision $(s_1 = 0, s_2, ..., s_{K+1} = \max_{(i,j)} \|\mathbf{x}^i - \mathbf{x}^j\|)$, one gets an estimation of the variogram in K points: for any $k \in [\![1, K]\!]$,

$$\left(\frac{s_k + s_{k+1}}{2} \ , \ \frac{\displaystyle\sum_{1 \leqslant i < j \leqslant m} \frac{1}{2}(f(\mathbf{x}^i) - f(\mathbf{x}^j))^2 \mathbf{1}_{\{s_k \leqslant \|\mathbf{x}^i - \mathbf{x}^j\| < s_{k+1}\}}}{\displaystyle\sum_{1 \leqslant i < j \leqslant m} \mathbf{1}_{\{s_k \leqslant \|\mathbf{x}^i - \mathbf{x}^j\| < s_{k+1}\}}} \right) \tag{4.34}$$

Such visualization is illustrated on one of the outputs defined at the next chapter (the wind spectrum slope). The set $\mathcal{V}(\tilde{\mathbf{X}})$ is represented by the black dots in the figure 4.1. In abscissa are the gap values $s = \|\mathbf{y} - \mathbf{x}\|$, in ordinate are the squared increments $\frac{1}{2}(f(\mathbf{x}^i) - f(\mathbf{x}^j))^2$. The red crosses are the average within a subdivision: they draw an empirical variogram. To ensure the existence of mean and variance for the estimator, the variogram must be a positive-semidefinite function. There is no reason for the empirical variogram to be admissible. Thus we look for the closest admissible variogram. This empirical variogram is reported in figure 4.2 and compared with the Gaussian variogram obtained after fitting the parameters to the data (by maximizing the likelihood). This figures proves that the choice of a Gaussian variogram is relevant for this output.

On the other hand, for some output the Gaussian form is not the most relevant. For

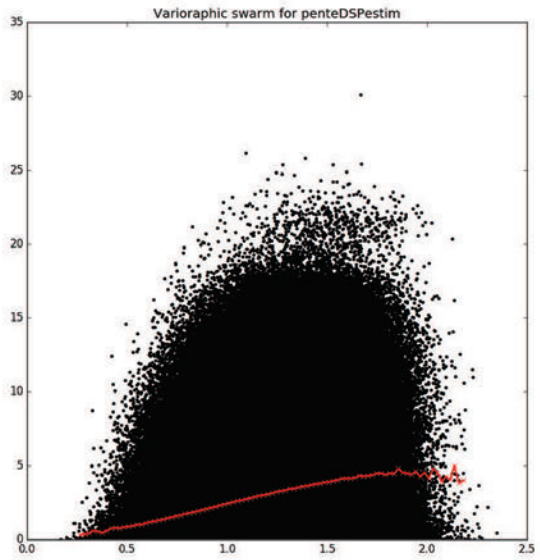


Figure 4.1 – Variographic swarm and empirical variogram for the output *b*.

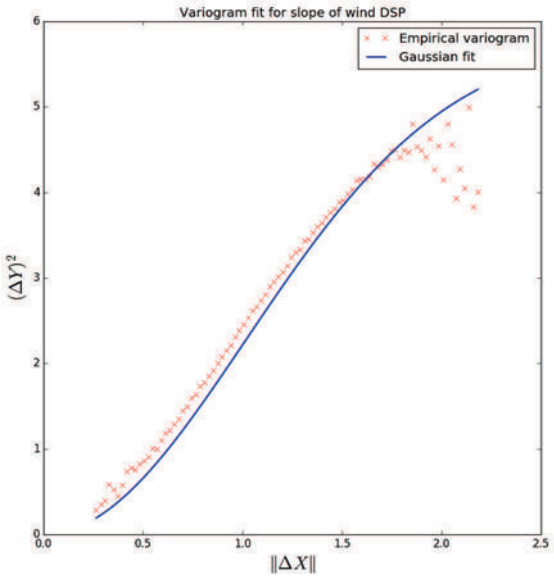


Figure 4.2 – Empirical variogram and fitted Gaussian variogram.

example, figure 4.3 shows the empirical variogram for the wind RMSE output (see chapter 5, def. 5.1 for exact definition) with different variogram types (namely Gaussian, Matérn 3/2 and 5/2, rational quadratic and exponential). The one with the best accordance is a power function. But such variogram is possible only for non-stationary processes and it is not implemented in the Python package used for the study. Therefore the Gaussian variogram was also used for this input, in the absence of any better available. The situation was similar for the other outputs, shown in figures 4.4 to 4.6.

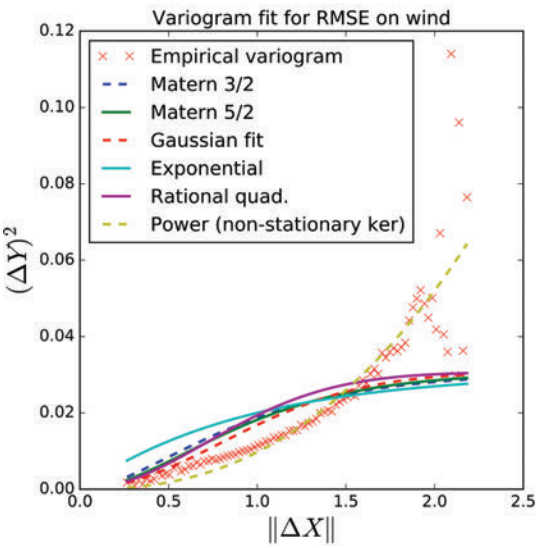


Figure 4.3 – Empirical variogram and Gaussian variogram fitting for the output r_V.

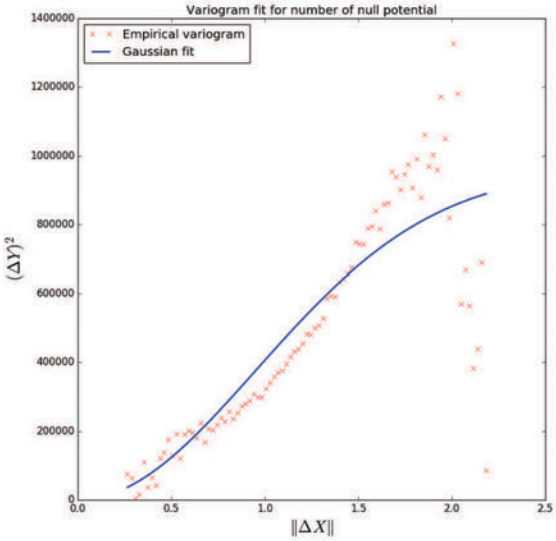


Figure 4.4 – Empirical variogram and Gaussian variogram fitting for the output N_{G0}.

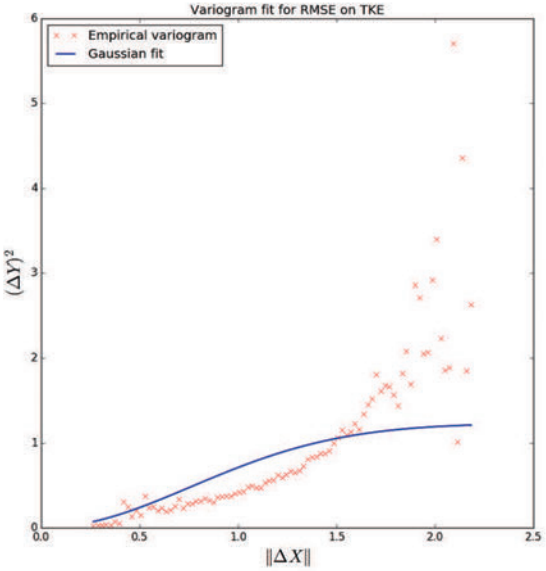


Figure 4.5 – Empirical variogram and Gaussian variogram fitting for the output r_k.

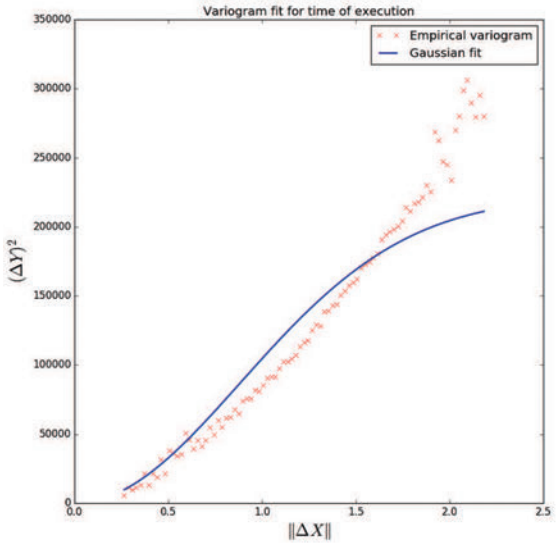


Figure 4.6 – Empirical variogram and Gaussian variogram fitting for the output T_{exe}.

4.4.3 Nugget effect and noisy observations

The so-called "**nugget effect**" is a discontinuity of the variogram in zero. All variograms satisfy $\gamma(0) = 0$. Without nugget effect, they also satisfy $\lim_{s\to 0} \gamma(s) = 0$. One talks about nugget effect when this limit is not zero. The consequence is that the the kriging estimator is discontinuous on observation sites: as close as possible of the observation sites, it may give a value different from observation. But the kriging estimator is still an exact interpolator: the equation (4.29) is modified into:

$$\hat{Y}_{\mathbf{x}} = (\mathbf{c}(\mathbf{x}) + \sigma_0^2 \mathbf{1}_{\mathbf{X}}(\mathbf{x}))^T (\mathbf{C}(\mathbf{X}) + \sigma_0^2 \mathbf{I})^{-1} \mathbf{Y} \tag{4.35}$$

where $\mathbf{1}_{\mathbf{X}}(\mathbf{x}) = 1$ if $\mathbf{x} \in \mathbf{X}$, 0 else; and $\mathbf{I}$ the $m \times m$ identity matrix. When $\mathbf{x} = \mathbf{x}^1$ for example, the terms $(\mathbf{c}(\mathbf{x}^1) + \sigma_0^2 \delta_{\mathbf{X}}(\mathbf{x}^1))$ and $(\mathbf{C}(\mathbf{X}) + \sigma_0^2 \mathbf{I})^{-1}$ cancel each other and the result is $\hat{Y}_{\mathbf{x}^1} = f(\mathbf{x}^1)$.

To take into account some **noise** in the data, the nugget is added only on the observation points: the equation (4.29) is modified into:

$$\hat{Y}_{\mathbf{x}} = \mathbf{c}(\mathbf{x})^T (\mathbf{C}(\mathbf{X}) + \sigma_0^2 \mathbf{I})^{-1} \mathbf{Y} \tag{4.36}$$

When $\mathbf{x} = \mathbf{x}^1$ for example, the terms $\mathbf{c}(\mathbf{x}^1)$ and $(\mathbf{C}(\mathbf{X}) + \sigma_0^2 \mathbf{I})^{-1}$ no longer cancel each other and the result is $\hat{Y}_{\mathbf{x}^1} \neq f(\mathbf{x}^1)$. This distinction is clearly stated in the `DiceKriging` package documentation (Roustant et al. (2012), pages 7-8).

In the `sklearn` package, the documentation is not as specific but it appears that nugget and observation noise are not distinguished and only observation noise is considered. For example, in figure 4.7 is shown an example of 1-dimensional kriging with (right) and without (left) observation noise. One can see that the estimation without observation noise equals exactly the observed values while it is sometimes different when some observation noise is added. The interpolator is not exact, thus it is truly an observation noise and not a nugget effect.

Indeed the filtering method provides an estimation which may vary from one run to the next because the sample size is finite. Thus an observation noise is relevant in our system. To give a value for the noise level, the same run[2] was repeated 1000 times and the standard deviation of the output was taken as noise value.

[2]This run was with all parameters at their nominal value, given in the table 7.2, page 203.

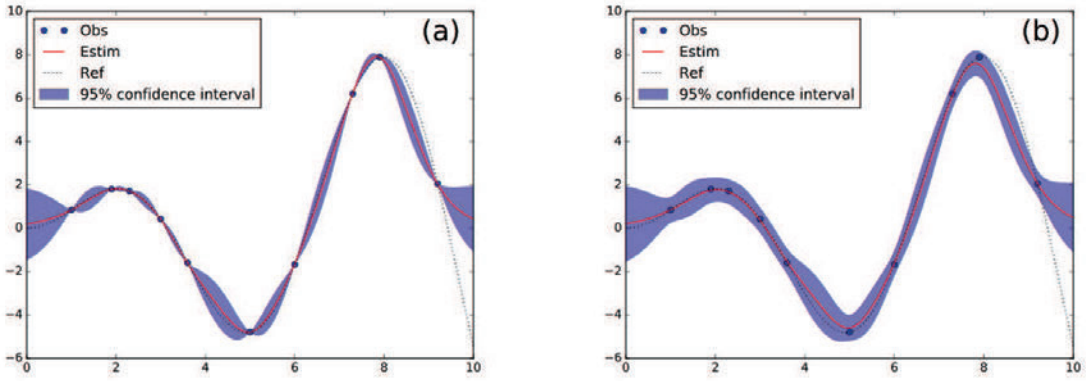


Figure 4.7 – Example of 1-dimensional kriging of the function $x \mapsto x\sin(x)$ with a Matèrn 5/2 variogram. Blue dots are the observations, dotted black line is the target function, red solid line is the kriging estimation, shaded area is the 95% confidence interval.
(a) without nugget effect. (b) with a noise value of 0.05.

4.4.4 Performances of the meta-models

The surrogate model was evaluated by cross-validation (see section 8.3.1 for an introduction to cross-validation). The original sample was split randomly in a training part (80% of the sample) and a testing part (20% of the sample). The training part is used to fit the meta-model, then the fitted meta-model makes its prediction on the testing part. Like so, the error of the meta-model can be estimated. To lessen the subjectivity of the random split, the K-fold method is used with $K = 5$: the sample is split randomly in 5 equally-sized parts and they are taken one after the other as the testing part. For each *fold*[3], the determination coefficient R^2 is estimated:

$$R^2 = 1 - \frac{\sum_{i=1}^{N_{test}} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{N_{test}} (y_i - \bar{y})^2} \tag{4.37}$$

where N_{test} the size of the testing sample, y_i the observed value given in the testing sample, $\hat{y}_i$ the estimation with the meta-model and $\bar{y} = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} y_i$ the estimated average on the testing sample.

The R^2 score has for maximum value 1, that would be reached for a model making the same prediction as the testing sample. For a model predicting the average value $\hat{y}$ everytime, the R^2 score would be 0. However, the lowest bound for R^2 is $-\infty$ as the model can always be made worse. In summary, the highest is R^2, the better is the model.

The K-fold cross-validation was repeated for each output of the system (see chapter 5

[3]The K-fold method can be visualized like if one of the K parts of the sample (the testing part) is folded against the others. This operation is repeated K times and can be called a fold (shortcut).

for their definition). The table 4.2 gives the R^2 averaged over the 5 folds. One can see that the meta-models for b (wind spectrum slope) and T_{exe} (execution time) are quite good. On the other hand, the meta-models for N_{G0} (number of null potential), r_k (TKE error) and r_V (wind error) are of lower quality. It means that the predictions of wind spectrum slope and execution time are more trustworthy than the predictions of number of null potential, TKE error and wind error. However, some graphics presented at the chapter 6 are independent from meta-models (cobweb plots) and strengthen the given conclusion.

Output	N_{G0}	b	r_k	r_V	T_{exe}
Average R^2	0.6418	0.8512	0.6204	0.6269	0.8574

Table 4.2 – Results of the evaluation of the meta-model by K-fold cross-validation ($K = 5$).

4.5 Conclusion

This chapter is an introduction to the variance-based sensitivity analysis carried out in this thesis. Although it is not the only way to perform a sensitivity analysis, variance-based techniques has the advantage to be general (finite variance apart, no assumption on the computer code is made), with many numerical estimations available. The sensitivity indices (so-called Sobol indices) are based on the decomposition of the output variance. First, the decomposition is functional. The deterministic version is given as an introduction. For the stochastic version, the output is seen as a random variable projected onto Hoeffding spaces. Hoeffding spaces are orthogonal, which ensures the decomposition is unique. In the end the variance of the output is written as a sum of terms which be attributed to a single group of inputs. Such terms are the unnormalized version of the simple Sobol indices. From simple Sobol indices, we define total Sobol indices (reckoning with all contributions of the group members) and complete Sobol indices (reckoning with all contributions of lower order). In this thesis, only first order simple Sobol indices, first order total Sobol indices and second order simple Sobol indices are computed. The corresponding numerical estimators are precised in the table 4.1. To by-pass the computing cost of running the model many times, a Gaussian process meta-model is used. It is built by simple kriging with a Gaussian variogram. All the calculus have been made with Python opensource packages: `sklearn` for Gaussian process kriging, `SALib` for Sobol indices estimation. The results of the sensitivity analysis are presented at the chapter 6.

Part II

Applications and results

Chapter 5

Reconstruction of turbulent medium

Contents

5.1 Introduction

The purpose of turbulent medium reconstruction (shortened in *reconstruction method*) is to filter fast measurements of wind in a turbulent fluid and provide estimates of turbulence as fast as the observation. The application case is the measurement of the wind in the atmospheric boundary layer with a multiple point remote sensor. The principle is to represent the fluid by many fluid particles (which give also a Monte Carlo representation of the wind probability

law), to propagate these particles with a stochastic Lagrangian model and to correct their speed according to the observation with a filtering algorithm.

At the chapter 1, we have introduced the stochastic Lagrangian model used here. At the chapter 2, we reviewed some parts of the filtering theory and explained the choice of the algorithm. In chapter 3, we saw the principle of measurement of a Doppler lidar. Turbulent medium reconstruction is at the crossroad of these domains, as summarized by figure 5.1. This section aims to link the previous elements to build the reconstruction method algorithm.

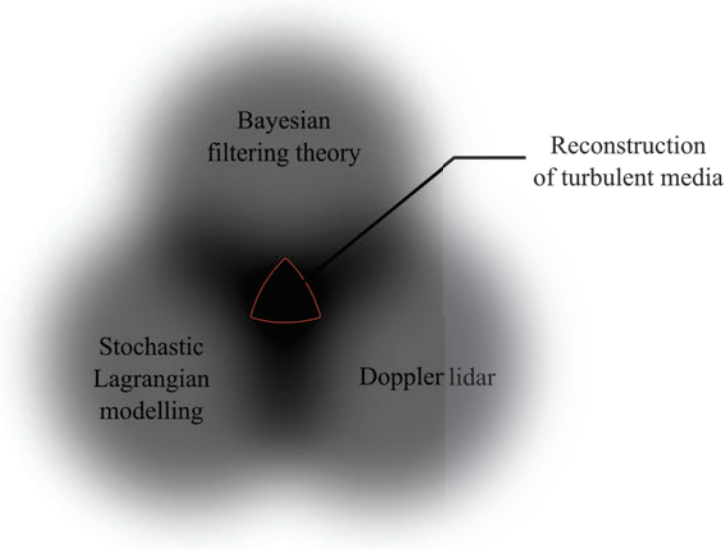


Figure 5.1 – Situation of the reconstruction regarding to previous chapters.

5.1.1 Notations

For the sake of readability, the notations used along this chapter are introduced and illustrated in this section.

In figure 5.2 is shown a visual representation of the framework. A *lidar* is staring vertically. This lidar measures the wind along the laser *beam* (it measures only the vertical component of the wind). The beam scans a *probe volume*. To get a vertical profile of vertical wind, the beam is divided into segments (*vertical levels*) and pulses coming back from a single segment are averaged to provide one wind measurement per vertical level and per time step. Measurements are denoted $V^{\mathbf{o}}(z,t)$. Such a vertical profile is the input of the reconstruction. Considering the wind probability law does not change inside a vertical level, the probe volume is seeded with *particles*. These particles are fluid particles, so that a Lagrangian model of turbulence is used to guess their movement. Particles at the same vertical level are also samples of the same wind probability law, thus Bayesian filtering algorithm for Monte Carlo methods is used to perform the filtering.

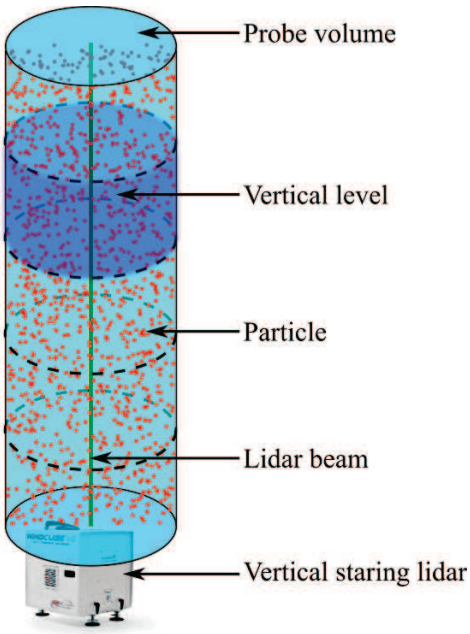


Figure 5.2 – Geometry of the 1D turbulence reconstruction problem and vocabulary.

Particles are represented numerically by position-speed (X, V) couples indexed by a time step t and a particle number i. They are initialized randomly ($\forall i,\ X_0^i \rightsquigarrow \mathcal{U}(X_{min}, X_{max}),\ V_0^i \rightsquigarrow \mathcal{N}(V^{\mathbf{o}}(z(i),0), \sigma^V)$) but after few time steps, they get some information from the following observations $V^{\mathbf{o}}(z,1),\ V^{\mathbf{o}}(z,2), ...$ Within one time step, the four steps (mutation, conditioning, selection, estimation) are done sequentially. Figure 5.3 clarifies how the state vector is denoted between these steps.

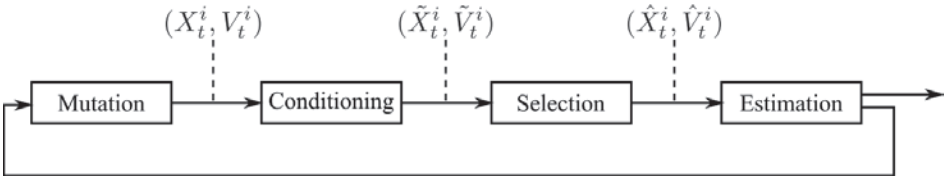


Figure 5.3 – Evolution of particle state vector within a time step.

Here is a short list of some of the notations used in the text.

- N_z : number of vertical levels ($N_z = 14$ in these experiments)
- N_t : number of time steps ($N_t = 1850$ for a dataset around 2 hours long)
- N : total number of particles ($N = 1400$ by default)
- $i \in [\![1, N]\!]$: particle number.
- $t \in [\![1, N_t]\!]$: time step number.
- $z \in [\![1, N_z]\!]$: vertical level number.

- $z(i) = \left\lfloor \frac{X^i - X_{min}}{\Delta z} \right\rfloor + 1$: vertical level number of particle i.
- Δz : width of a vertical level, portion of beam integrated to get a single measurement ($\Delta z = 50$m for this lidar).
- X_{min} : Lowest altitude measured by the lidar (limited by the time of emission: $X_{min} = 100m$).
- X_{max} : Highest altitude measured by the lidar (voluntary limited to 850m because of to many missing data above). $X_{max} = X_{min} + (N_z + 1)\Delta z$
- $\mathcal{B}(z) = [X_{min} + (z-1)\Delta z, X_{min} + z\Delta z[$: Vertical level number z.
- $B(z,t) = \{i,\, X_t^i \in \mathcal{B}(z)\}$: set of particles at level z after mutation.
- $\tilde{B}(z,t) = \{i,\, \tilde{X}_t^i \in \mathcal{B}(z)\}$: set of particles at level z after conditioning.
- $\hat{B}(z,t) = \{i,\, \hat{X}_t^i \in \mathcal{B}(z)\}$: set of particles at level z after selection.

5.1.2 Objectives

The reconstruction provides an estimation of the wind $V^{\mathbf{e}}(z,t)$ (**e** for *estimation*). This estimation involve two other signals: $V^{\mathbf{r}}(z,t)$ (**r** for *reference*) and $V^{\mathbf{o}}(z,t)$ (**o** for *observation*), which are realisations of the respective stochastic process $V^{\mathbf{r}}_{z,t}$ and $V^{\mathbf{o}}_{z,t}$. The distinctions between deterministic and stochastic signal is made in figure 5.4 in addition to their qualitative relationship.

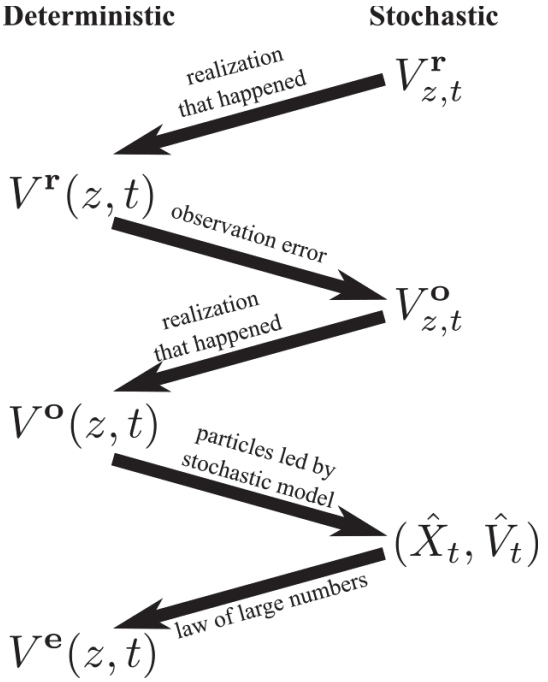


Figure 5.4 – "Who's who?" among stochastic processes.

The real wind $V^{\mathbf{r}}_{z,t}$ is a stochastic process solution of the Navier-Stokes equation in their stochastic form. Such stochastic process is unreachable since there is no analytical solution of Navier-Stokes equations. In the atmosphere, a single realisation of it is going on. This realisation is denoted $V^{\mathbf{r}}(z,t)$ and it will the signal measured by the instrument.

But the instrument is not perfect and doing a measure introduces an error. Hence, the measurement (or the observation) is described with a stochastic process $V^{\mathbf{o}}_{z,t}$

$$V^{\mathbf{o}}_{z,t} = V^{\mathbf{r}}(z,t) + \epsilon^{\mathbf{o}}_{z,t} \tag{5.1}$$

where $\epsilon^{\mathbf{o}}_{z,t}$ is the measurement error. To perform the filtering, it is assumed stationary centred and Gaussian. From the stochastic process $V^{\mathbf{o}}_{z,t}$, the instrument records only one realization: $V^{\mathbf{o}}(z,t)$. This is where starts the filtering algorithm. $V^{\mathbf{o}}(z,t)$ is the input of the reconstruction method.

In output of the reconstruction, we have the signal $V^{\mathbf{e}}(z,t)$ which is expected to be a better approximation of the wind than $V^{\mathbf{o}}(z,t)$. The computation of $V^{\mathbf{e}}(z,t)$ from $V^{\mathbf{o}}(z,t)$ involve the Lagrangian particles pictured in figure 5.2. Such particles require a Lagrangian approach and are led by a stochastoc Lagrangian model. Thus, the last layer of stochasticity in figure 5.4 is due to the particles.

What exactly is $V^{\mathbf{e}}(z,t)$? In the Eulerian approach, the wind field is the stochastic process $V^{\mathbf{r}}_{z,t}$. In the Lagrangian approach, the wind field is the couple of stochastic processes (X_t, V_t). Because Eulerian and Lagrangian approaches are consistent we have:

$$V_t = V^{\mathbf{r}}_{X_t,t} \tag{5.2}$$

Raw, the particles can get away from the lidar, and they are not linked to the observation. The different steps in figure 5.3 solve each of these issues.

- Mutation: make the particles behave according to fluid mechanics.
  $\mathcal{L}(V_t, X_t)$ is such that $V_t = V^{\mathbf{r}}_{X_t,t}$.
- Conditioning: keep the particles equally spread in the probe volume.
  $\mathcal{L}(\tilde{V}_t, \tilde{X}_t) = \mathcal{L}(V_t | X_t \in \mathcal{B}(z))$.
- Selection: use the observation to discard non-representative particles.
  $\mathcal{L}(\hat{V}_t, \hat{X}_t) = \mathcal{L}(V_t | X_t \in \mathcal{B}(z),\, V^{\mathbf{o}}_{z,t} = V^{\mathbf{o}}(z,t))$.

The estimation step, at last, computes the outputs and updates the quantities needed at the next time step. The wind estimation is performed on the particles out of the selection step:

$$V^{\mathbf{e}}(z,t) = \mathbb{E}\left[V_t | X_t \in \mathcal{B}(z),\, V^{\mathbf{o}}_{z,t} = V^{\mathbf{o}}(z,t)\right] \simeq \mathbb{E}\left[V^{\mathbf{r}}_{z,t} | V^{\mathbf{o}}_{z,t} = V^{\mathbf{o}}(z,t)\right] \tag{5.3}$$

The estimated wind $V^{\mathbf{e}}(z,t)$ is thus the average of all possible realizations of $V^{\mathbf{r}}_{z,t}$ which can

lead to the the observation $V^{\mathbf{o}}(z,t)$. In that sense, $V^{\mathbf{e}}(z,t)$ is a better estimation of wind than the single realization $V^{\mathbf{o}}(z,t)$.

In figure 5.5, the stochastic processes $V_t^{\mathbf{r}}$ and $V_t^{\mathbf{o}}$ (the z is omitted for simplicity) are represented by their average (thick line in the middle of the blur area) and their variance (span of the blur area). On the top panel, the variance of the wind increases during the day because of turbulence and weakens during the night. This change is also visible on realizations (black lines). One of these realization is the one measured. On the bottom panel, the true realization is displayed and the blur area now represents the measurement error.

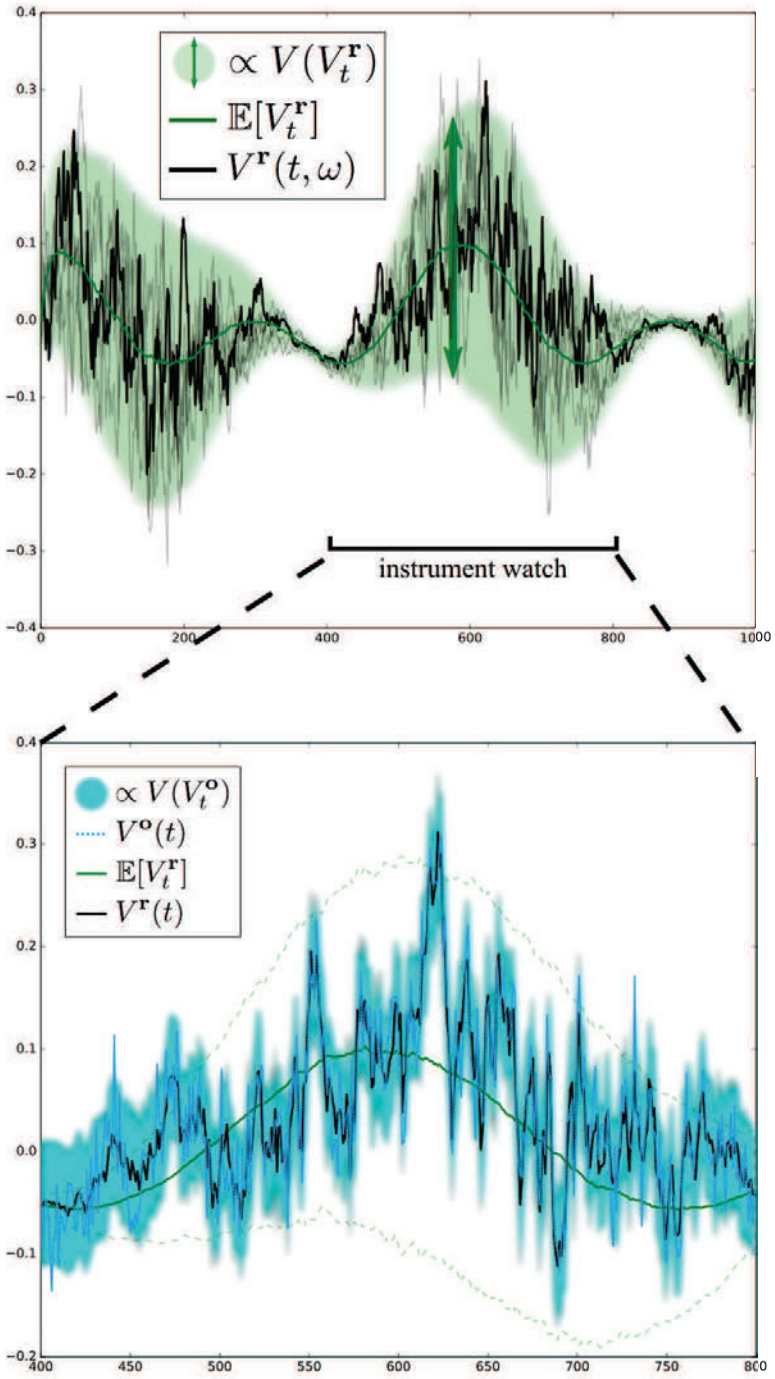


Figure 5.5 – Illustration of stochastic processes $V_t^{\mathbf{r}}$ (**r** for "reference" or "real") and $V_t^{\mathbf{o}}$ (**o** for "observation").
In the top panel, the real wind stochastic process is shown. Its average $\mathbb{E}[V_t^{\mathbf{r}}]$ is in green. The realisation that happened is in thick black. Centred around $\mathbb{E}[V_t^{\mathbf{r}}]$, the blur area depicts the most likely values for $V_t^{\mathbf{r}}$. The TKE is half the variance of that process.
In the bottom panel, the observation stochastic process is displayed. Its average is in black (equal to the realisation of $V_t^{\mathbf{r}}$ that happened). Centred around $\mathbb{E}[V_t^{\mathbf{r}}]$, the blur area depicts the most likely values for $V_t^{\mathbf{o}}$.

5.2 From theory to algorithm

5.2.1 Mutation

The mutation step is the one pushing the system from time t to time $t+dt$. Fluid is discretized with particles, and time evolution is hashed into time steps. The illustration of this step on the particle system is shown figure 5.6.

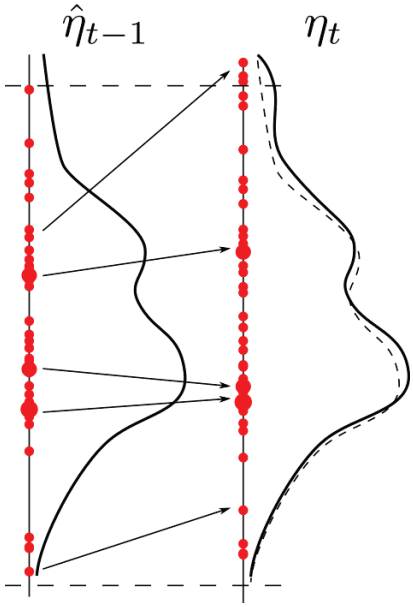


Figure 5.6 – Illustration of the mutation step on a 1D population of particles. Each particle is moved with DSLM, the initial distribution $\hat{\eta}_{t-1}$ is modified into η_t.

Initially, the stochastic Lagrangian model describes the evolution of the coupled stochastic process (X_t, V_t) (equation 1.26), recalled here:

$$\left\{ \begin{array}{rcl} dX_t &=& V_t dt \\ dV_t &=& -A_t dt - C_1 \frac{\varepsilon_t}{k_t} \left(V_t - \langle V_t \rangle\right) dt + \sqrt{C_0 \varepsilon_t} dB_t \end{array} \right. \tag{5.4}$$

Then, the stochastic process (X_t, V_t) is approached by a Monte-Carlo method.

$$X_t \overset{MonteCarlo}{\longrightarrow} (X_t^1, X_t^2, ..., X_t^N)$$

$$V_t \overset{MonteCarlo}{\longrightarrow} (V_t^1, V_t^2, ..., V_t^N)$$

The continuous time stochastic Lagragian model is given by equation (1.27). The continuous time evolution is approached by a discrete time evolution with an explicit Euler scheme. The so-called discrete stochastic Lagrangian model (DSLM) is given by equation (1.28) and

recalled here.

$$\left\{ \begin{array}{rcl} X_{t+1}^i & = & X_t^i + V_t^i \Delta t + \sigma^X \sqrt{\Delta t} \zeta_t^X \\ V_{t+1}^i & = & V_t^i - A_t^i \Delta t - C_1 \frac{\varepsilon_t^{z(i)}}{k_t^i} \left(V_t^i - \langle V_t^i \rangle^\ell \right) \Delta t + \sqrt{C_0 \varepsilon_t^{z(i)}} \zeta^V t \end{array} \right. \tag{5.5}$$

This explicit Euler scheme introduces error that Baehr chose to take into account by introducing a new term σ^X (see in Baehr (2010), system 4.4, denoted σ_n^X). It is known that stochastic differential equations as 5.4 are condensed notations hiding the Ito integral. For the position equation,

$$dX_t = V_t dt \iff X_t - X_0 = \int_0^t V_s ds$$

The Euler scheme is written as follows.

$$X_{t+\Delta t} - X_t = \int_t^{t+\Delta t} V_s ds \implies X_{t+\Delta t} = X_t + \underbrace{\int_t^{t+\Delta t} V_s ds - V_t \Delta t}_{\sigma^X \sqrt{\Delta t} \zeta^X} + V_t \Delta t$$

The error made through this discretisation is assumed to be Gaussian, centred, of standard deviation σ^X:

$$\int_t^{t+\Delta t} V_s ds - V_t \Delta t = \sigma^X \sqrt{\Delta t} \zeta_t^X \tag{5.6}$$

with ζ_t^X a centred and reduced normal random variable. The additional parameter σ^X introduced here is usually set to 0 (Bernardin et al., 2010; Rousseau et al., 2007; Suzat et al., 2011). By default, it is set to 10 meters. But the influence of this value will be assessed in the sensitivity analysis.

Terms are briefly described in table 5.1. The following list gives more details:

- C_0 : Kolmogorov's constant.
  Appear in the structure function of Monin and Yaglom (1963) used to identify the Lagrangian time in the Langevin equation (see chapter 1)

- C_1 : Fluctuation constant.
  Appear in the stochastic Lagrangian model. After the identification of the term in the Langevin equation, it is given as $C_1 = \frac{3}{4} C_0$. Pope (1994) modifies it into $C_1 = \frac{1}{2} + \frac{3}{4} C_0$. For this work, we choose to consider it as an independent constant.

- σ^X : Error of discretisation (standard deviation).
  Term usually set to 0, but taken into account by Baehr (2010) in the form of equation 5.6.

- Δt : Width of the time step ($\simeq$ 4s).
  Time between two successive measurements of lidar for all vertical levels.

- ζ_t^X, ζ_t^V : independent identically distributed as $\mathcal{N}(0,1)$ random variables.

Discrete stochastic Lagrangian model (DSLM)		
$\begin{cases} X^i_{t+1} &= X^i_t + V^i_t \Delta t + \sigma^X \sqrt{\Delta t} \zeta^X \\ V^i_{t+1} &= V^i_t - A^{z(i)}_t \Delta t - C_1 \frac{\varepsilon^{z(i)}_t}{k^i_t} \left(V^i_t - \langle V^i_t \rangle^\ell \right) \Delta t + \sqrt{C_0 \varepsilon^{z(i)}_t} \zeta^V \end{cases}$		
Coord.	i	Particle number
	t	Time step number
	$z(i)$	Vertical level at which is particle i
Constants	C_0	Kolmogorov's constant
	C_1	Fluctuation constant
	σ^X	Error of discretisation (standard deviation)
	Δt	Width of the time step ($\simeq$ 4s)
	ζ^X, ζ^V	Random variable iid as $\mathcal{N}(0,1)$
Fields	X^i_t	Position of the particle i at time step t
	V^i_t	Speed of the particle i at time step t
	$\langle V^i_t \rangle^\ell$	Local average of wind around particle i at time step t.
	k^i_t	LSTKE at time step t and particle i
	$A^{z(i)}_t$	Large scale variation at vertical level $z(i)$ and time step t
	$\varepsilon^{z(i)}_t$	Eddy-dissipation rate at vertical level $z(i)$ and time step t

Table 5.1 – Equation and nomenclature of discrete stochastic Lagrangian model used in the mutation step.

- X^i_t : position of the particle i at time step t.
  As we are in 1D, the position is only the altitude.

- V^i_t : speed of the particle i at time step t.
  As we are in 1D, the speed is only the vertical component of the wind.

- $\langle V^i_t \rangle^\ell$: local average of wind around particle i at time step t.
  The local average is a Gaussian-weighted average with standard deviation ℓ (see figure 1.5 and the corresponding section). The parameter ℓ is thus the *locality parameter* through which particles are interacting.

- k^i_t : LSTKE at time step t and particle i.
  Introduced in Pope's model as Eulerian variance (see section 1.4). The Eulerian variance is computed with Lagrangian particles thanks to a regularisation kernel (Gaussian local average) seen at the section 1.4.

- A^i_t : large scale effect on the wind at vertical level $z(i)$ and time step t.
  Account for the large scale forces acting on the particles. Set to a consistent value described at section 1.3.3.2.

- ε^i_t : Eddy-dissipation rate at vertical level $z(i)$ and time step t.
  Rate of energy transfer from larger scales to smaller scales. Set to a consistent value described at section 1.3.3.3.

As one can see, the DSLM needs several input parameters: C_0, C_1, σ^X. The main work done by the mutation step is to push forward all the prognostic fields. Thus it can be seen as the diagram 5.7.

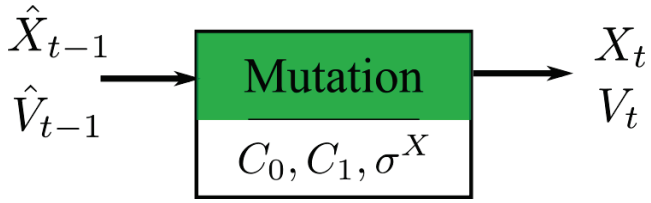


Figure 5.7 – Diagram of input/outputs and parameters for the mutation step.

5.2.2 Particle management (conditioning)

In the filtering theory (chapter 2), the filtering method has two steps: mutation and selection. When it comes to the application to moving systems (such as Lagrangian fluids), an additional step is required. Indeed, particles fluids are moving with the wind while the observing sensor is fixed. A fluid particle right in the beam at 9am can be really far away at 10am. The observation is representative of a fixed sample volume and the fluid particles do not stay into it. To handle this problem, only particles inside the probe volume are considered. It implies to give up particles which went out and to pick new particles inside (to keep a constant amount of particles). This step is called *particle management* or *conditioning*. It is illustrated by the figure 5.8.

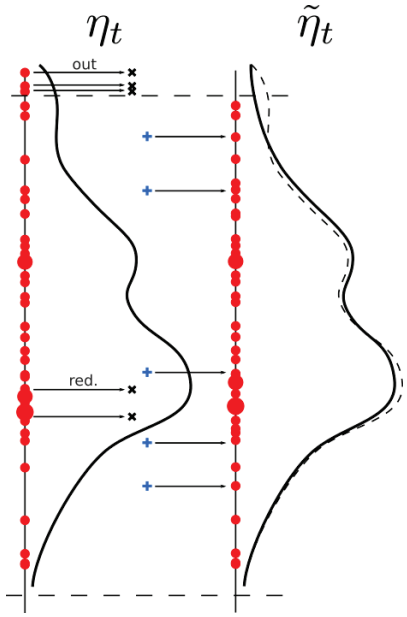


Figure 5.8 – Illustration of the conditioning step on 1D examples.

Acquisition processes, as defined in Baehr (2010) (definitions 1.1 and 1.2) are the good framework to handle this. The fluid particles of the reconstruction system are acquisition processes. With the vocabulary of Baehr, the acquisition path X_t is the trajectory of a particle X_t^i, the acquisition field $X'_{t,x}$ is the wind and the acquisition process X'_{t,X_t} is the particle speed V_t^i. The observations of lidar are also an acquisition process with a stationary acquisition path.

There are two consequences of using acquisition processes:

- Eulerian wind is estimated from Lagrangian particles with the use a local average of parameter ℓ (Gaussian regularisation kernel in Baehr (2010), equation 1.3).
- The domain of evolution is restricted to ensure compatibility among the acquisition process *particles* and the acquisition process *instrument* ("modified Markov kernel" $\tilde{M}_{n+1,\pi_n}$ in Baehr (2010), p.928).

The first consequence has been mentioned already in the section 1.3.3. The second consequence is this particle management step.

The problem is to ensure the particles correctly represent the probe volume. As particles are fluid particles, we expect the repartition of particles to be in agreement with physical insights (e.g. $\rho(z) = \rho_0 e^{-zg/RT_0}$ in an isotherm atmosphere). Hence, the new particles will be distributed according to a target Eulerian density. Our first approach is to consider $\rho(z) = \rho_0$ as target Eulerian density (one wants particles equally spread on all the probe volume).

The free repartition of particles resulting from mutation has two cases that are pathological:

- particles out of the probe volume
- part of the probe volume rid of particles

Many answers are possible. The first answer was provided by Baehr for the case of a single anemometer Baehr (2008). But when the probe volume is measured in different points (as it is the case with lidar), things get more complicated because one must allow particles to move among the measurement points. Hence, the procedure proposed by Rottner (2015) is much more complex than for single anemometer, but gives a first solution for lidar. We will focus only on the solution actually implemented here, slightly different from Rottner's one.

The algorithm A.1 will be used intensively. As reminder, the algorithm A.1 provides realizations of any random variable described by a potential G (G can be any real positive integrable function, stored numerically as an array of positive number). The notation

$$X \rightsquigarrow G$$

means "X is a G-distributed sample generated with algorithm A.1". All the details of the algorithm are given page 267. We also recall that the notation $X \rightsquigarrow \mathcal{U}(E)$ means "X is a

uniform-distributed sample of elements of E".

The *particle management* step is divided in two distinct algorithms: one to replace particles out the probe volume 5.1, and one to redistribute the particles in the probe volume 5.2.

5.2.2.1 Particles out of the probe volume

The algorithm 5.1 describe how particles which are out of the probe volume after the mutation step are forgotten in favour of new ones. First, particles out are detected (I_{out} denotes the set of particles out of the domain). Particles numbers $i \in I_{out}$ of these particles will be reallocated to new particles in the probe volume. Particles out are reallocated in order to equilibrate particle population among vertical levels. For each particle out, we first choose a vertical level of arrival z_{arr} randomly, according to a potential. The potential used (denoted G_{out}) is chosen in a way which brings closer to the target density.

$$G_{out} : \begin{array}{ccc} [\![1, N_z]\!] & \to & [0, +\infty[\\ z & \mapsto & G_{out}(z) = N - |B(z,t)| \end{array} \tag{5.7}$$

with $B(z,t) = \{i,\, X_t^i \in \mathcal{B}(z)\}$ ($B_z(t)$ is the set of particles at level z at time t) and $|B(z,t)|$ its cardinal.

This potential is high for vertical levels with few particles. The higher is its potential, the more likely this vertical level will be picked up. Hence, the vertical level z_{arr} chosen according to G_{out} is probably poor in particles. Once the arrival vertical level z_{arr} is found, the particle out is assigned to this level: its position is updated. The speed of the reallocated particle is updated differently whether the arrival level is empty or not. If there are particles at the arrival level after the mutation step, one of them i_{arr} is chosen randomly (uniformly) and the reallocated particle takes its speed with a slight modification ($\tilde{V}_t^i = V_t^{i_{arr}} + \sigma^V \zeta_t$). If the arrival level is empty ($B(z_{arr}, t) = \emptyset$), there is no particle to copy, then the speed of reallocated particle is set to the observation with a slight modification ($\tilde{V}_t^i = V^{\mathbf{o}}(z_{arr}, t) + \sigma^V \zeta_t$). The slight modification $\sigma^V \zeta_t$ (with $\zeta_t \rightsquigarrow \mathcal{N}(0,1)$) is added to avoid wads of particles with the same speed. Wads of particles with the exact same speed are a problem because they increase the chance of null potential in the selection step (see following section).

5.2.2.2 Redistribution of particles

Once all the particles are inside the domain, the redistribution of the particles is done. This step ensures that there are enough particles at each level to perform relevant statistics. Baehr sees it has a redistribution to reach a target density profile $\rho(z) \propto |\tilde{B}(z,t)|/N$. In this framework, our target would be a constant density : $\rho(z) = \rho_0$. But the presented algorithm (5.2) also keeps place for the density profile ($z \mapsto |\tilde{B}(z,t)|$) to evolve freely. It only ensures

Algorithm 5.1 Replace particles out the probe volume

Input: $(X_t^i, V_t^i)_{i\in[\![1,N]\!]}$, $V^{\mathbf{o}}(z,t)_{z\in[\![1,N_z]\!]}$
Output: $(\tilde{X}_t^i, \tilde{V}_t^i)_{i\in[\![1,N]\!]}$

```
I_out = {i, X_t^i ∉ [X_min, X_max[}
for i ∈ I_out do
    G_out(z) = N − |B(z,t)|
    z_arr ⇝ G_out
    X̃_t^i ⇝ U(B(z_arr))
    if B(z_arr, t) = ∅ then
        ζ ⇝ N(0,1)
        Ṽ_t^i = V^o(z_arr, t) + σ^V ζ
    else
        i_arr ⇝ U(B(z_arr, t))
        ζ ⇝ N(0,1)
        Ṽ_t^i = V_t^{i_arr} + σ^V ζ
    end if
end for
return (X̃_t^i, Ṽ_t^i)_{i∈[[1,N]]}
```

that a minimum amount of N_{min} particles is present at each level. Inequalities are not treated as long as $|\tilde{B}(z,t)| \geqslant N_{min}$.

The algorithm 5.2 comes right after the algorithm 5.1. It is the second task of the conditioning step on figure 5.3. For each vertical level z_{arr}, its population $|\tilde{B}(z_{arr},t)|$ is checked and increase if necessary. If $|\tilde{B}(z_{arr},t)| \geqslant N_{min}$, the computer does not enter the `while` loop: nothing is done for this level, the next level is tested. If $|\tilde{B}(z_{arr},t)| < N_{min}$, then particles will be added to this level until there are at least N_{min} particles. The threshold N_{min} is arbitrarily set to 80% of the average population ($N_{min} = 0.8N/N_z$), which leaves a 20% span of freedom for the density profile to evolve with the flow. Particles to refill the vertical level z_{arr} in need are taken from another vertical level z_{dep}. The departure vertical level z_{dep} is chosen randomly, according to the potential G_{red}.

$$G_{red} : \begin{array}{ccc} [\![1,N_z]\!] & \to & [0,+\infty[\\ z & \mapsto & G_{dep}(z) = max(|\tilde{B}(z,t)| - N_{min}, 0) \end{array} \tag{5.8}$$

The potential G_{red} has the inverse trend than G_{out} (equation 5.7): level loaded with particles have high G_{red} (and low G_{out}), as it can be visualized on figure 5.9. Hence, the chosen vertical level z_{dep} has a population probably high. In addition, G_{red} has been constructed to ensure that underloaded levels (with $|\tilde{B}(z,t)| < N_{min}$) cannot be chosen. Indeed, such levels have $G_{red}(z) = 0$ and levels with potential 0 cannot be chosen (as explained in section A.2.3). In the end, only overloaded levels can be chosen, with a probability proportional to their population. Once the departure level z_{dep} is chosen, a particle i_{dep} is chosen uniformly inside. The particle i_{dep} is assigned to the underloaded level: its position is updated. The speed of the reallocated particle is updated differently whether the arrival level is empty or not. If there are particles at the arrival level after the mutation step, one of them i_{arr} is chosen

randomly (uniformly) and the reallocated particle takes its speed with a slight modification ($\tilde{V}_t^{i_{dep}} = \tilde{V}_t^{i_{arr}} + \sigma^V \zeta_t$). If the arrival level is empty ($\tilde{B}(z_{arr}, t) = \emptyset$), there is no particle to copy, then the speed of reallocated particle is set to the observation with a slight modification ($\tilde{V}_t^{i_{dep}} = V^{\mathbf{o}}(z_{arr}, t) + \sigma^V \zeta_t$). As for reallocation of particles out, the slight modification $\sigma^V \zeta_t$ (with $\zeta_t \rightsquigarrow \mathcal{N}(0,1)$) is added to avoid problematic wads of particles with the same speed.

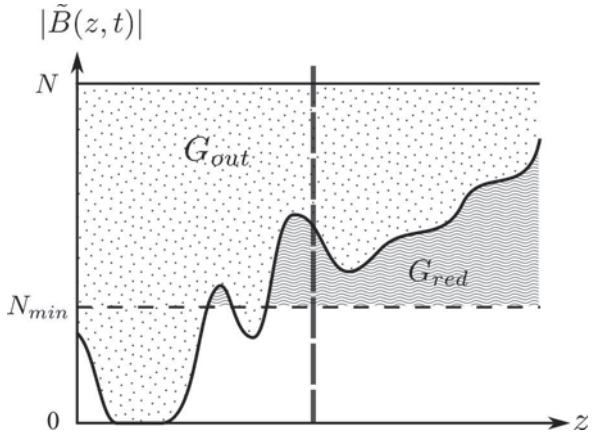


Figure 5.9 – For each vertical level (vertical dashed line), the potential for particles out G_{out} (resp. redistribution G_{red}) are proportional to the segment in the dotted (resp. wavy) area.

In the end, the *conditioning* step is a two tasks step to ensure particles are correctly spread in the probe volume. The first task is to reallocate particles out of the probe volume. The second task is to ensure a minimum amount of particles is present at each level. These two tasks are respectively detailed in algorithms 5.1 and 5.2. In these algorithms, it can be found several parameters: N, σ^V. This step push the particle from their state "after mutation" $(X_t^i, V_t^i)_{i \in [\![1,N]\!]}$ to their state *after conditioning* $(\tilde{X}_t^i, \tilde{V}_t^i)_{i \in [\![1,N]\!]}$. Hence, the resulting diagram of this step would the figure 5.10.

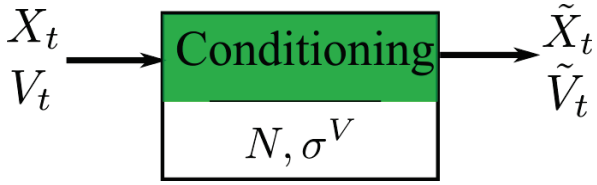


Figure 5.10 – Diagram of input/outputs and parameters for the conditioning step.

Algorithm 5.2 Redistribute particles inside the probe volume

Input: $(\tilde{X}_t^i, \tilde{V}_t^i)_{i\in[\![1,N]\!]}$, $V^{\mathbf{o}}(z,t)_{z\in[\![1,N_z]\!]}$
Output: $(\tilde{X}_t^i, \tilde{V}_t^i)_{i\in[\![1,N]\!]}$

$N_{min} = 0.8\frac{N}{N_z}$
Refresh $\tilde{B}(z,t)$ for all z.
for $z_{arr} = 1 : N_z$ **do**
  Refresh $\tilde{B}(z_{arr},t)$.
  while $|\tilde{B}(z_{arr},t)| < N_{min}$ **do**
    $G_{red} = \max(|\tilde{B}(z,t)| - N_{min}, 0)$
    $z_{dep} \rightsquigarrow G_{red}$
    $i_{dep} \rightsquigarrow \mathcal{U}(\tilde{B}(z_{dep},t))$
    $\tilde{X}_t^{i_{dep}} \rightsquigarrow \mathcal{U}(\mathcal{B}(z_{arr}))$
    if $B(z_{arr},t) = \emptyset$ **then**
      $\zeta \rightsquigarrow \mathcal{N}(0,1)$
      $\tilde{V}_t^{i_{dep}} = V^{\mathbf{o}}(z_{arr},t) + \sigma^V\zeta$
    else
      $i_{arr} \rightsquigarrow \mathcal{U}(B(z_{arr},t))$
      $\zeta \rightsquigarrow \mathcal{N}(0,1)$
      $\tilde{V}_t^{i_{dep}} = \tilde{V}_t^{i_{arr}} + \sigma^V\zeta$
    end if
  end while
end for
return $(\tilde{X}_t^i, \tilde{V}_t^i)_{i\in[\![1,N]\!]}$

5.2.3 Selection

The selection step is the core of the filtering method. As seen in the filtering theory (chapter 2), the prior information is brought by the Lagrangian model (mutation step) but this information is incorporated into the cloud of particles through the use of Bayes' theorem: this is the selection step. The swarm of particles coming from mutation and conditioning brings one information: what would be the wind distribution Δt seconds after the last observation? Δt seconds later, a new (noisy) observation is available: this is another source of information. The selection step modifies the cloud of particles to fit the best with both sources of information. It is called *selection* because the modification is done by selecting *good* particles (that will stay the same) and getting rid of *bad* particles (that will be reallocated). This is illustrated on figure 5.11.

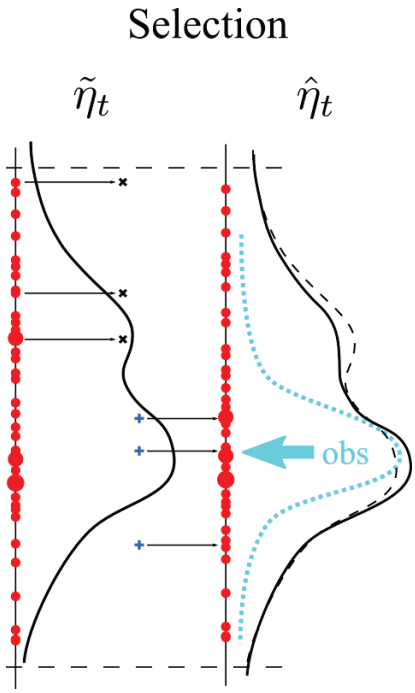


Figure 5.11 – Illustration of the selection step on a 1D population of particles. Particles are rejected and resampled according to their likelihood, the initial distribution $\tilde{\eta}_t$ is modified into $\hat{\eta}_t$. One can see that the histogram have grown where the observation is and is flatter elsewhere.

The distinction between *good* and *bad* is made by using the likelihood of particles with respect to the new observation. As in conditioning, the reallocation of particles is made using the algorithm A.1 to get G-distributed sample. For the selection, the potential G is the likelihood of particles with respect to the new observation (denoted G_{obs}). To compute this likelihood, the probability law of the measurement error must be known. It is common to assume Doppler lidar have Gaussian error (Frehlich and Yadlowsky, 1994). Hence, we assume the error of measurement is Gaussian with a standard deviation σ^{obs} (input parameter set to "reasonable" value). It yields to the potential G_{obs}, defined for each vertical level by equation

5.9.

$$G_{obs}: \begin{array}{ccc} \tilde{B}(z,t) & \rightarrow & [0,+\infty[\\ i & \mapsto & G_{obs}(i) = \exp\left(-\frac{\left(\tilde{V}_t^i - V^{\mathbf{o}}(z,t)\right)^2}{2(\sigma^{obs})^2}\right) \end{array} \tag{5.9}$$

Three comments are to be made on the expression of G_{obs} (5.9). First, even if the observation is known to be very noisy, particles with the exact same speed as measured will have a potential 1. They are sure to be conserved. It is the consequence of the assumption the observation is unbiased. Otherwise, it would be required to have a guess of the bias and remove it from the measured value. Second, the decrease of the potential as $\tilde{V}_t^i$ is getting away from $V^{\mathbf{o}}(z,t)$ is scaled by σ^{obs}. If σ^{obs} is large (which means the observation is very noisy), the potential will decrease slowly and even particles with a very different speed than measured can be conserved. If σ^{obs} is small (which means the observation is very trustworthy), the potential will decrease sharply and only particles with a speed very close to the measurement are conserved. The parameter σ^{obs} is thus of prime importance. Its influence will be studied in the sensitivity analysis (see chapter 6). Third, the value of potential G_{obs} is only relative: the potential of a single particle makes no sense. It makes sense only in comparison with other particles.

We have seen in the section 2.3.3 that different ways exist to perform the selection step. The chosen selection kernel implies to make a acceptance-rejection step before resampling according to the likelihood. The acceptance-rejection is also made with the likelihood. Hence, the potential G_{obs} is used for two purposes during the selection step: first for acceptance-rejection of particles, second for the resampling of the rejected particles I_{rej}. Accepted particles are kept unchanged. Rejected particles are resampled with an SIR algorithm. Only speed is used to select the particles (through the definition of G_{obs}) but speed and position are modified in the resampling. The complete algorithm of the selection step is given next (algorithm 5.3). Diagram with inputs, outputs and parameters is given figure 5.12.

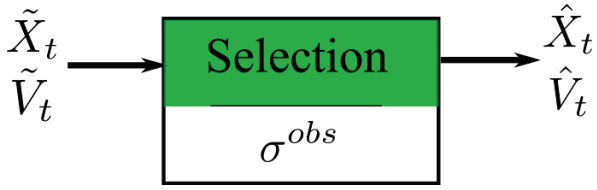


Figure 5.12 – Diagram of input/outputs and parameters for the selection step.

Algorithm 5.3 Selection of particles according to their likelihood.

Input: $(\tilde{X}_t^i, \tilde{V}_t^i)_{i\in[\![1,N]\!]}$, $V^{\mathbf{o}}(z,t)_{z\in[\![1,N_z]\!]}$
Output: $(\hat{X}_t^i, \hat{V}_t^i)_{i\in[\![1,N]\!]}$
  Refresh $\tilde{B}(z,t)$ for all z.
  $(\hat{X}_t^i, \hat{V}_t^i)_{i\in[\![1,N]\!]} = (\tilde{X}_t^i, \tilde{V}_t^i)_{i\in[\![1,N]\!]}$
  for $z = 1 : N_z$ **do**
    $G_{obs}(i) = \exp\left(-\frac{(\tilde{V}_t^i - V^{\mathbf{o}}(z,t))^2}{2(\sigma^{obs})^2}\right), \forall i \in \tilde{B}_z(t)$ # Likelihood
    $u \rightsquigarrow \mathcal{U}([0,1])$
    $I_{rej} = \{i \in \tilde{B}(z,t),\ G_{obs}(i)/\max(G_{obs}) < u\}$
    for $i_{rej} \in I_{rej}$ **do**
      $i_{new} \rightsquigarrow G_{obs}$
      $(\hat{X}_t^{i_{rej}}, \hat{V}_t^{i_{rej}}) = (\tilde{X}_t^{i_{new}}, \tilde{V}_t^{i_{new}})$
    end for
  end for
  return $(\hat{X}_t^i, \hat{V}_t^i)_{i\in[\![1,N]\!]}$

5.2.4 Estimation

The cloud of particles in output of the selection step is the best guess we have of the wind probability law. It is the population on which the estimation must be performed. Hence, output of the reconstruction are computed at this step. The main output is the estimation of wind $V^{\mathbf{e}}(z,t)$ (equation 5.10) at each vertical level. It can thus be written with the spatial average $\langle\cdot\rangle$ introduced in the definition 1.7:

$$V^{\mathbf{e}}(z,t) = \left\langle \hat{V}_t \right\rangle = \frac{1}{|\hat{B}(z,t)|} \sum_{i\in\hat{B}(z,t)} \hat{V}_t^i \tag{5.10}$$

The estimated wind $V^{\mathbf{e}}(z,t)$ differs from the observed wind $V^{\mathbf{o}}(z,t)$ because most of the noise in the $t \mapsto V^{\mathbf{o}}(z,t)$ signal has been removed. It is visible on the time series (see figure 5.15) and even clearer on the PSD (see figure 5.16). Hence $V^{\mathbf{e}}(z,t)$ is a much more accurate guess of the wind in turbulent conditions. Rottner's later works show the improvement of using $V^{\mathbf{e}}(z,t)$ instead of $V^{\mathbf{o}}(z,t)$ in turbulence monitoring for wind energy. In Rottner and Baehr (2014) and in Rottner (2015) (figure 4.21, page 90), it is shown that using the filtered wind to estimate the turbulent intensity[1] is better than using the observed wind directly (Gryning et al. (2017), figure 2.5).

Beside filtered wind, the reconstruction provides turbulent kinetic energy estimation. We have seen in the chapter 1 several TKE estimators. In order to compare them, they can be computed all together. In the previous works of Baehr (2008) and Rottner (2015), only the LSTKE was computed. In compliance with these work, the LSTKE is the estimator used in the stochastic Lagrangian model. LSTKE is defined in the chapter 1, definition 1.10, we

[1] Turbulent intensity (TI) is the quantity in use in wind power industry to assess the loss of power due to turbulence. It is the ratio between wind standard deviation and mean wind $TI = \sqrt{u'^2}/\bar{u}$

recall it here (superscript $\mathbf{e}$ is for *estimation*):

$$k^{\mathbf{e}}(z,t) = k^{LS}(z,t) = \frac{1}{2}\left\langle\left\langle\left(\hat{V}_t^i - \left\langle\hat{V}_t^i\right\rangle^\ell\right)^2\right\rangle^\ell\right\rangle \tag{5.11}$$

This step is also the opportunity to update all the quantities required in the Lagrangian model because the particles have been selected. From the chapter 1, section 1.3.3, we have the following estimates:

$$\left\langle V_t^i\right\rangle^\ell = \frac{\sum_{j=1}^N V_t^j \Phi_\ell(X_t^i, X_t^j)}{\sum_{j=1}^N \Phi_\ell(X_t^i, X_t^j)}$$

$$A_t^i = -\frac{1}{\Delta t}\sum_{j\in\hat{B}(z(i),t)} \frac{(\hat{V}^j - \hat{V}_{t-1}^j)}{|\hat{B}(z(i),t)|}$$

$$\varepsilon_t^i = -\frac{1}{C_0\Delta t}\sum_{j\in\hat{B}(z(i),t)} \frac{(\hat{V}_t^j - \hat{V}_{t-1}^j)^2}{|\hat{B}(z(i),t)|}$$

with Φ_ℓ the Gaussian regularization kernel displayed in figure 1.5 and $\langle\cdot\rangle^\ell$ the corresponding average operator. Because of the many calculations carried out at this step, it is often the heaviest step in terms of computing time, as shown in the figure 5.22. The diagram for this step would be 5.13. Note that it does not update the state vector, it just provides the outputs and updates quantities needed in the mutation step.

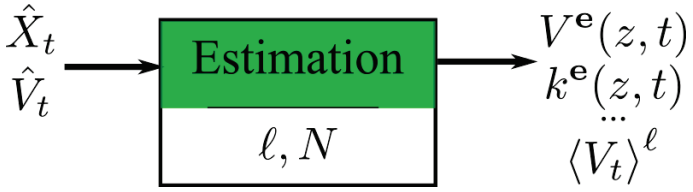


Figure 5.13 – Diagram of input/outputs and parameters for the estimation step.

5.2.5 Summary

The four main steps of the reconstruction algorithm have been described in this section. They are repeated as long as there are observation available. The resulting diagram is in figure 5.14.

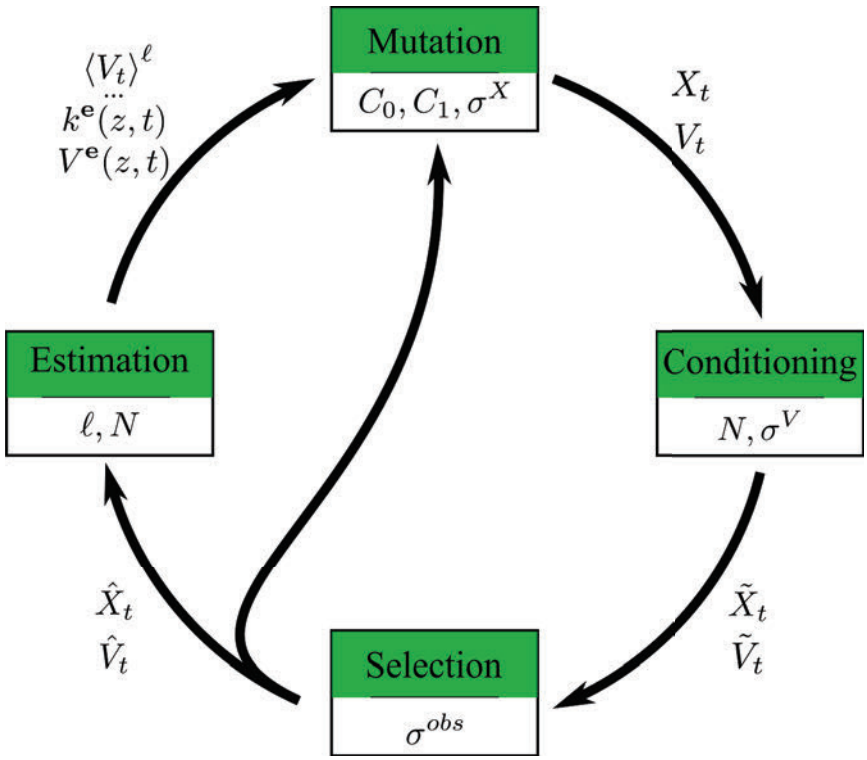


Figure 5.14 – Organisation of the four main steps of the reconstruction algorithm in cycle.

5.3 Hints for practitioners

Initially, the code of the reconstruction algorithm was in Scilab. In order to run the code many times, as required for sensitivity analysis, it has been translated to Fortran 90. At the occasion of the translation, a special care has been taken to improve the algorithm efficiency (reduction of loops, use of optimized libraries such as BLAS-LAPACK, a start of parallelisation). As a result, the computing time was improved of a factor 150: for comparable settings ($N = 1400$), the Scilab code was processing 2h of data in 3h, the Fortran code processes 2h of data in 70 seconds. The restriction of the Fortran code to the 1-dimensional case must temper this result. The software engineering work, although massive, will not be presented here. Instead of a cumbersome description of the unitary and integration tests carried out, we present a summary of the experiment ensuing from such work. First, a list of usual graphic outputs is given. Then, a list of checkpoints to ensure the code is doing what it is supposed to. Finally, few words about the known limitations of the algorithm.

5.3.1 Usual outputs

Turbulence medium reconstruction is a complex system in which many things are worth looking at. This paragraph intents to give an idea of the most used outputs. Here is the list of the presented figures:

- Times series of wind (5.15).
- PSDs of wind (5.16).
- Scatter-plot of wind (5.17).
- Times series of TKE (several estimators of TKE available) (5.18).
- Scatter-plot of TKE (5.19).
- Amount of particles out, redistributed and rejected (5.20).
- Histogram of maximum potential (5.21).
- The pie chart of time spent inside each step (5.22).

All of them are commented in their caption. To build the system used for the sensitivity analysis, some of these figures have been reduced to a score. For example, the visual comparison of the time series of wind is reduced to the estimation of the root-mean-squared error. The outputs presented here are relevant for a single run of the reconstruction, that we want to analyse carefully. The scores are presented in the next section.

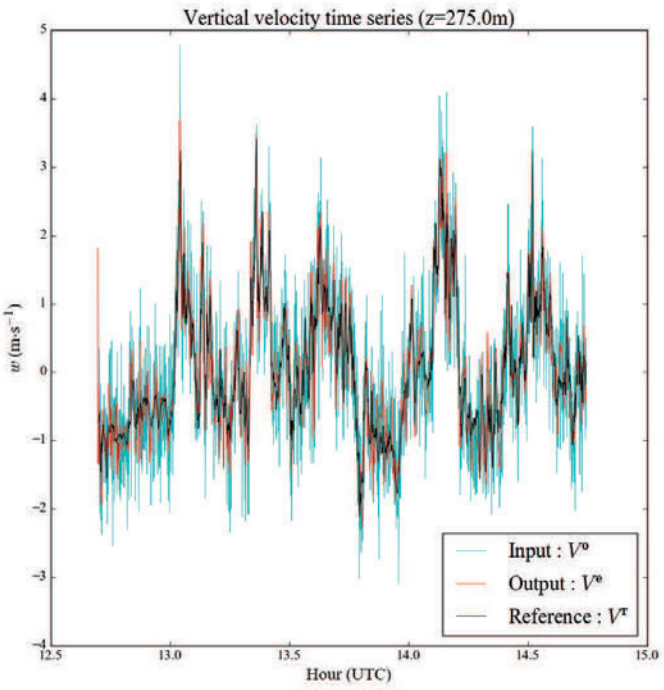


Figure 5.15 – Time series of wind. Reference (black), simulated observation (cyan) and estimated wind (red) are displayed. In this figure one can check is the estimated signal matches better the reference than the observation.

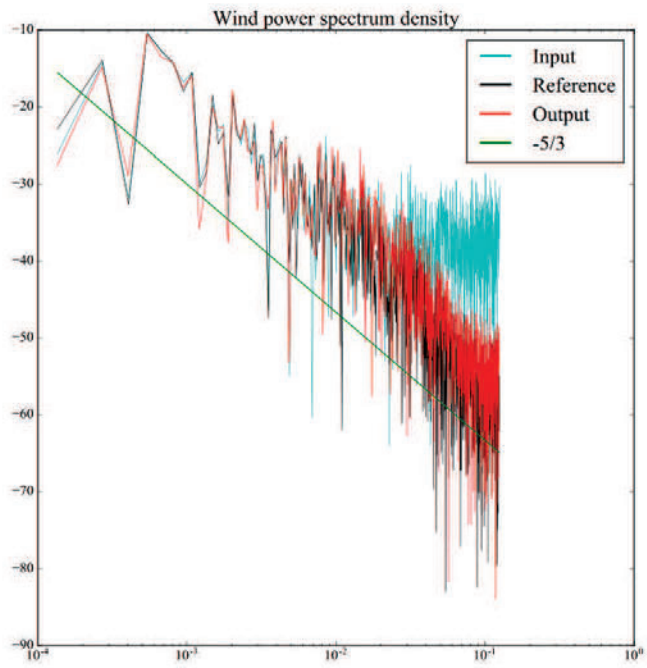


Figure 5.16 – PSDs of reference (black), simulated observation (cyan) and estimated (red) wind are diplayed. One can check the observation has a horizontal spectrum at high frequency, signature of white noise in the signal, while the reference and the estimation have almost the same spectrum, close to the -5/3 target slope.

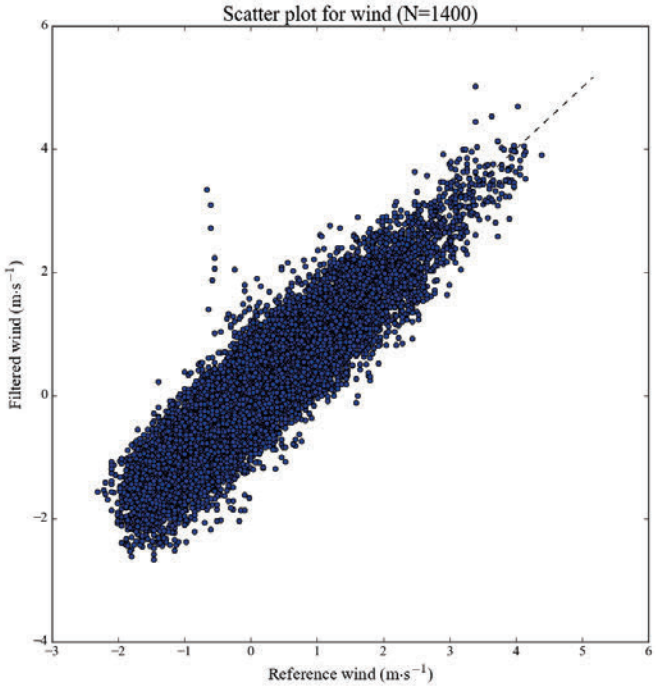


Figure 5.17 – Scatter-plot of estimated wind against reference wind. Points are all near the 1:1 line, which indicate they match pretty well.

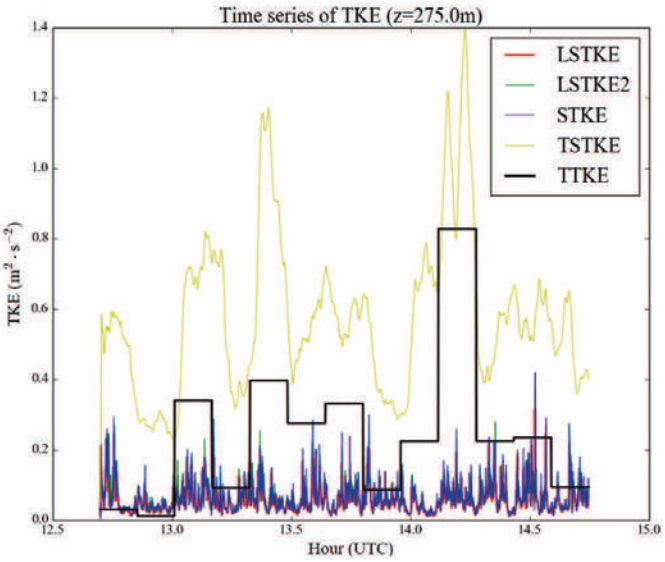


Figure 5.18 – Times series of TKE (several estimators of TKE available).

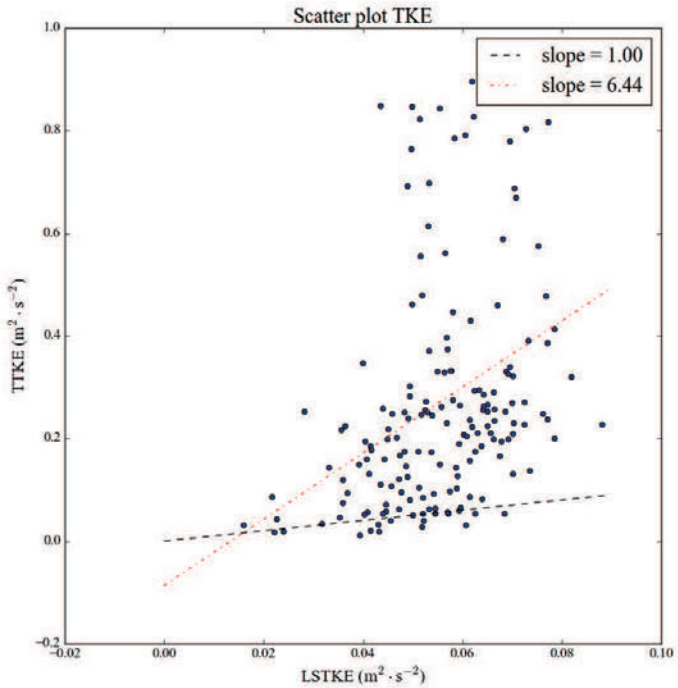


Figure 5.19 – Scatter-plot of LSTKE against TTKE. One can see the LSTKE is fairly underestimated compared to TTKE (about 6.44 times). See chapter 1, page 32, for interpretation.

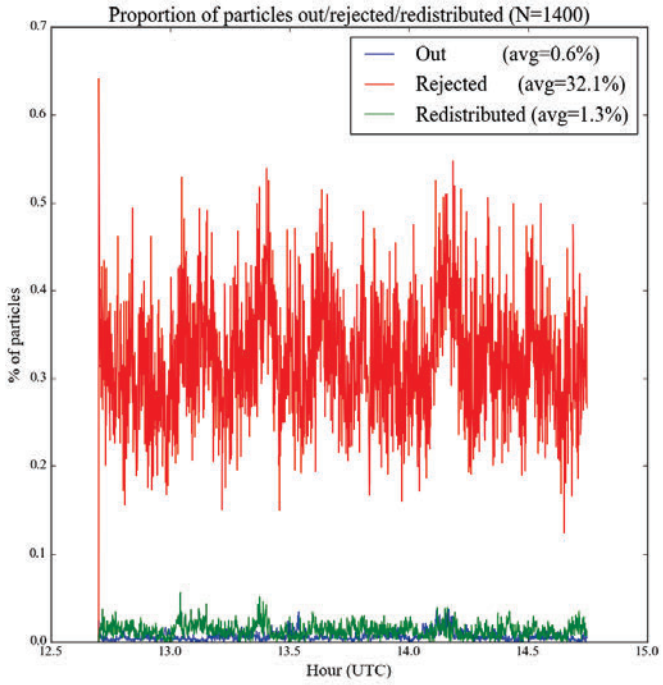


Figure 5.20 – Amount of particles out (blue), redistributed (green) and rejected (red). One can see there are very few particles affected by the conditioning (out or redistributed) while the number of rejected is significant. Too many rejected particles (>90%) might be the symptom of a degeneracy. Conversely, too few rejected (<20%) particles might the symptom of a bypass of the selection.

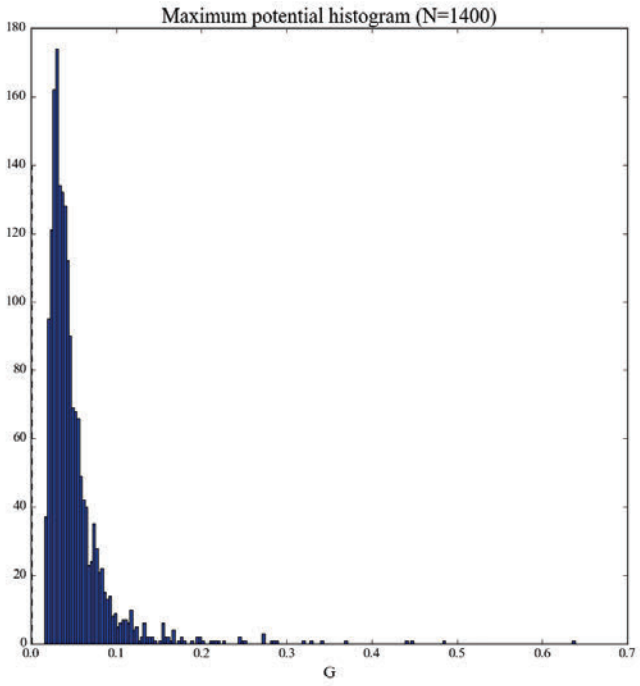


Figure 5.21 – Example of histogram of maximum potential. This histogram has been shown to have a shape which can help to diagnose degeneracy of the filter.

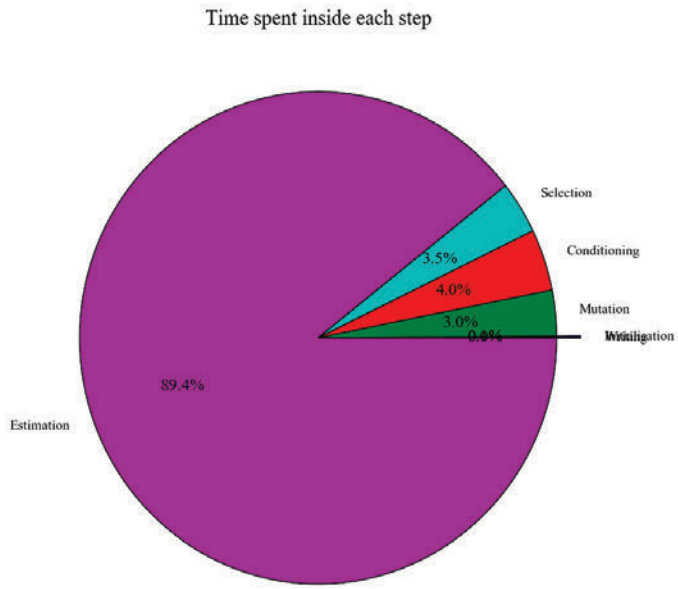


Figure 5.22 – The pie chart of time spent inside each step. The total time for one run of the 2 hours dataset is about 70 seconds. Most of this time is spent at the estimation step because of the large amount of bulk calculations. Other steps have been greatly optimized during the recoding in Fortran 90.

5.3.2 Criteria of good behaviour

A large part of time was dedicated to the writing of the code in Fortran. From this engineering work, we capitalized a good experiment of the reconstruction code, especially when it doesn't work. To sum things up, the table 5.2 gathers the main checkpoints a developer should look at in case of problem.

	Point to check	**Criterion**	**Justification**
1	Order of magnitude of wind	about ± 1 m·s^{-1}	Same as lidar measurement
2	Order of magnitude of TKE	about 0.9 m^2·s^{-2} daytime, 0.2 nighttime.	For instance Darbieu et al. (2015) (figure 9)
3	Order of magnitude of EDR	de 10^{-5} à 0.2 m^2·s^{-3}	Smalikho et al. (2005), Chan (2011) (figure 7) et Cohn (1995) (figures 3, 5 et 7)
4	Slope of sind spectrum	Close to -5/3	Kolmogorov's spectrum Kolmogorov (1941); Frisch (1995)
5	Shape of wind time series	see figure 5.23	Same as lidar measurement
6	Shape of TKE time series	see figure 5.25	From Rottner (2015) (figure 4.17c), not found elsewhere at such frequency.
7	Shape of EDR time series	see figure 5.26	From Rottner (2015) (figure 4.18c), similar to Meneveau and Sreenivasan (1987) (figure 1)
8	Shape of maximum weight histogram	see figure 5.24	Bengtsson et al. (2008)
9	RMSE value	$<$ RMSE observation alone	Filtering is supposed to reduce error
10	Amount of rejected particles	about 30%	Experience...
12	Order of magnitude of terms in mutation	± 0.5m·s^{-1}	Experience...
13	Order of magnitude of terms in fluctuation	± 1	Experience...

Table 5.2 – List of criteria to check the program is well written (sorted by importance)

5.3.3 Known limitations

The reconstruction has been designed with a stochastic Lagrangian model based on Kolmogorov's hypothesis. But Kolmogorov's hypothesis hold for high Reynolds number. At night, or in stable conditions, the wind and the turbulence weaken. The stochastic Lagrangian model is not designed for such conditions. As a consequence, the filtering is not performing really well in stable conditions. It has been pointed out by Rottner et al. (2016)

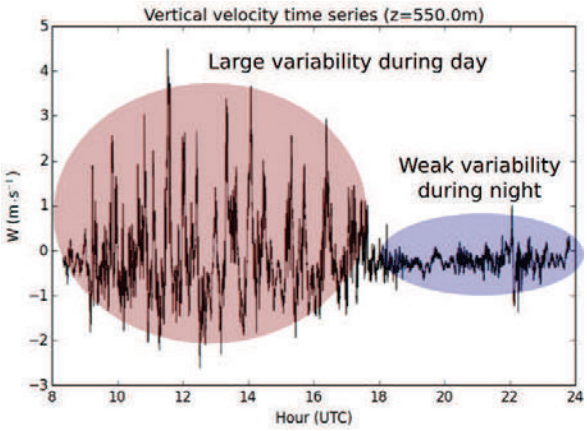


Figure 5.23 – Typical shape of a wind time series.

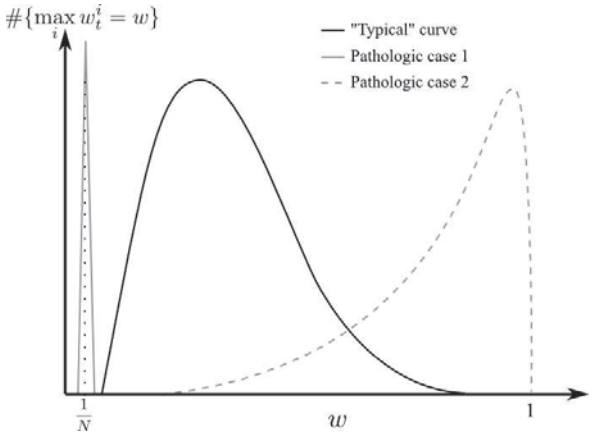


Figure 5.24 – Shape of maximum weight histogram.

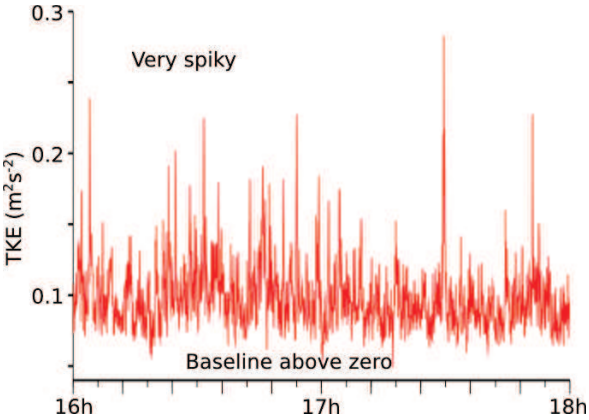


Figure 5.25 – Typical shape of a TKE time series (vertical component, from Rottner (2015)).

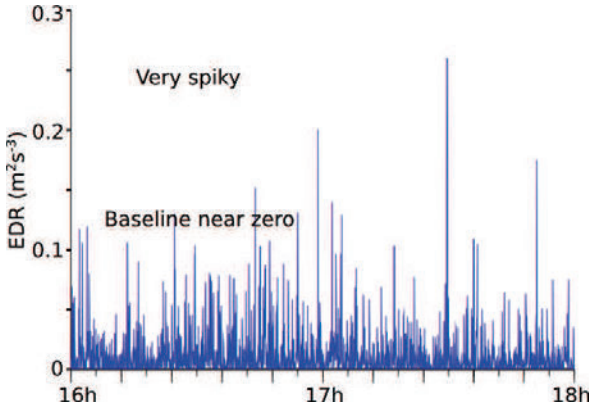


Figure 5.26 – Typical shape of a EDR time series (vertical component, from Rottner (2015)).

and in Rottner and Baehr (2017). This limitation is known but has not been a barrier for the sensitivity analysis carried out in this work.

5.4 Validation experiment and resulting scores

5.4.1 The validation framework

In order to test the filtering process, it is embraced into a larger framework. The reconstruction takes in input the measurements $V^{\mathbf{o}}(z,t)$ and gives in output a set of particles describing the PDF of the wind. Such particles are used for the estimation of the real wind (with less measurement error) $V^{\mathbf{e}}(z,t)$ and fast estimation of TKE $k^{\mathbf{e}}(z,t)$. Hence, the reconstruction itself can be seen as a function $\mathscr{R}$ (equation 5.12 and figure 5.27)

$$\begin{array}{rccc} \mathscr{R}: & L^2(\mathbb{R})\times\mathbb{R}^p & \longrightarrow & L^2(\mathbb{R})\times L^2(\mathbb{R}) \\ & V^{\mathbf{o}},\theta_R & \longmapsto & V^{\mathbf{e}},k^{\mathbf{e}} \end{array} \tag{5.12}$$

with

$$\begin{array}{rccc} V^{\mathbf{o}},V^{\mathbf{e}},k^{\mathbf{e}}: & \mathbb{R}\times\mathbb{R}^+ & \longrightarrow & \mathbb{R} \\ & z,t & \longmapsto & V^{\mathbf{o}}(z,t),V^{\mathbf{e}}(z,t),k^{\mathbf{e}}(z,t) \end{array}$$

and

$$\theta_R = \begin{pmatrix} C_0 \\ C_1 \\ \ell \\ N \\ \sigma^{obs} \\ \sigma^V \\ \sigma^X \end{pmatrix} \tag{5.13}$$

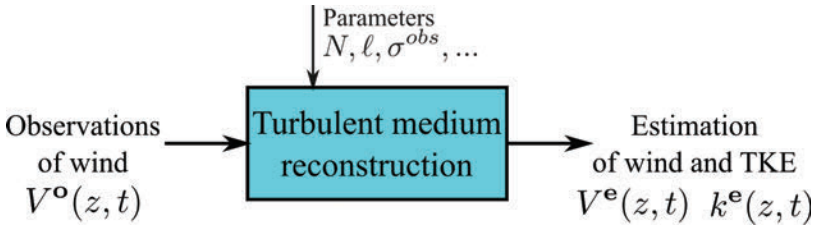


Figure 5.27 – Inputs and outputs of turbulence reconstruction.

The problem to assess the filtering process is that the estimation $V^{\mathbf{e}}(z,t)$ is supposed to approach $V^{\mathbf{r}}(z,t)$, not $V^{\mathbf{o}}(z,t)$. Hence, the embracing framework have to provide a reference $V^{\mathbf{r}}(z,t)$ to which the estimation can be compared. The reference $V^{\mathbf{r}}(z,t)$ is chosen: it is high quality lidar measurement (no noise visible on spectra). From that reference, the observation is simulated. This framework allows to control the observation noise. The filtering method (turbulent medium reconstruction) is applied to the simulated observation. Next, outputs of the reconstruction are compared to the reference. Hence, validation scores can be computed. The global framework is summarized in figure 5.28. Since new parameters appear in this construction (for example, the observation noise is now under control, its variance is σ^{add}),

the vector of input parameters is completed.

$$\theta = \begin{pmatrix} C_0 \\ C_1 \\ \ell \\ N \\ \sigma^{add} \\ \sigma^{obs} \\ \sigma^{V} \\ \sigma^{X} \\ \tau \end{pmatrix} \tag{5.14}$$

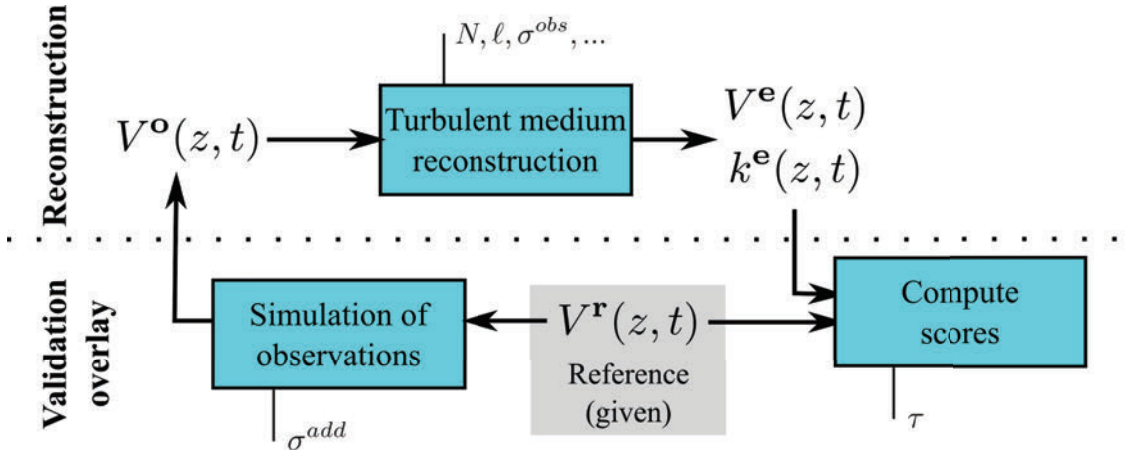


Figure 5.28 – Diagram of the validation experiment allowing to define the output scores.

To quantitatively assess the efficiency of the reconstruction, several validation scores are estimated. Next section intents to introduce them.

5.4.2 Five validation scores

The scores we have been looking at to test the reconstruction system are the following: error (RMSE) on the wind, error (RMSE) on the TKE, number of null potential, wind spectrum slope and execution time. Each of them is now described with more details.

5.4.2.1 RMSE on wind

As we have seen in the section 5.3.1, many information can be provided from the system. But the common root of all these information is the set of particles after selection.

$$\mathbf{V}(\theta,\omega,t) = \begin{pmatrix} \hat{X}^1(\theta,\omega,t), \hat{V}^1(\theta,\omega,t) \\ \vdots \\ \hat{X}^N(\theta,\omega,t), \hat{V}^N(\theta,\omega,t) \end{pmatrix} \tag{5.15}$$

The estimation of the wind, $V^{\mathbf{e}}(\theta,z,t)$, depends on the set of parameters in input θ.

$$V^{\mathbf{e}}(\theta,z,t) = \frac{\sum_{i=1}^{N} \hat{V}^i(\theta,\omega,t)\mathbf{1}_{\hat{X}^i(\theta,\omega,t)\in\mathcal{B}(z)}}{\sum_{i=1}^{N} \mathbf{1}_{\hat{X}^i(\theta,\omega,t)\in\mathcal{B}(z)}} \tag{5.16}$$

Rigorously, because $N < \infty$, the estimation is still a random variable. One should denote $V^{\mathbf{e}}(\theta,\omega,z,t)$. To get rid of the stochastic part of the code, it is assumed N is large enough to use the large number law. Then, the quantity $V^{\mathbf{e}}(\theta,z,t)$ approaches a mathematical expectation:

$$\begin{array}{rcl} V^{\mathbf{e}}(\theta,z,t) & \underset{N\to\infty}{\simeq} & \mathbb{E}\left[V_t | X_t \in \mathcal{B}(z), V^{\mathbf{o}}_{z,t} = V^{\mathbf{o}}(\theta,z,t)\right] \\ & \underset{\Delta z\to 0}{\simeq} & \mathbb{E}\left[V^{\mathbf{r}}_{z,t} | V^{\mathbf{o}}_{z,t} = V^{\mathbf{o}}(\theta,z,t)\right] \end{array}$$

The estimation is the average of all possible realizations of $V^{\mathbf{r}}_{z,t}$ which can lead to the the observation $V^{\mathbf{o}}(z,t)$. As a consequence, $V^{\mathbf{e}}(z,t)$ is not equal to the given reference $V^{\mathbf{r}}(z,t)$, but it is not far neither because $V^{\mathbf{r}}(z,t)$ is precisely "a realization of $V^{\mathbf{r}}_{z,t}$ which can lead to the the observation $V^{\mathbf{o}}(z,t)$". Hence, $V^{\mathbf{e}}(z,t)$ is still compared to $V^{\mathbf{r}}(z,t)$. A criterion of quality of the reconstruction will the wind root mean squared error (RMSE).

Definition 5.1 (Output r_V)**.**

With words *r_V is the estimated root-mean-squared error (RMSE) between the estimated wind and the reference wind.*

Formally

$$r_V(\theta) = \sqrt{\frac{1}{N_z N_t}\sum_{z=1}^{N_z}\sum_{t=1}^{N_t}\left(V^{\mathbf{e}}(\theta,z,t) - V^{\mathbf{r}}(z,t)\right)^2} \tag{5.17}$$

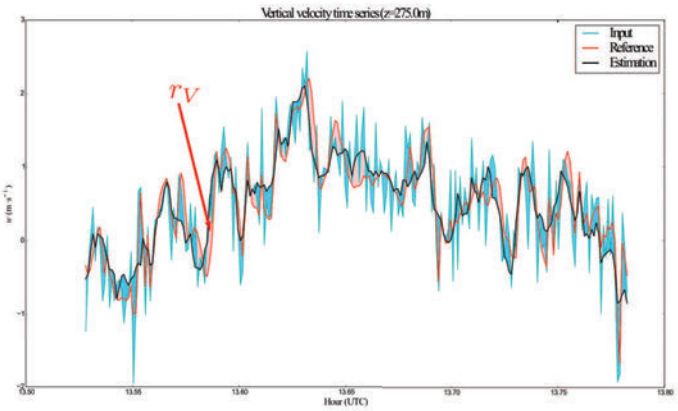


Figure 5.29 – Illustration of RMSE of the wind : one of the output of the system.

5.4.2.2 RMSE on TKE

Turbulent medium reconstruction offers several estimations of wind variance, as seen at the section 1.4. The 1-dimensional TKE is the desired quantity: $k^{\mathbf{r}}(z,t) = \frac{1}{2}V(V_{z,t})$. It is approached by an estimation on the set of particles, $k^{\mathbf{e}}(\theta,z,t)$. In compliance with Baehr (2010); Rottner (2015), the chosen estimation is the LSTKE (definition 1.10):

$$k^{\mathbf{e}}(\theta,z,t) = k^{LS}(\theta,z,t) = \frac{1}{2}\left\langle\left\langle\left(V^i(\theta,z,t) - \left\langle V_t^i(\theta,z,t)\right\rangle^{\ell}\right)^2\right\rangle^{\ell}\right\rangle$$

The problem is that the reference is not accessible and must be estimated too, with an independent and well proven estimation. The time TKE (definition 1.6) has such qualities.

$$k^{\mathbf{r}}(\theta,z,t') = \frac{1}{2}\overline{\left(V^{\mathbf{r}}(z,t) - \overline{V^{\mathbf{r}}(z,t)}^{\tau}\right)^2}^{\tau}$$

However, it requires a time averaging which degrades the time resolution. The LSTKE $k^{\mathbf{e}}(\theta,z,t)$ is defined on the time-scale t (every 4 seconds). The TTKE $k^{\mathbf{r}}(\theta,z,t')$ is defined on the time-scale t' (every τ minutes). Thus the score r_k compares two estimations at different resolution. To fix the difference of resolution, we keep the coarsest resolution for both. That is to say, the time average is applied to the LSTKE before to be compared to the TTKE. The associated score is given by the following definition and illustrated by the figure 5.30.

Definition 5.2 (Output r_k)**.**

With words *r_k is the root-mean-squared error (RMSE) between the guess of TKE from the particles and the guess of TKE from the reference.*

Formally

$$r_k(\theta) = \sqrt{\frac{1}{N_z N_t'} \sum_{z=1}^{N_z} \sum_{t'=1}^{N_t'} \left(\overline{k^{\mathbf{e}}}^{\tau}(\theta, z, t') - k^{\mathbf{r}}(\theta, z, t') \right)^2 dz dt'} \tag{5.18}$$

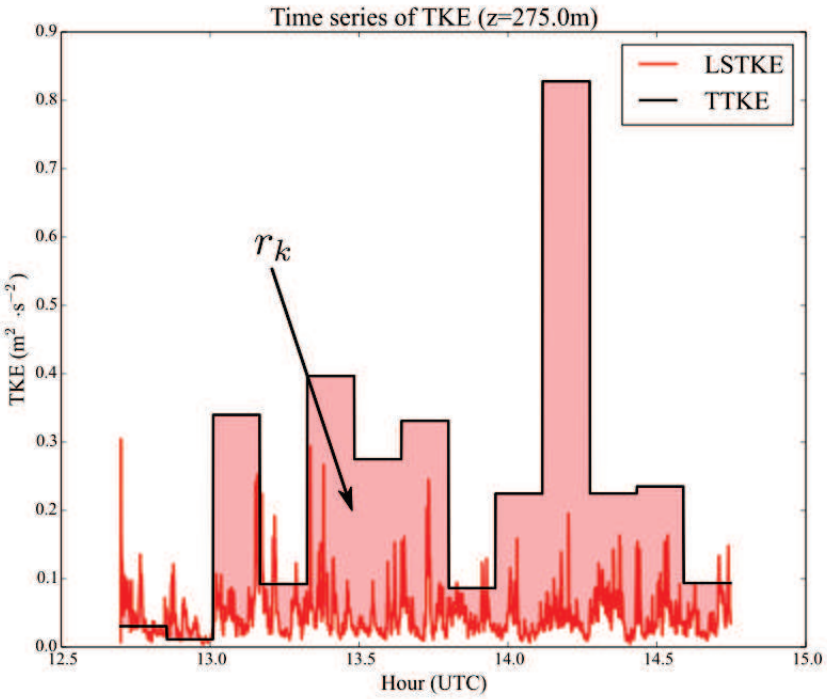


Figure 5.30 – Illustration of RMSE of the TKE : one of the output of the system.

5.4.2.3 Number of null potential

The next score is specific to the filtering process. Particle filters are known to degenerate when the number of particles is small or the dimension is high (see chapter 2 and (Snyder et al., 2008; Bengtsson et al., 2008; Del Moral, 2004)). This degeneration manifests itself by the collapse of the potential of particles at the selection step (potential G_{obs}, equation (5.9)). The problem when this potential collapses is that a large number of particles is resampled on a small number of remaining particles. The cloud of particles turns abnormally concentrated onto few spots. Pushed to its extremity (i.e. when the maximum potential is below the machine precision), this degeneracy yields to cases where all particles have a null potential. This situation is illustrated by the figure 5.31. When it occurs, the resampling cannot be

done: the filtering algorithm crashes. In such cases, it is recommended to reset the cloud of particles (Del Moral, 2004). Although the resetting avoids the crash, it does not cure the problem: reset particles are back to an *ex nihilo* distribution which will need several time steps to represent again the wind. Hence, the occurrences of such degeneracy are tracked. Their number is score to assess the well proceeding of the reconstruction: the lower, the better.

Definition 5.3 (Output N_{G0})**.**

With words *N_{G0} is the total number of occurrences of null potential.*

Formally

$$N_{G0}(\theta) = \left| \left\{ (z,t) / \sum_{i=1}^{N} G_{obs}(\theta, z, t, i) = 0 \right\} \right| \tag{5.19}$$

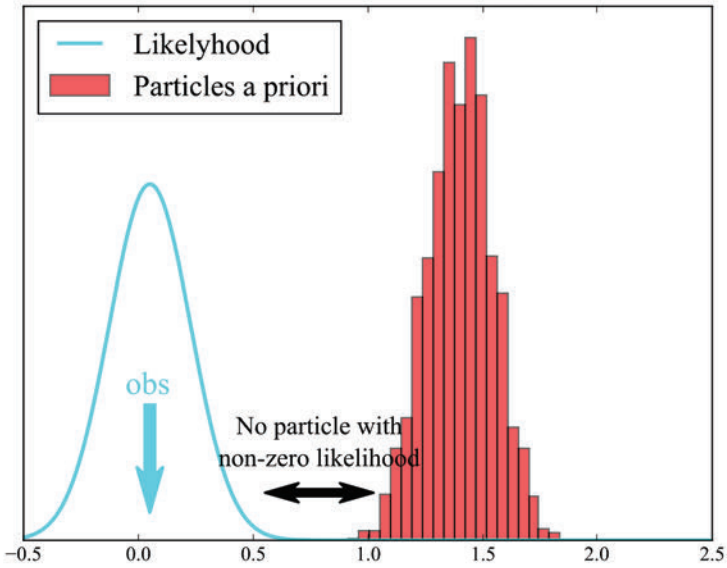


Figure 5.31 – Illustration of null potential: it occurs when particles fairly don't match the measurement.

5.4.2.4 Slope of wind power density spectrum

We have seen in the chapter 1 that the power spectrum density (PSD) of the wind has a -5/3 decrease which is the signature of turbulence. It is required the estimated wind has the characteristics of turbulence. As a consequence, the PSD of the wind is drawn and compared to the reference (figure 5.16). When the PSD is steeper than the reference, the filter takes some frequencies of turbulence as noise and remove them. It results an output signal which

does not look like turbulence. When the PSD is flatter than the reference, the filter does not remove all the noise. The filter is not efficient. Given the easy interpretations coming out from the PSD examination, a score has been built to sum up the information. The PSD slope is estimated by linear regression on the highest frequencies, as illustrated by figure 5.32.

At the vertical level z, the PSD $\mathbf{\Gamma}$ is obtained by taking the squared modulus of the Fast Fourier Tranform of the time series $\{V^{\mathbf{e}}(z,t),\, t \in [\![1, N_t]\!]\}$. The vector $\mathbf{\Gamma}$ has $n = \frac{N_t}{2}$ components, one for each frequency among $\xi = (\xi_1, ..., \xi_n)$ where $\xi_i = \frac{1}{i2\Delta t}$. According to the Kolmogorov theory, the PSD of the wind is of the following form:

$$\Gamma_W(\xi) = C_0 \varepsilon^{2/3} \xi^{-5/3}$$

Hence, we would like to describe the estimated PSD with the model (5.20).

$$\mathbf{\Gamma} = A\xi^b \tag{5.20}$$

Taking the logarithm of (5.20) gives a linear model suitable for an ordinary least squares regression. Nevertheless, the smallest frequencies are not in the inertial domain of turbulence and the slope of the PSD is no longer supposed to be $-5/3$. Hence, the regression is done only on the frequencies above $\xi_c = \frac{1}{100\Delta t}$. This cutting frequency has been chosen after visual identification of the inertial domain on several examples. It results the following definition for the output b.

Definition 5.4 (Output b)**.**

With words *b is the slope of the wind power spectrum density estimated by linear regression on the highest frequencies.*

Formally

$$(a, b) = \underset{(\alpha,\beta)}{arg\min} \left\{ \| \log(\mathbf{\Gamma}_I) - \alpha - \beta \log(\xi_I) \|_2^2,\, I = \{i,\, \xi_i > \xi_c\} \right\} \tag{5.21}$$

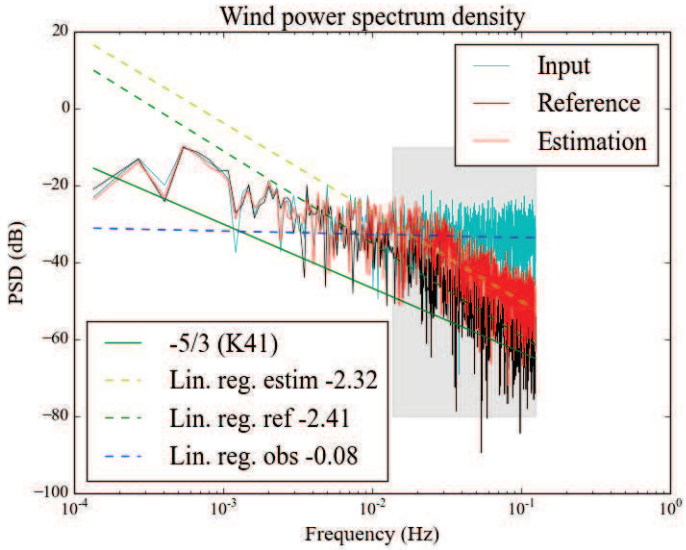


Figure 5.32 – The slope b of the wind PSD is estimated by linear regression on the points in the shaded area.

5.4.2.5 Time of execution

The time of execution T_{exe} has also been added as a criterion of performance. Although it is not related to the quality of the estimation, it is an important practical constraint. It includes the loading of the data and the initialisation, but not the writing on output files neither the calculations on the outputs because these calculations are various and depends on the desired use of the run.

Definition 5.5 (Output T_{exe})**.**

With words *T_{exe} is the time spent for one execution of the reconstruction code.*

Formally

$$T_{exe} = t_{end} - t_{start}$$

5.4.3 Known variations of the outputs

With 9 inputs and 5 outputs, the tested system is rather complex. This is what sensitivity analysis is about: simplify the study of complex systems. Nevertheless, few results exist to describe the influence of a parameter on an output.

5.4.3.1 Influence of N on r_V

The influence of N on r_V is a direct consequence of law of large number. Hence, it is one of the most mentioned in the literature (see Baehr (2008); Del Moral (2004); Baehr (2010) at least). For instance, theorem 3.3 in Baehr (2010) establishes the L^p bound of the Monte Carlo error for this Lagrangian model:

Theorem 5.1 (Influence of N on r_V)**.**

$$\forall p \in \mathbb{N}^*, \forall t \in [\![1, N_t]\!], \exists C_t(p) \in \mathbb{R}, \qquad \mathbb{E}\left[\|\eta_t^N - \eta_t\|^p\right]^{1/p} \leqslant \frac{C_t(p)}{\sqrt{N}} \tag{5.22}$$

Applied with $p = 2$, it tells the score r_V should decrease as $1/\sqrt{N}$. This influence is expected to give a large Sobol index for N over r_V.

5.4.3.2 Influence of N on N_{G0}

The number of null potential decreases exponentially with the number of particles. The proof is in the appendix B.1. It is based on a result from Del Moral (2004), itself coming from Azuma-Hoeffding's inequality.

Theorem 5.2 (Influence of N on N_{G0})**.** *If $N > 1$ is the number of particles and at any time there are particles in the probe volume, for any $n \in \mathbb{N}$, there exist $\alpha(n) > 0$ and $\beta(n) > 0$ such that*

$$\mathbb{P}\left(N_{G0} = n\right) \leqslant \alpha(n) e^{-N/\beta(n)} \tag{5.23}$$

5.4.3.3 Influence of σ^{obs} and σ^{add} on N_{G0}

Under some conditions, it is possible to have an approximation of the influence of σ^{obs} and σ^{add} on N_{G0}.

Theorem 5.3 (Influence of σ^{obs} and σ^{add} on N_{G0})**.** *If the following assumptions are satisfied*

- *The real wind $V^{\mathbf{r}}_{z,t}$ is stationary at order 2 and ergodic in space and time.*
- *The particles after conditioning $\tilde{V}_t$ are Gaussian with the same mean and variance as $V^{\mathbf{r}}_{z,t}$.*

Then, the average number of null potential N_{G0} is bounded from above by a function of σ^{obs} and σ^{add}.

$$\mathbb{E}\left[N_{G0}\right] \leqslant N_t N_z \frac{(\sigma^{add})^2 + 2k}{-\left((\sigma^{obs})^2 + 2k\right) \log\left(\iota^2 2\pi \left((\sigma^{obs})^2 + 2k\right)\right)} \tag{5.24}$$

with $\iota = 10^{-16}$, the zero machine threshold, and $k = \frac{1}{2}V\left(V^{\mathbf{r}}_{z,t}\right)$ (constant thanks to the stationarity assumption).

The proof is in the appendix B.1. This result tells the average value of N_{G0} is bounded by a known function of σ^{obs} and σ^{add}. This bound is displayed in figure 5.33 for $k = 0.05$ and $\iota = 10^{-16}$. One can see that the bound is very low for a large range of values. It increases rapidly when σ^{add} is high and σ^{obs} is low. In the figure 5.31, this situation would be represented by a narrow histogram and an observation (cyan Gaussian) likely to be far from it, which makes sense.

The assumptions to fulfil are strong, but usual. The assumption of stationarity is not as strong as it seems, given that the TKE k is usually estimated over 10, 20 or 30 minutes while the lidar measures every 4 seconds. One can assume the variations of k are small enough to be neglected at that scale. The assumption of ergodicity is the strongest. Here as in many occasions, it is done in spite of something better. The assumption that the particles follow a Gaussian of mean $\mathbb{E}\left[V^{\mathbf{r}}_{z,t}\right]$ and variance $2k$ implies the particles sample the full probability law of $V^{\mathbf{r}}_{z,t}$ while in reality they are conditioned to the probe volume and the past observations.

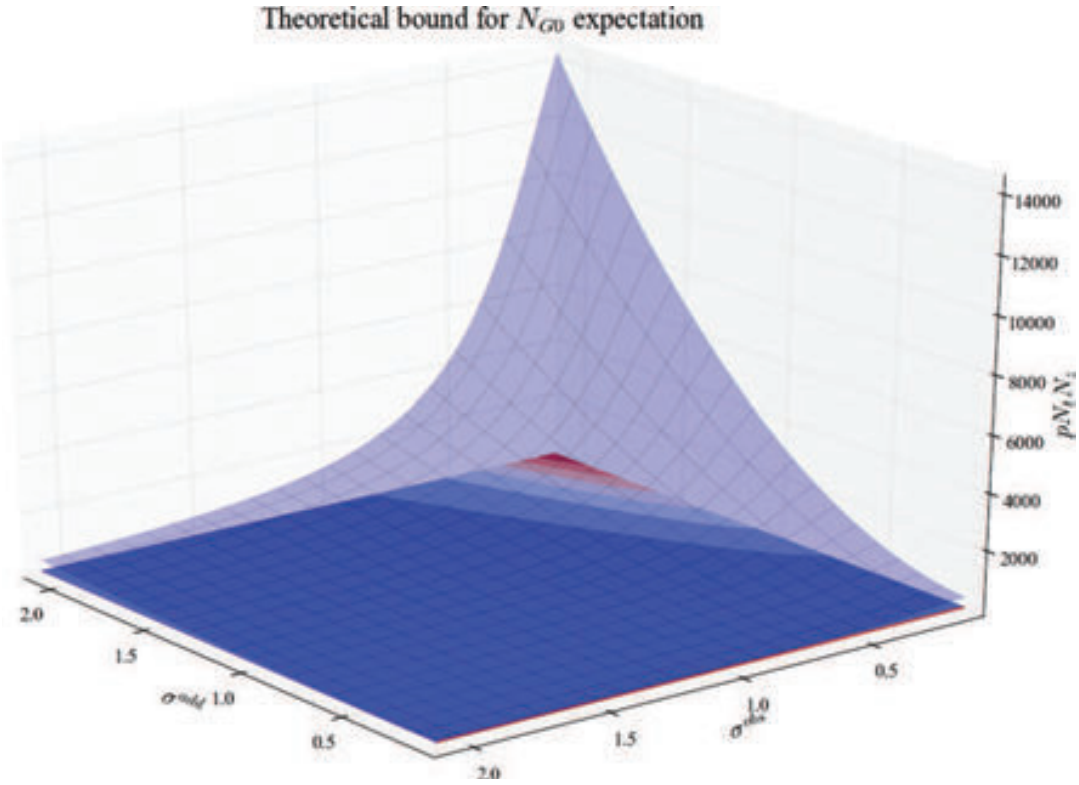


Figure 5.33 – Theoretical bound for the average of N_{G0} against σ^{add} and σ^{obs}.

5.5 Conclusion

This chapter is at the core of this manuscript, between the state-of-the art chapters and the results chapters. It presents the reconstruction system with a high level of details and a particular emphasis on pedagogy. After an introduction with the motivation and notations, the four steps of the algorithm are described (role, inputs, outputs, comments). Then a small section is dedicated to developers intending to implement the reconstruction method. Finally the validation experiment and the resulting scores are presented (diagram, definitions, theoretical results).

Turbulent wind is a stochastic process. A single realization of it happens and is measured by a noisy instrument. The reconstruction system aims to retrieve the turbulent wind that gave such observations. It is composed of four steps: mutation, conditioning, selection and estimation. The mutation step pushes the state vector (set of particles) from one time step to the next. The conditioning step fix the problematic case due to the Lagrangian nature of particles and the Eulerian nature of the measure (that is to say, it treats particles moving away from the instrument and big wads/gaps of particles). The selection step includes the observation in the system of particles by a *genetic selection* algorithm (acceptance/rejection followed by *sampling with importance resampling*). The estimation step updates the diagnostic variables needed for the next mutation step and computes the outputs. It is the only step which let the state vector unchanged. These four steps form a cycle (see figure 5.14) repeated

at each time step, up to the end of measurement.

Many quantities are worth to look at in the system of particles. Some examples are given. From the long hours spent into the implementation, a concise set of checkpoints have been established. It is recalled that the actual reconstruction system is not suited for stable atmospheric conditions. To validate the reconstruction system, an experiment has been set up and quality scores have been defined. The experiment is based on a reference wind. Some noise is added to this reference wind to get simulated observations. The reconstruction method is applied to the simulated observations. Then the output of the reconstruction is compared to the reference wind. Five scores are computed: the root-mean-squared error (RMSE) of the wind, the RMSE of the wind variance, the number of null potential, the wind spectrum slope and the execution time. From the construction of such scores, one can derive some theoretical formulae linking some parameters of the reconstruction to the outputs. The RMSE on the wind decreases as a square root of the number of particles. The number of null potential decreases as an exponential of the number of particles. The average number of null potential is bounded by a function of the added noise and the noise given to the filter.

Chapter 6

Results of the sensitivity analysis

Contents

6.1 Methodology

6.1.1 System, inputs and outputs

6.1.1.1 System on which sensitivity analysis is done

The system on which the sensitivity analysis is done is a computer code function f that has some input arguments ($X \in \mathbb{R}^p$) and provides some outputs ($Y \in \mathbb{R}^s$).

$$Y = f(X)$$

We restrict ourselves to consider the s outputs are independent. Each of the s outputs is treated separately, so that it is equivalent to the case $Y \in \mathbb{R}$, repeated s times. The sensitivity analysis is carried out on a system allowing the assessment of the reconstruction. It encloses the reconstruction system, but not only. In particular, a reference wind $V^{\mathbf{r}}(z,t)$ is included to compute the output score. The diagram of this system is given on figure 6.1. It has been described with more details in the section 5.4.1.

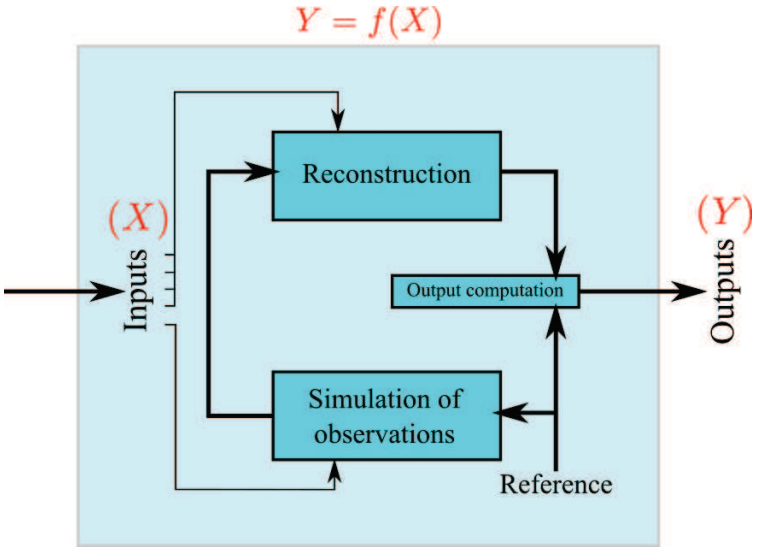


Figure 6.1 – Diagram of the system on which is done the sensitivity analysis.

The inputs have been introduced throughout the the previous chapter. The figure 6.2 reviews the organisation of the computer code and locete the input parameters. At the end of this chapter, the outputs are described one by one. Let us briefly recall the system here.

6.1.1.2 Inputs of the system

The inputs of the diagram 6.1 are the parameters listed in the boxes 5.7, 5.10, 5.12 and 5.13 of the previous chapter. There are gathered here only in a concise way since they have been introduced in the previous chapter.

- C_0 : Kolmogorov "constant".

  This constant appears in the expression of the -5/3 spectrum and in the dispersive term of the Lagrangian model.

- C_1 : The fluctuation coefficient.

  This constant in the Lagrangian model pilots the fluctuation term. According to Pope (1994), there is an empirical relantionship with C_0. To test if such relationship is retrieved by the sensitivity analysis, C_1 is considered as separated input.

- ℓ : Spatial interaction length.

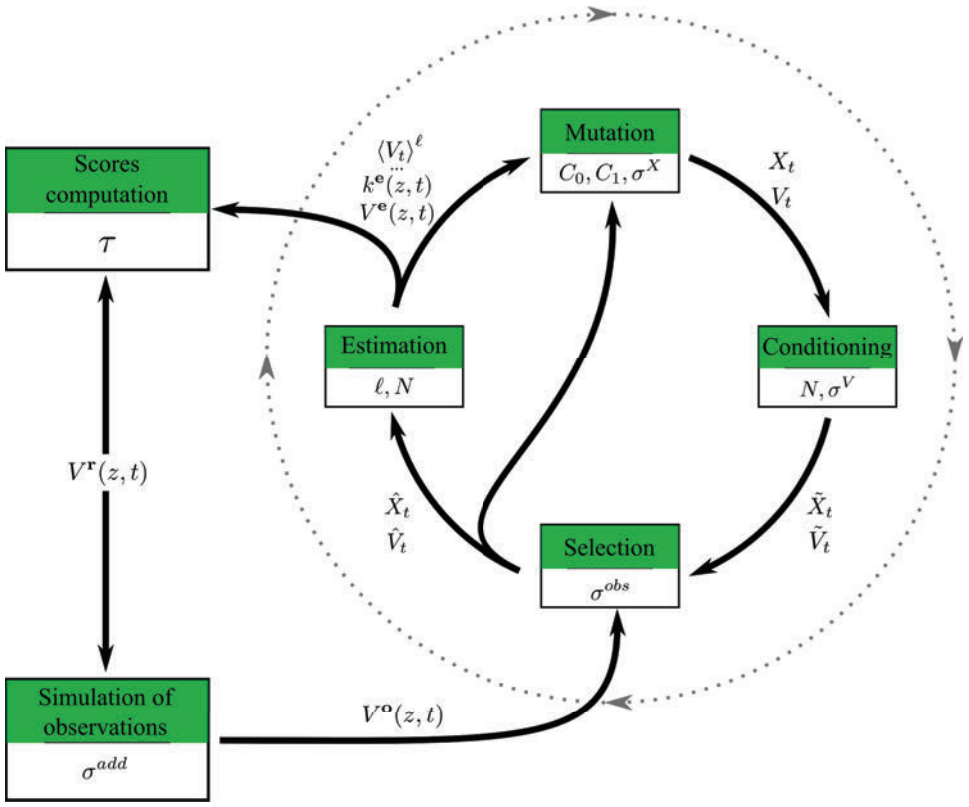


Figure 6.2 – Situation of the input parameters in the program. The dotted outer circle stand for the time loop in the reconstruction algorithm.

The estimation of Eulerian average with Lagrangian particles requires the use of a regularisation kernel of parameter ℓ. The smaller ℓ, the faster the kernel decreases. Thus, ℓ is homogeneous to an interaction length.

- N : Total number of particles.

  Both the filtering algorithm and the Lagrangian model rely on Monte Carlo approximation of the wind probability law. The total number of particles (including all vertical levels) is N.

- σ^{add} : True observation noise.

  The instrument is supposed to make a centred Gaussian error. The observations are simulated by adding a Gaussian error of standard deviation σ^{add} to the reference wind.

- σ^{obs} : Observation noise given to the filter.

  In practice, the observation noise is set by the instrument and may be unknown. Nevertheless, the filtering algorithm needs a guess for it. This guess is a centred Gaussian observation noise with standard deviation σ^{obs}.

- σ^V : Default standard deviation of wind speed.

To avoid concentration of particles on the exact same point during the resampling stages in the conditioning step, a centred Gaussian of standard deviation σ^V is added to the new speed of resampled particles.

- σ^X : Standard deviation of discretization error in the Lagrangian model.

  The Lagrangian model is a stochastic differential equation which is discretized in order to be solved. The error introduced by this discretization is assumed to be a centred Gaussian of standard deviation σ^X and added to the position equation.

- τ : Integration time for TKE calculation.

  The TKE is usually estimated by the variance of the wind time series over a time τ (about 15 minutes). The reconstruction provides an estimation of TKE from spatial variance every 4s. They are averaged over τ to be compared.

In the end, we count 9 input parameters that are summarized in table 6.1. Hence the sensitivity analysis model, p equals 9 and X belongs to $\mathbb{R}^9$. The sensitivity analysis spots the impact of a change of an input.

Notation	**Description**	**Place in the system**
C_0	Kolmogorov "constant"	Reconstruction (Lagrangian model)
C_1	Fluctuation coefficient	Reconstruction (Lagrangian model)
ℓ	Spatial interaction length	Reconstruction (Lagrangian model)
N	Number of particles	Reconstruction (all steps)
σ^{add}	True observation noise	Simulation of observation
σ^{obs}	Guess of observation noise	Reconstruction (filtering)
σ^X	Discretization error in the Lagrangian model	Reconstruction (mutation step)
σ^V	Default standard deviation of wind speed	Reconstruction (conditioning step)
τ	Integration time	Output computation

Table 6.1 – Summary of input parameters for the sensitivity analysis

6.1.1.3 Outputs of the system

Five outputs considered independently are used to qualify the system. They have been described in section 5.4.2. They are recalled in the table 6.2. The sensitivity analysis aims to identify which inputs are to be changed in order to modify the outputs.

6.1.2 Organisation of the results

To pull out the maximum benefit of sensitivity analysis, an efficient graphical representation is essential. This is why many effort has been done here to provide a wide panel of visualization. The paper of Iooss and Lemaître (2015) was a fruitful source of inspiration. For each of the

Notation	Description	Definition equation	Illustration
N_{G0}	Number of null potential	$N_{G0} = \lvert\{G = 0\}\rvert$	5.31
b	Slope of the wind PSD	$\Gamma(\xi) = A\xi^b$	5.32
r_k	Root-mean-squared error on the TKE	$\lVert\overline{k^{LS}(z,t)}^{\tau} - k^T(z,t')\rVert_2$	5.30
r_V	Root-mean-squared error on the wind	$r_V = \lVert\hat{V}_{z,t} - V_{z,t}^{ref}\rVert_2$	5.29
T_{exe}	Time of execution	$T_{exe} = t_{end} - t_{start}$	

Table 6.2 – Summary of output parameters for the sensitivity analysis

5 outputs, an independent sensitivity has been carried out. The inputs are the same. The sensitivity analysis provides:

- First order main effect Sobol indices estimates $\widehat{S_i}$ (color key: blue)
- First order total effect Sobol indice estimatess $\widehat{S_{Ti}}$ (color key: green).
- Uncertainty on the estimation of first order Sobol indices.
- Second order Sobol indices estimates $\widehat{S_{ij}}$ (color key: red).

The estimation of Sobol indices is performed with (Saltelli et al., 2010) estimator for the first order indices (main and total effect). For the second order Sobol indices, the estimator from (Saltelli, 2002a) is used. Further details about the diffences of estimations is in section 4.3.2, especially in table 4.1.

First, the sensitivity analysis are commented separately, one output after another. The figures representing these information are explained in the next subsection ("Analysis of figures presented"). For each output, the way it is influenced is examined and commented. The quality of an output is assessed by the clarity of its response: an output complexly influenced by many inputs with many interactions will not have a clear response, while an output influenced directly by few inputs is more informative.

Second, the outputs are gathered to have a more global point of view. The Sobol indices from the five sensitivity analysis are summed to produce the same figures, thus representing the system broadly. The Sobol indices are also brought face to face on a tile in order to identify what input influences what output with how much interactions. Finally, the inputs are grouped by importance.

6.1.3 Analysis of figures presented

For each output, the same figures are commented:

- Bar chart to visualize the Sobol indices and their uncertainty

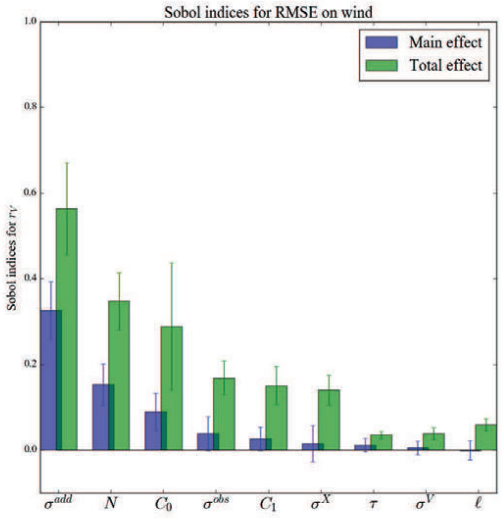
- Pie chart to visualize the share of first and second order (and higher ?)
- Tile to visualize the second order Sobol indices
- Graphs to visualize interactions and important inputs
- Cobweb plots to visualize some effects directly on the response surface

The table 6.3 recaps what information each figure is showing. Each figure is commented in its caption. The most interesting features are commented in the text at the beginning of the section.

Information → **Figure type ↓**	1st order Sobol indices	2nd order Sobol indices	Total Sobol indices	Uncertainty on estimation	Initial response surface
Bar chart	•		•	•	
Pie chart	•	•			
Tile of second order	•	•			
Graphs of interaction	•	•	•		
Cobweb plots					•

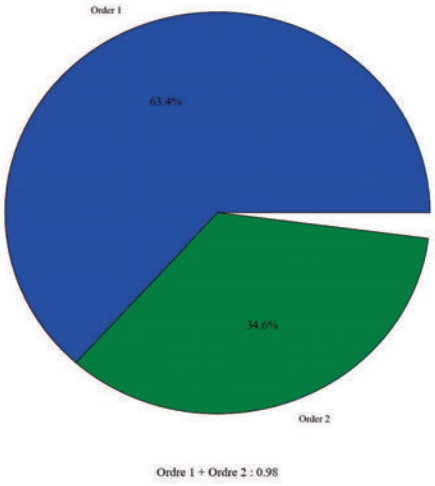
Table 6.3 – Information displayed on each figure type.

6.1.3.1 Bar chart of first order Sobol indices



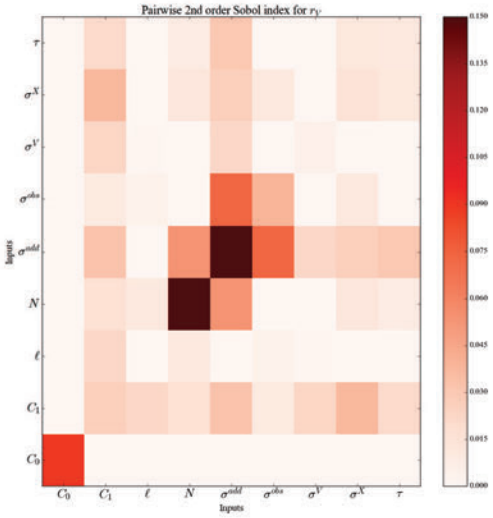
For each output, a bar plot gives the first order Sobol indices of all inputs. Blues bars show the main effect Sobol index of the corresponding input: it represents the influence of this input *alone* on the output. Green bars show the total effect Sobol index of the corresponding input: it represents the influence of this input *including all its interactions* on the output. The error ticks and the extremity of the bars stand for the 95% confidence interval of the estimation. Inputs are sorted according their main effect Sobol index.

6.1.3.2 Pie chart of first and second order Sobol indices



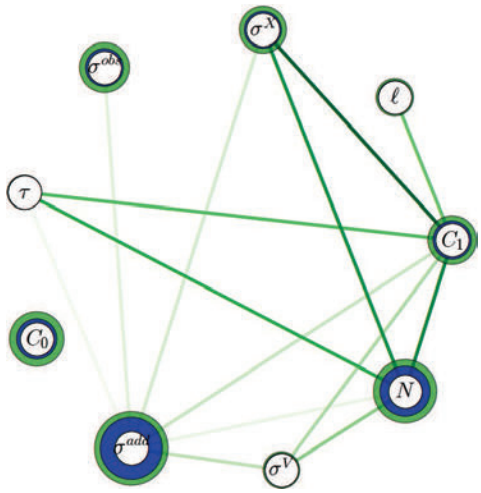
The sum of first order (main effect) Sobol indices stands for the proportion of variance explained by a single inputs. The sum of second order Sobol indices quantifies the importance of pairwise interaction. The remaining part of variance is attributed to higher order interactions. Total Sobol indices are not used in this figure. This simple chart is used to tell apart simple influence from high order interaction influence. Simple influence can be used to tune the system, while high order interactions imply side-effects.

6.1.3.3 Tile of second order Sobol indices



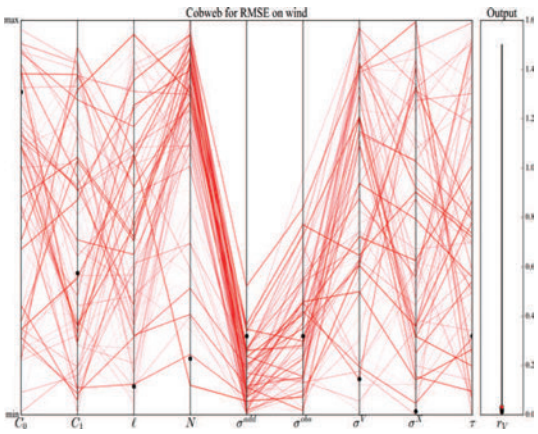
On a tile crossing the inputs, the second order Sobol indices are represented in shades of red. Each cell is dedicated to the Sobol index of the interaction between the two inputs in abscissa and ordinate. The more coloured is a cell in the tile, the stronger is the interaction between the two inputs. The tile is thus symmetric. On the diagonal, the shade is proportional to the first order (main effect) Sobol index.

6.1.3.4 Graphs of interaction



Graphs of interaction are inspired from Fruth et al. (2013, 2014). Vertices are inputs of the system. Edges connect two inputs which have a strong enough pairwise interaction (an edge is displayed if the second order Sobol index is above a 0.02 threshold). The edges are shaded according to the second order Sobol index: small interactions vanish compared to the strongest ones. On the vertices, two concentric rings are displayed. The inner ring (blue) has a width proportional to the main effect Sobol index. The outer ring (green) has a width proportional to the total effect Sobol index. By crossing first order, second order and total Sobol indices, the graphs also give hints about the amount of higher order interactions. On the example on the left, C_0 has a large green outer ring but no edges linked to it: C_0 interacts with more than one input. The overall graph gives an overview of influencing spots and their interaction.

6.1.3.5 Cobweb plots



Kurowicka and Cooke (2006) introduce cobweb plots in the visualization chapter of their book (see p.193). It represents only the response surface of the code f, before the meta-modelling and the sensitivity analysis. This graph should be read as follows: the last bar on the right show the output value (sometimes in log-scale). It is either the 100 highest values or the the 100 lowest values. For each value of output, the corresponding input parameters are linked by a thread (there is one vertical bar per input and all inputs are normalized to be inside $[0, 1]$). The overall picture reveals if there is any pattern leading high/low output values. On the example on the left, the 100 lowest outputs are displayed. The threads show it is associated to low values of σ^{add} and σ^{obs} and high values of N. Thus such inputs are expected to be influential and the cobweb plot also give the "sign" of their influence. As this graph depicts only the response surface, it does not suffer from meta-modelling approximation neither from Sobol indices estimation.

6.2 Output by output analysis

To start the presentation of the results, the five sensitivity analysis are taken separately. For each output, the influential inputs are identified, their level of interaction is commented. Since the outputs are scores to assess the reconstruction, they can also be used to tune the input parameters to make the reconstruction better performing. The sensitivity analysis help to discuss this opportunity.

6.2.1 Influence on number of null potential

The number of null potential (denoted N_{G0}) is the symptom of a problem when it is large. Full description is given is the previous section.

The bar chart 6.3 is sorted according the main effect Sobol indices (blue bars). The total Sobol indices (green bars) have a very different order. It points out a large importance of interactions which are not involving the inputs influencing alone. The large weight of interactions is sustained by the pie chart 6.4 (76% of influence is due to pairwise interactions).

One can see that the most influencing input, both in main and total effect, is the *a priori* standard deviation of the noise σ^{obs}. It is the highest bar in figure 6.3. The influence of σ^{obs} is for half direct, for half through interactions. The tile 6.5 shows these interactions are with C_0, C_1 and ℓ.

Next in the figure 6.3 is σ^{add}. It has the second largest first order Sobol index. But it interacts less than C_0 and C_1, which makes its total Sobol index only the 4^{th} largest. These interactions are also with C_0, C_1 and ℓ, according to the graph 6.6.

The standard deviation of discretization error σ^X has also a quite large first order Sobol index. It is also involved in many pairwise interactions (especially with C_0 and C_1, σ^V on a second hand).

The next two parameters in 6.3 are C_0 and C_1. Although their first order Sobol index is quite small, their total Sobol index is large. It means they are not influential alone, but mostly through interactions. These interactions are mostly pairwise, as one can check on the tile 6.5 and the graph 6.6. In particular, C_0 has strong pairwise interaction with all other inputs.

It is notable that N seems to have a very small influence on the number of null potential, although there is a theoretical result to describe it. Either it points out a lack in the sensitivity analysis (this influence should be visible in Sobol indices and is not), either the influence of N on N_{G0} is weaker than other influences (because N is large enough for example). To sort this out, the cobweb plot is useful. One can see in the figure 6.7 that no scheme is visible, neither for N, nor for any input. For the 100 highest number of null potential, one can see in 6.8 that is associated to small C_0, high C_1, high σ^{add} and low σ^{obs}. In particular, N does not appear to be influential. Cobweb plots does not rely on the sensitivity analysis results,

but only on the direct response of the system to a sample of inputs. It proves that the Sobol index of N is due to the small influence of the mechanism describe by theorem 5.2 compared to others. An explanation for this small influence is that N is always large enough to ensure the filter does not degenerates.

The sensitivity analysis of N_{G0} shows this output is influenced mostly by pairwise interactions between inputs. Despite few inputs have a remarkable influence alone (σ^{obs} especially) it is not enough to master this output. The number of null potential is a hardly controllable output. It will not be used as a tuning output.

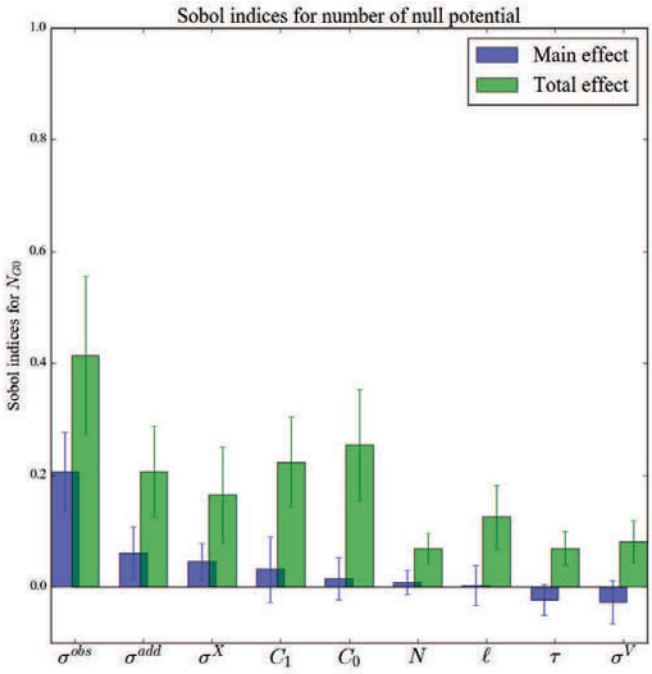


Figure 6.3 – First order and total Sobol indices for the number of null potential with 95% confident interval.

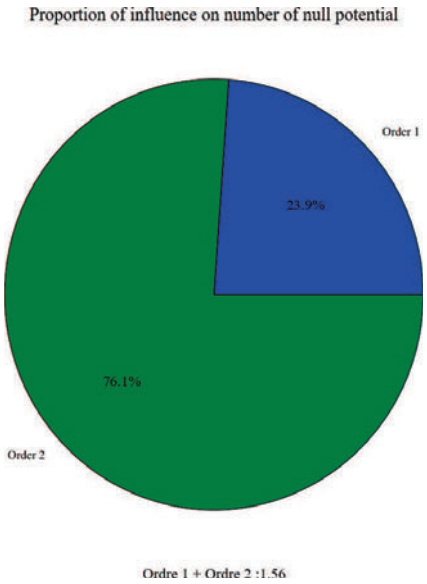


Figure 6.4 – Proportion of 2nd and 1st order for the number of null potential. Most of the influence is due to pairwise interaction. A quarter is due to direct effect. Almost none is due to higher order.

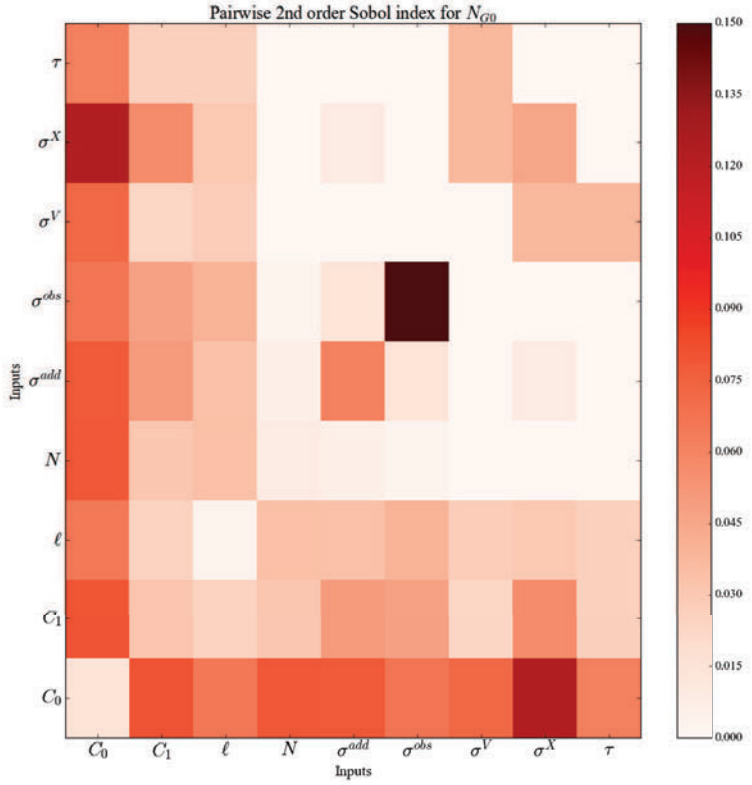


Figure 6.5 – Tile of 2nd order Sobol indices for the number of null potential. Many pairwise interactions are influencing the output. In particular C_0 interacts with almost every inputs but has a very little influence alone. Conversely, the influence of σ^{obs} is mostly direct.

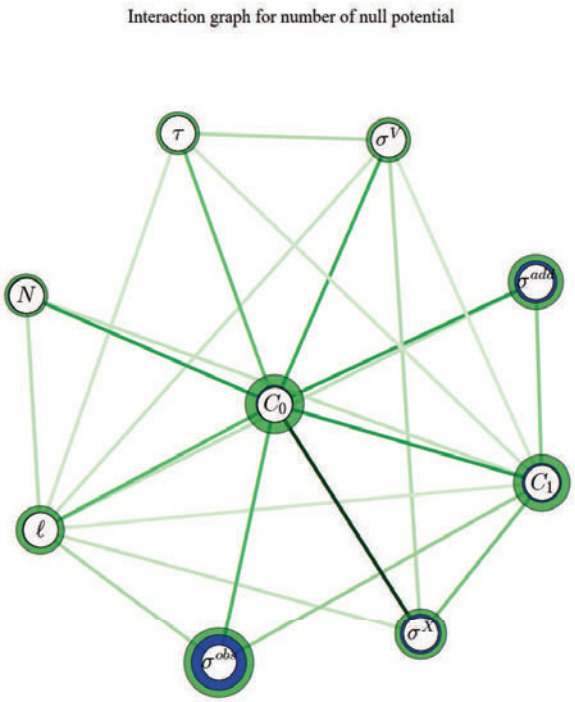


Figure 6.6 – Graph of interaction for the number of null potential. Pairwise interaction are strong and numerous. C_0 interacts with almost every inputs, especially with σ^X. The difference between total and simple Sobol index is mostly due to second order interaction, not higher.

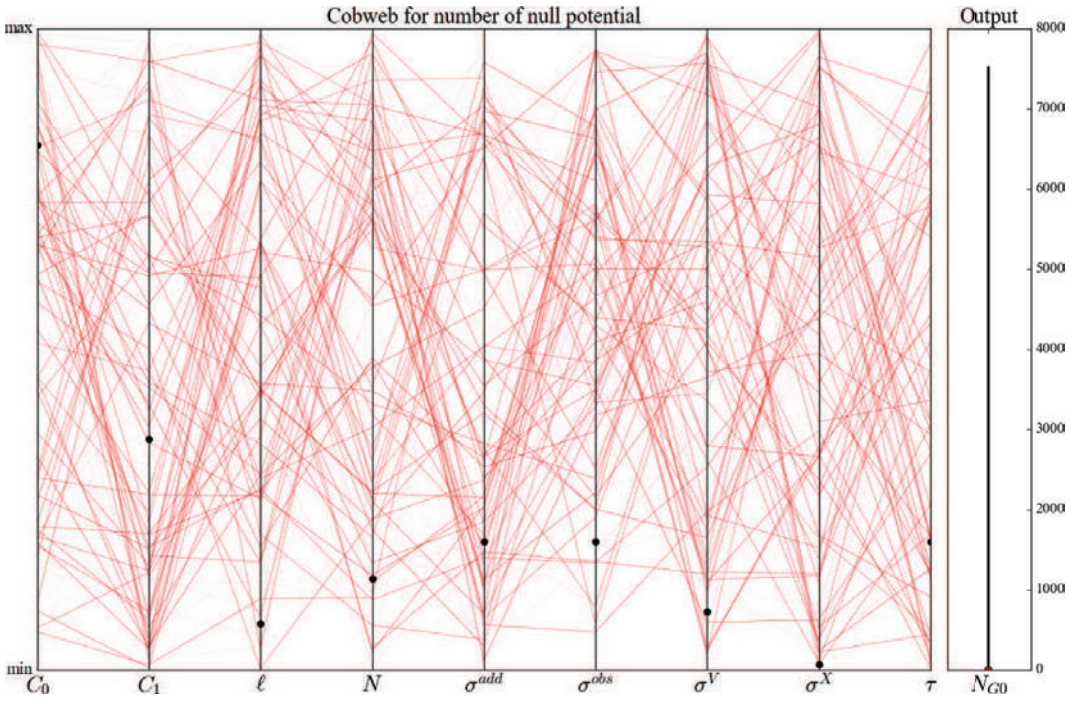


Figure 6.7 – Cobweb plot for low number of null potential. No scheme is visible.

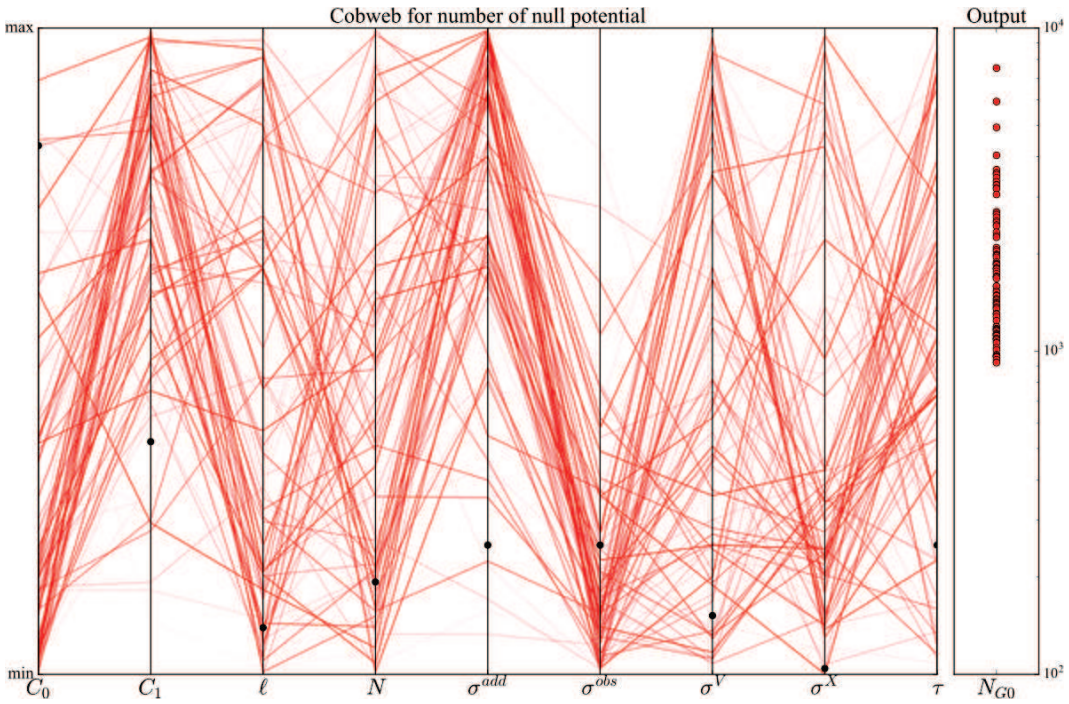


Figure 6.8 – Cobweb plot for high number of null potential (N_{G0} in log-scale). It is high for low C_0, high C_1, high σ^{add}, low σ^{obs}.

6.2.2 Influence on PSD slope

The power spectrum density (PSD) of the wind has a characteristic -5/3 slope in log-log scale when the wind is measured in inertial turbulence (see figure 5.32 for an illustration).

In figure 6.9, one can see that total Sobol indices estimates are close to simple Sobol indices estimates. It indicates few interactions are at stake for this output. It is confirmed by the pie chart 6.10 where direct influences account for 86% of the total. Pairwise interactions are very weak, as it is proved by the second order Sobol indices tile 6.11 and the graph of interactions 6.12. The influence is massively direct and only few parameters are influential. Only four inputs have a significant influence, the others are very weak. The most influential is σ^{add}. It is closely followed by σ^{obs}. And then by C_0 and C_1, which have almost equal influences. Such influence are visible on the cobweb plots 6.13 and 6.14. Low wind spectrum slope occur with high C_0, low C_1, low σ^{add}, high σ^{obs} (see figure 6.13), while high values for the wind spectrum slope are given by low C_0, high C_1, high σ^{add}, low σ^{obs} and small σ^X (see figure 6.14). It confirms the major influence of σ^{add} and σ^{obs} first, C_0 and C_1 next, but also points out an asymmetric influence of σ^X, probably due to high order interaction. The rest of the inputs have a negligible influence.

The wind spectrum is only affected by σ^{add}, σ^{obs}, C_0 and C_1. Their influence is mostly direct, interactions are very weak. Hence, the wind spectrum is easily controllable. It can be used to tune the inputs.

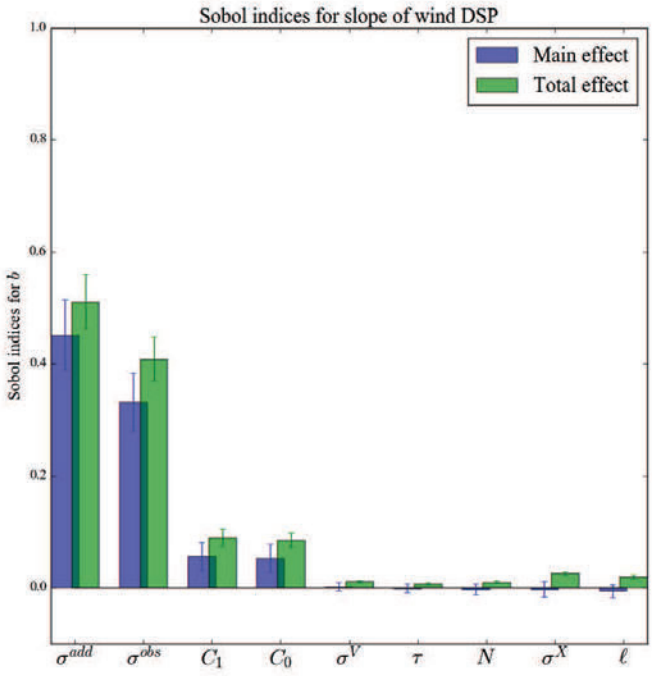


Figure 6.9 – First order and total Sobol indices for the wind spectrum slope with 95% confident interval.

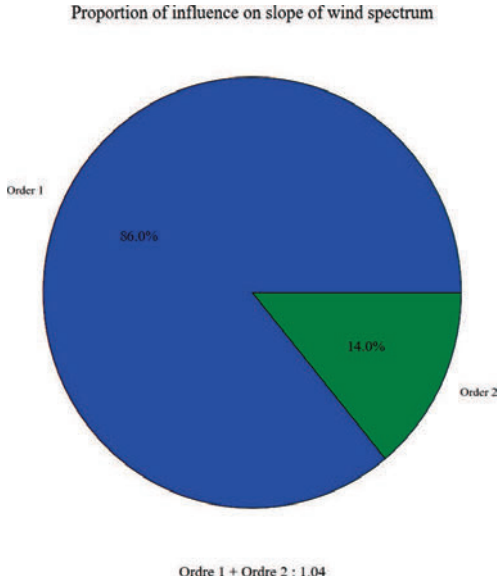


Figure 6.10 – Proportion of 2nd and 1st order influence for the wind spectrum slope. Most of the influence is due to direct effect (86%). Very little is due to pairwise interaction (14%). Order 1 and 2 totalize 1.01 over 1. Even with uncertainty, higher order interactions are negligible.

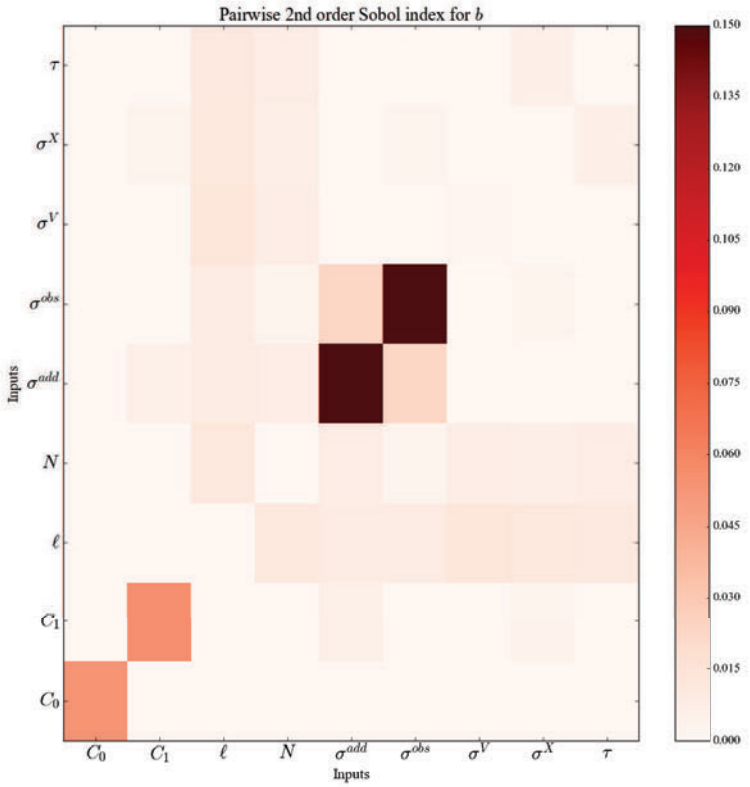


Figure 6.11 – Tile of 2nd order Sobol indices for the wind spectrum slope. Almost no pairwise interaction. Four inputs have a direct influence and that is all.

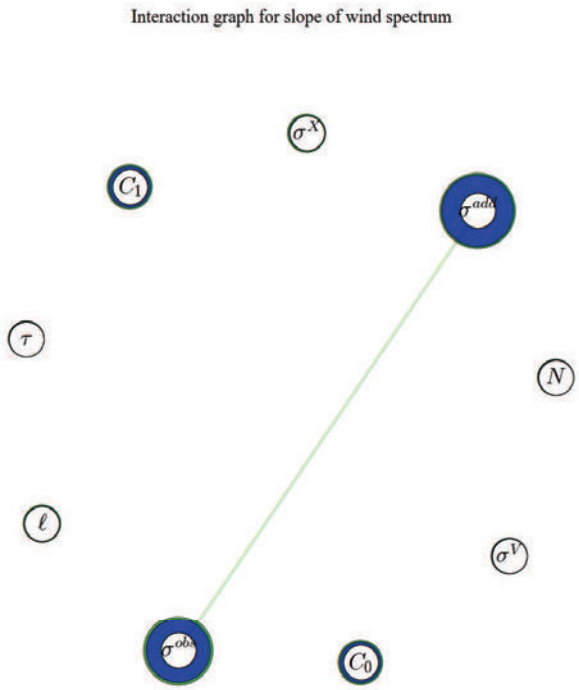


Figure 6.12 – Graph of interaction for the wind spectrum slope. Almost no pairwise interaction (only a weak interaction between σ^{add} and σ^{obs}). No trace of higher order interaction.

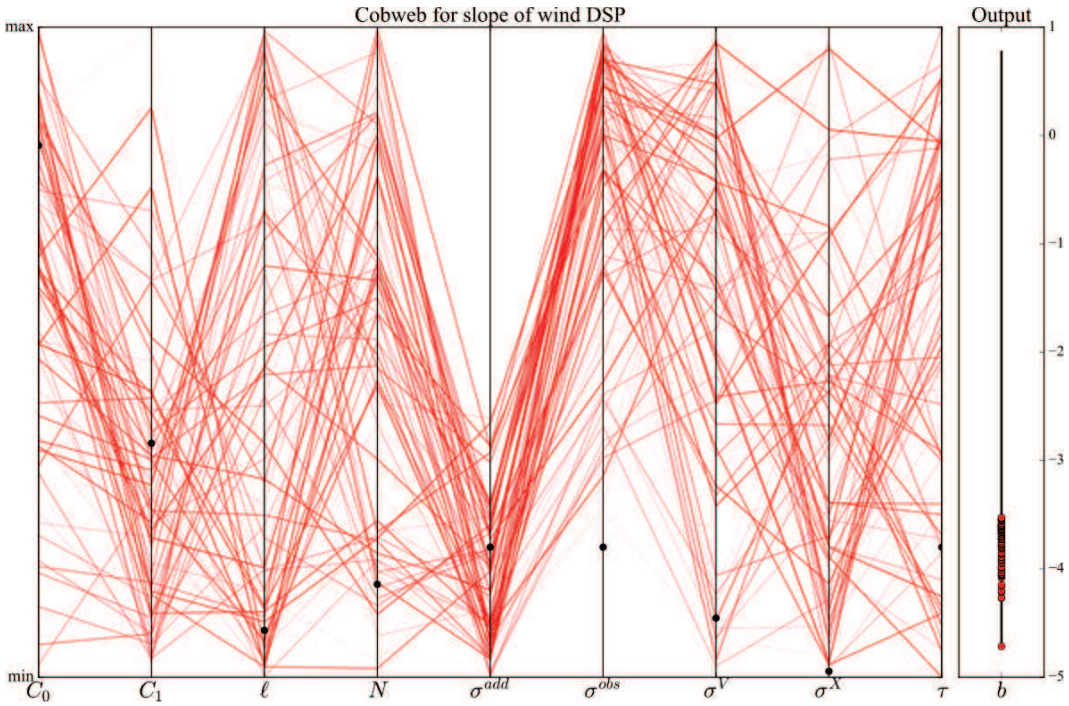


Figure 6.13 – Cobweb plot for low wind spectrum slope. It is associated to high C_0, low C_1, low σ^{add} and high σ^{obs}.

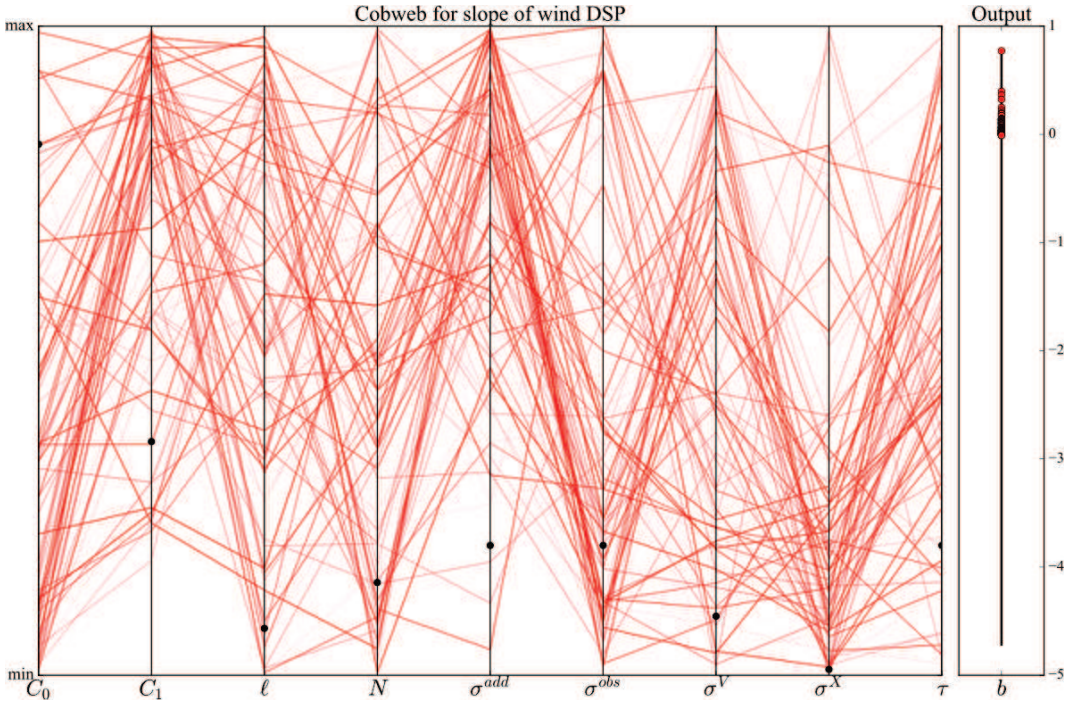


Figure 6.14 – Cobweb plot for high wind spectrum slope. It is associated to low C_0, high C_1, high σ^{add}, low σ^{obs} and small σ^X.

6.2.3 Influence on TKE RMSE

The RMSE on TKE (r_k) is a score to compare the fast TKE estimation with the usual way of computing TKE. Although its construction is complex, it is expected to have an exploitable behaviour. That is to say, it is expected to be small when the reconstruction performs well.

A look at the first order bar chart 6.15 shows that the main effect Sobol indices are very small for this output. The most influential input alone is σ^{add}, with a first order Sobol index of only 0.094. It is followed by C_1, C_0 (about 0.05) and ℓ (about 0.037). Most of the influence is due to interactions, as confirmed by the large gap between simple and total Sobol indices. What displays the pie chart 6.16 is that these interactions are not only pairwise, but at 27% of higher order. Conversely to other outputs, a large part of the variability of r_k cannot be explained with the experiments carried out in this thesis. The cobweb plot 6.19 for low r_k is very fuzzy, which sustains the available information are limited. RMSE on TKE seems to be low when C_0 and C_1 are different, σ^{add} is rather small and τ is small. Nevertheless, pairwise interactions are important in number (11 edges visible on the graph 6.18) and in strength (in figure 6.17, they have a shade comparable to the first order indices on the diagonal). None input is left alone. The cobweb plot 6.20 for high r_k displays visible features. RMSE on TKE is high when C_0 is low, C_1 is high, ℓ is high and σ^{add} is high. There are not the inverse of what the cobweb plot for low r_k displays. For example, τ needs to be low to provide low r_k but does not need to be high to provide high r_k. It highlights an asymmetric influence of τ and others.

The RMSE on TKE has very complex variations. It is poorly influenced by inputs alone, but mostly by inputs in interactions, including a large part of high order interactions. As a consequence, this output is very difficult to control. It cannot be trusted to tune the inputs.

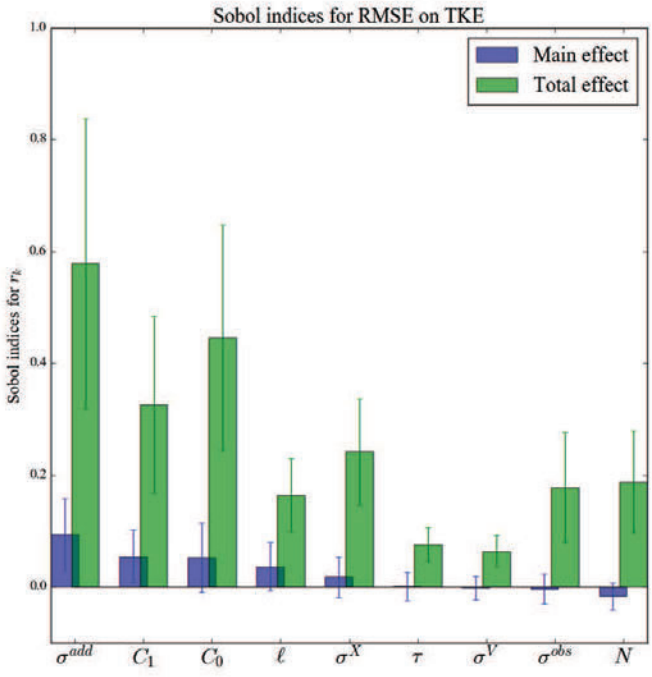


Figure 6.15 – First order and total Sobol indices for the RMSE on TKE with 95% confident interval.

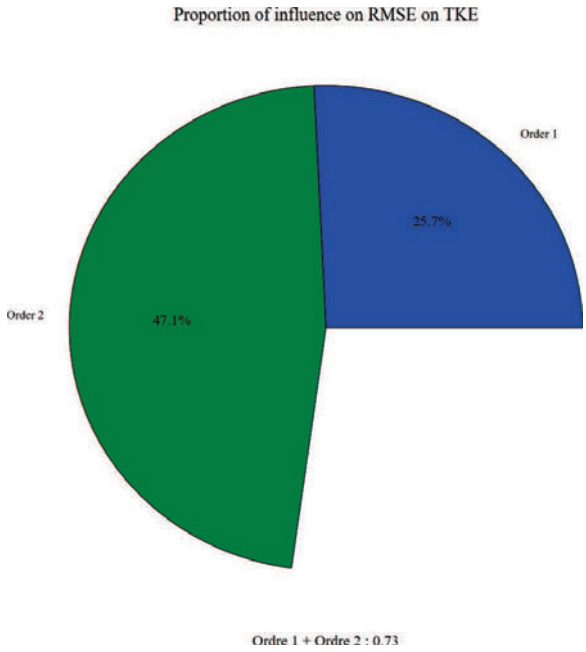


Figure 6.16 – Proportion of 2nd and 1st order for the RMSE on TKE. Only 25% of variance is due to direct influence. Second order interactions have a large share (about 47 %). But it leaves about 27% to higher order interactions.

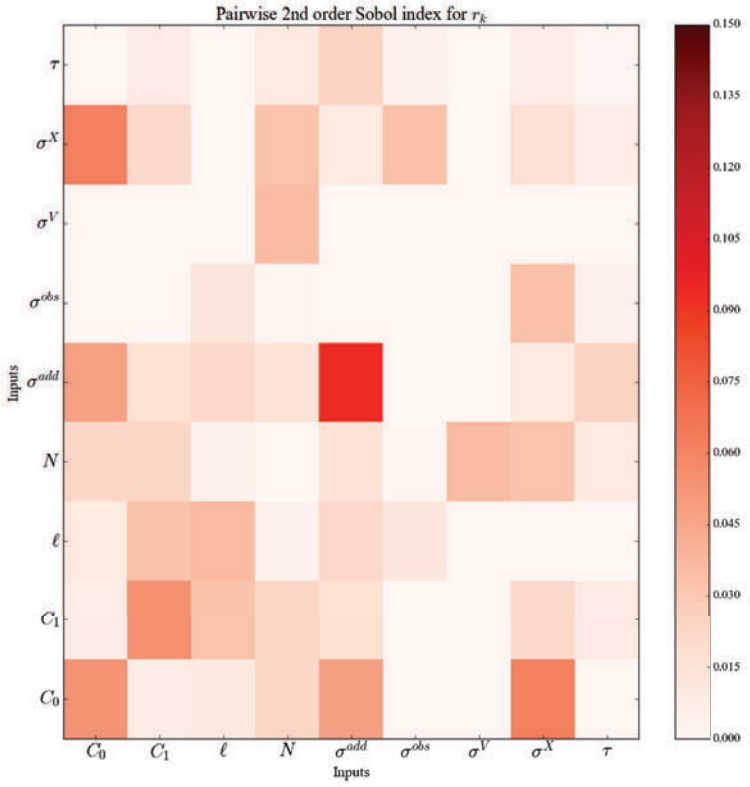


Figure 6.17 – Tile of 2nd order Sobol indices for the TKE RMSE. Pairwise interactions are as strong as direct influence (on the diagonal).

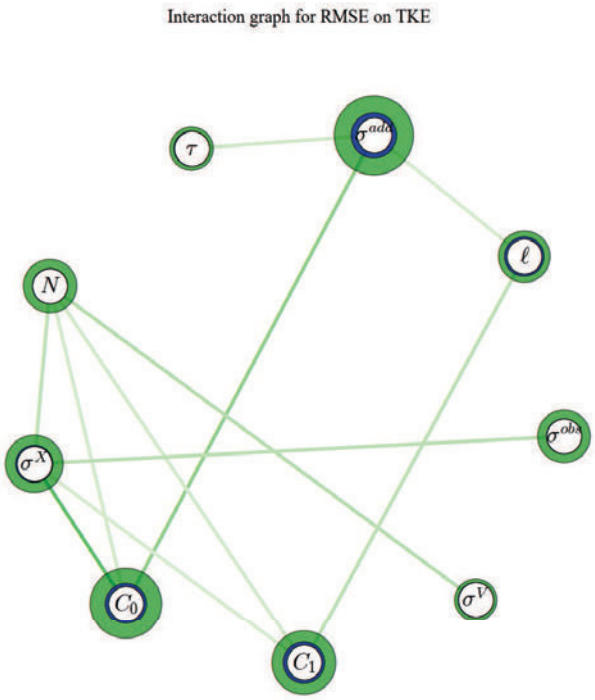


Figure 6.18 – Graph of interaction for the RMSE on TKE. Simple Sobol indices are notably small, to the advantage of total Sobol indices which are significant for all inputs. Pairwise interactions are strong and rather numerous. But higher interactions might be at stake to fill the gap between simple and total Sobol indices.

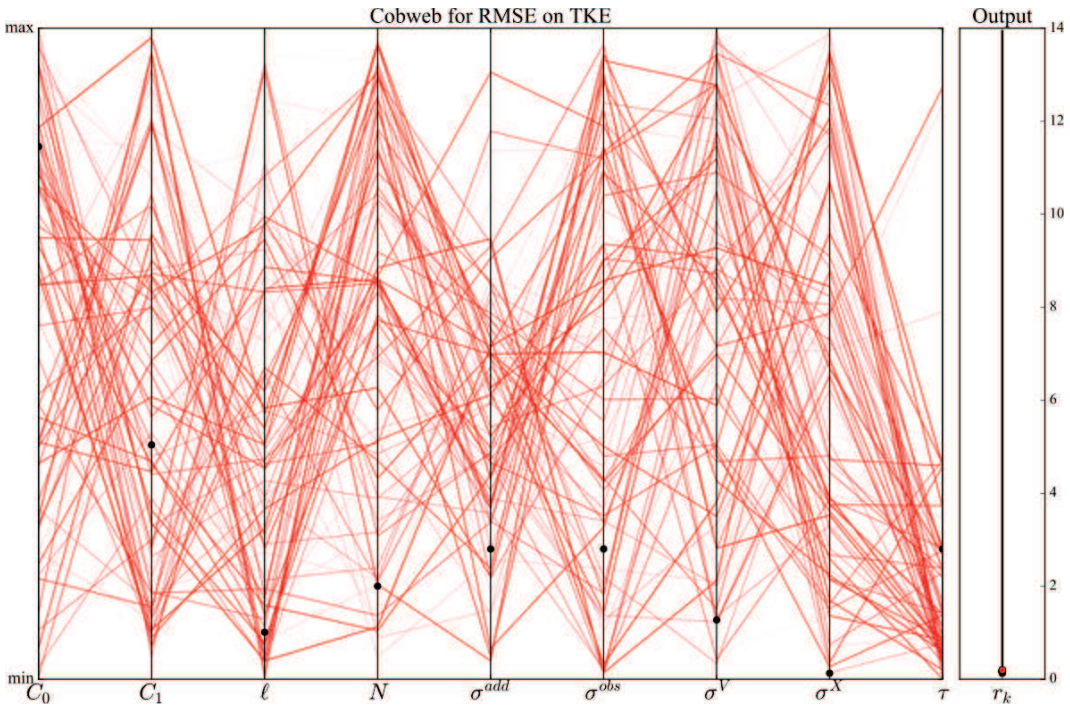


Figure 6.19 – Cobweb plot for low TKE RMSE. It is associated to nothing really clear, excepted opposite C_0 and C_1, small τ and rather small σ^{add}.

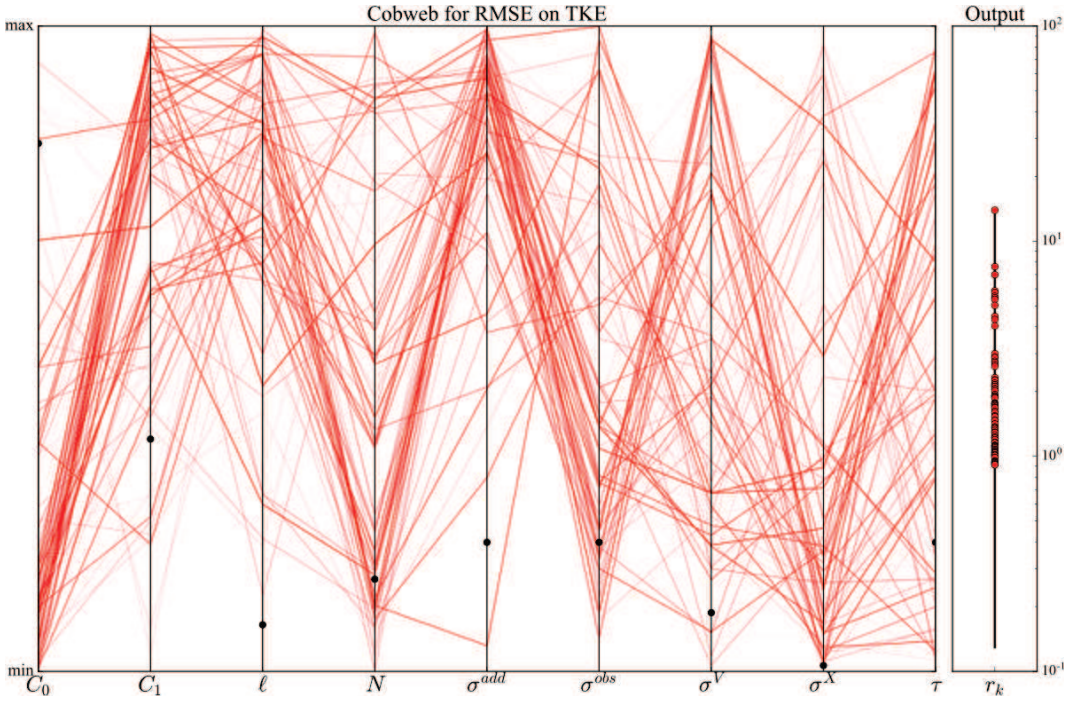


Figure 6.20 – Cobweb plot for high TKE RMSE. It is associated to small C_0, high C_1, high ℓ and high σ^{add}.

6.2.4 Influence on wind RMSE

The RMSE on wind (r_V) is the L^2 error of the estimated wind compared to the reference wind. The lower r_V, the better the wind retrieval.

According to the figure 6.21, the most influential inputs on the wind retrieval error are σ^{add}, N, C_0 and σ^{obs}. The influence of N though the Theorem 5.1 is thus a mechanism of importance. A notable feature is that the simple and total Sobol indices are ranked in the same order. It means the strongest inputs alone are also the strongest through interactions. Conversely to the wind spectrum slope, the interactions are significant here, as shown by the pie chart 6.22. The 2$^{\text{nd}}$ order tile 6.23 points out that the strongest interactions are essentially between inputs also strong alone (σ^{obs} and σ^{add} ; N and σ^{add}, but not N and σ^{obs}). On the graph of interactions 6.24, C_0 does not display any pairwise interaction but does display interactions. It proves that the interactions in which C_0 is involved are higher order than 2. The cobweb plots 6.26 and 6.25 point out quite clear schemes: r_V is high when N is low, σ^{add} is high, C_0 is low and C_1 is high. Conversely, r_V is low when N is high, σ^{add} is low and σ^{obs} is low. These trends confirm the influence seen in the Sobol indices.

The output r_V is mainly influenced by few inputs with a strong first order Sobol index (σ^{add}, N, C_0 and σ^{obs}). These inputs also responsible for the most important pairwise interactions. The sensitivity analysis reveals only few inputs are to consider to master the wind RMSE. As consequence, it is an output rather easy to control.

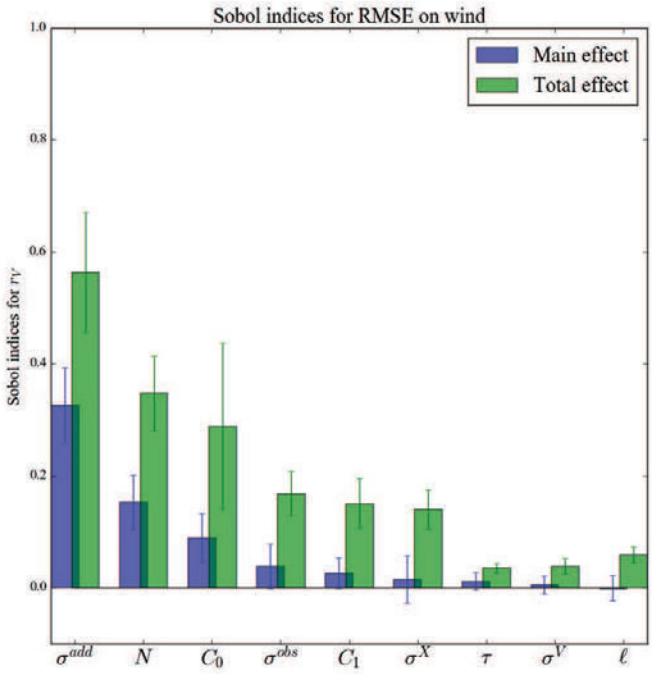


Figure 6.21 – First order and total Sobol indices for the RMSE on wind with 95% confident interval.

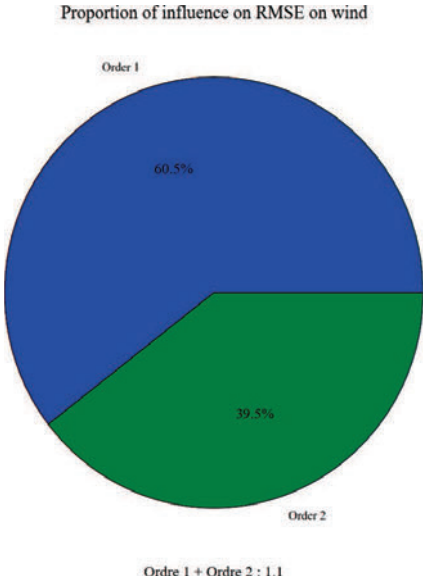


Figure 6.22 – Proportion of 2nd and 1st order for the wind RMSE. The major part of the influence is direct. Although, pairwise interaction account for 39% of the influence. No trace of higher order interaction is visible here.

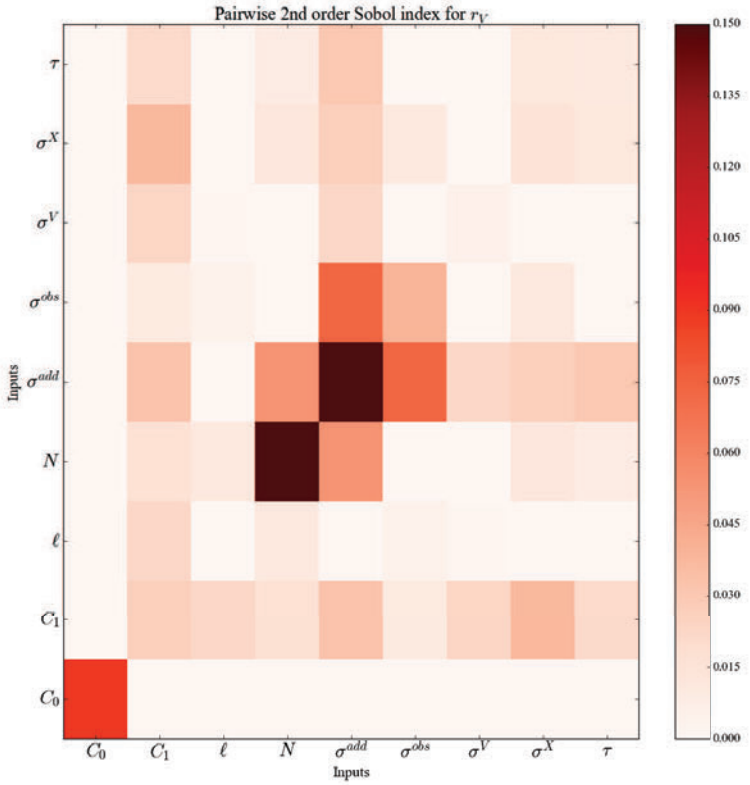


Figure 6.23 – Tile of 2nd order Sobol indices for the wind RMSE. Some pairwise interactions have a strong influence on the output (σ^{obs} and σ^{add} ; N and σ^{add}). The inputs involved in these interactions have also a strong influence alone.

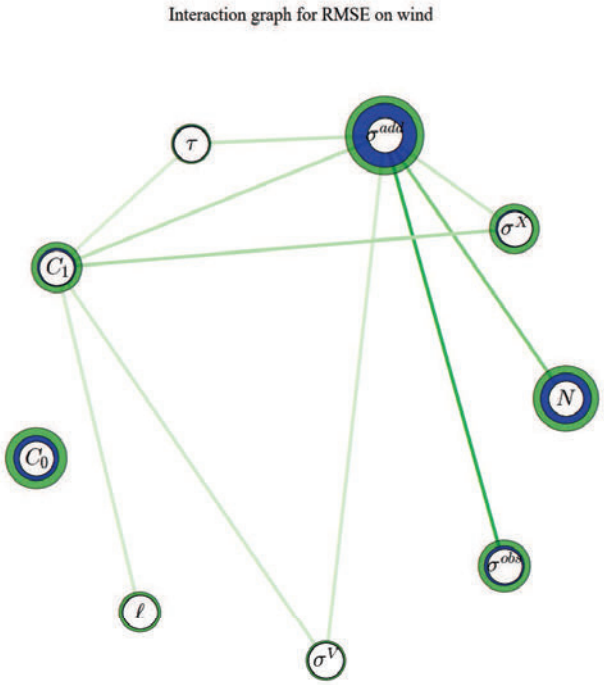


Figure 6.24 – Graph of interaction for the wind RMSE. The strong inputs alone (σ^{add}, σ^{obs}, N) also interact among them. C_0 interacts (large outer ring) but not pairwise, thus it interacts at higher order.

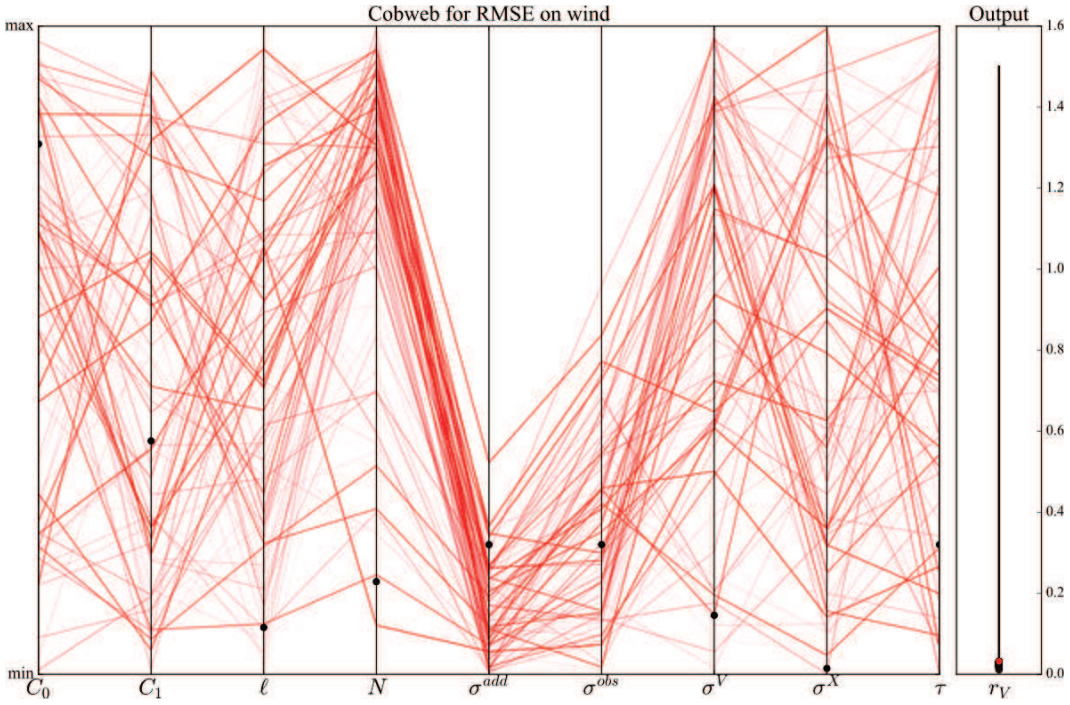


Figure 6.25 – Cobweb plot for low wind RMSE. It is associated to high N, low σ^{add} and low σ^{obs}.

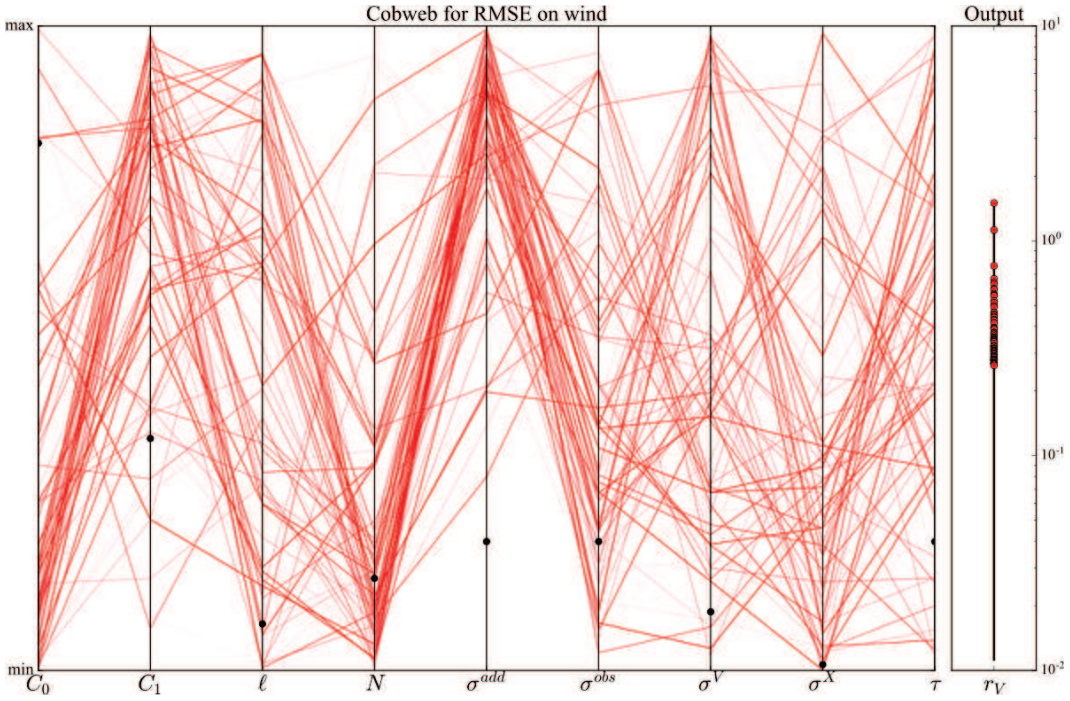


Figure 6.26 – Cobweb plot for high wind RMSE. It is associated to low N, high σ^{add}, low C_0 and high C_1.

6.2.5 Influence on computing time

The interpretation of all the figures related to this output is clear: only the number of particles leads the computing time. The influence of N overwhelms any other. It has a very large first order Sobol index (figure 6.27) which dominates all the rest (see pie chart 6.28, 2$^{\text{nd}}$ order tile 6.29 or interaction graph 6.30). It may have been possible that some of the others parameters have an influence. For example, σ^{obs} controls the number of rejected particles. The more rejected particles, the more resampling must be done. Another example, σ^V helps to avoid gathering of particles. A loop in the conditioning step is repeated while particles are not spread enough. Although resampling and repeating a loop requires more calculus, the sensitivity analysis shows they are negligible compared to the cost implied by the number of particles.

The execution time is a very handy output, which can be controlled easily. Unfortunately, it tells nothing about the quality of the reconstruction. It is only a trade-off variable to plan numerical experiments.

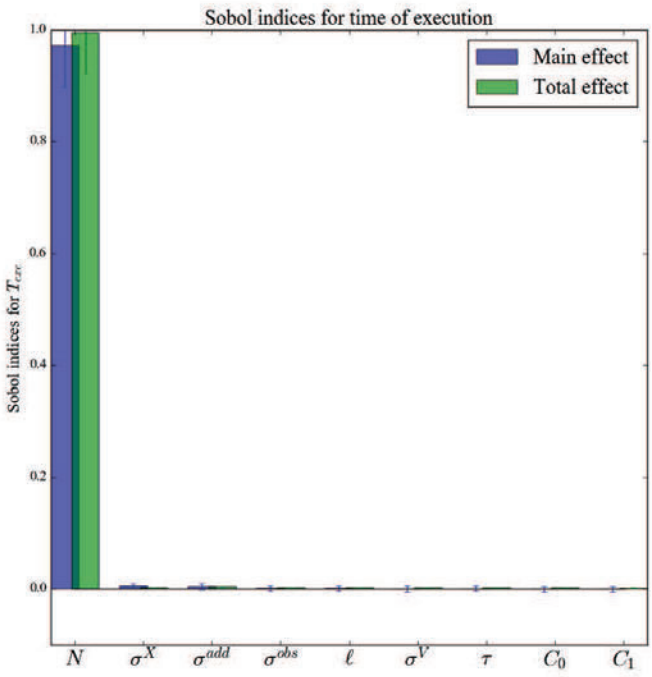


Figure 6.27 – First order and total Sobol indices for the execution time with 95% confident interval.

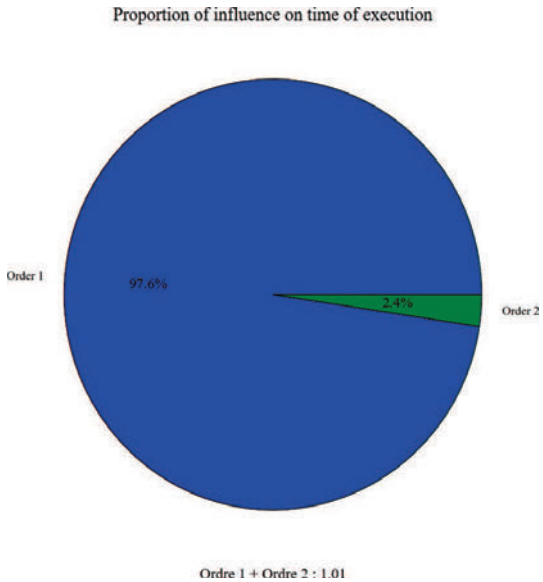


Figure 6.28 – Proportion of 2nd and 1st order influence for the execution time. Direct effect overwhelms interactions.

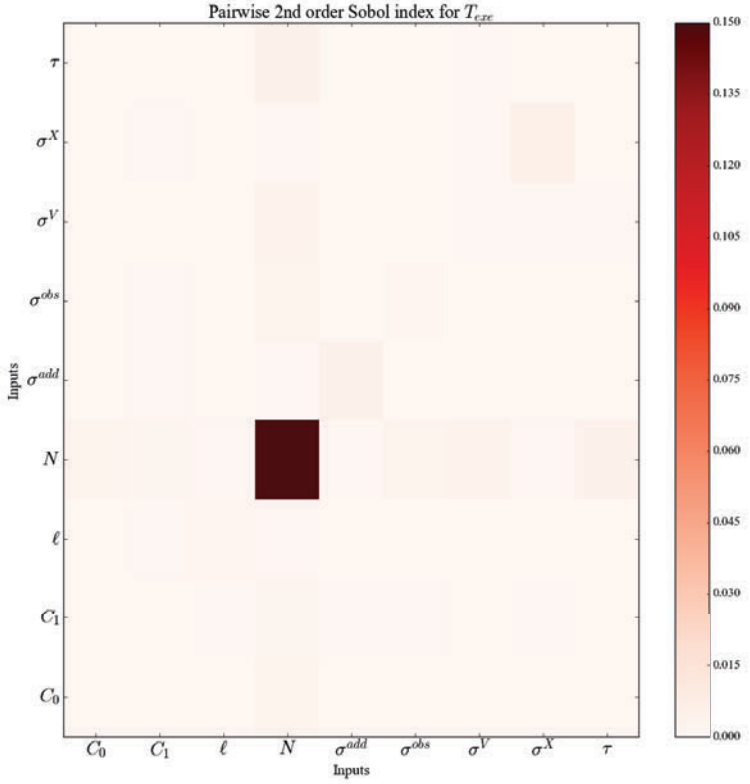


Figure 6.29 – Tile of 2nd order Sobol indices for the execution time. No pairwise interactions are significant. Only N has an influence on this output.

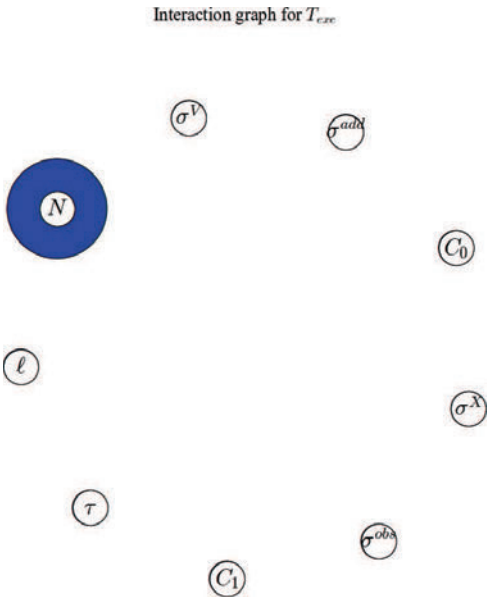


Figure 6.30 – Graph of interaction for the execution time. Only N has an influence.

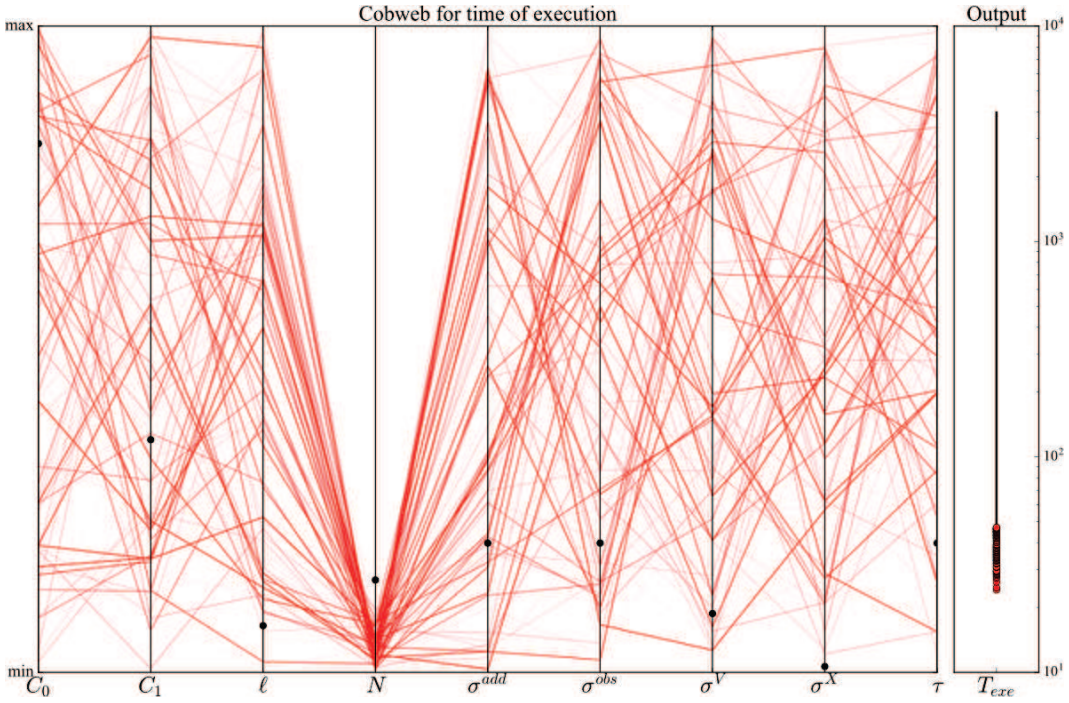


Figure 6.31 – Cobweb plot for low execution time. It is associated to low N.

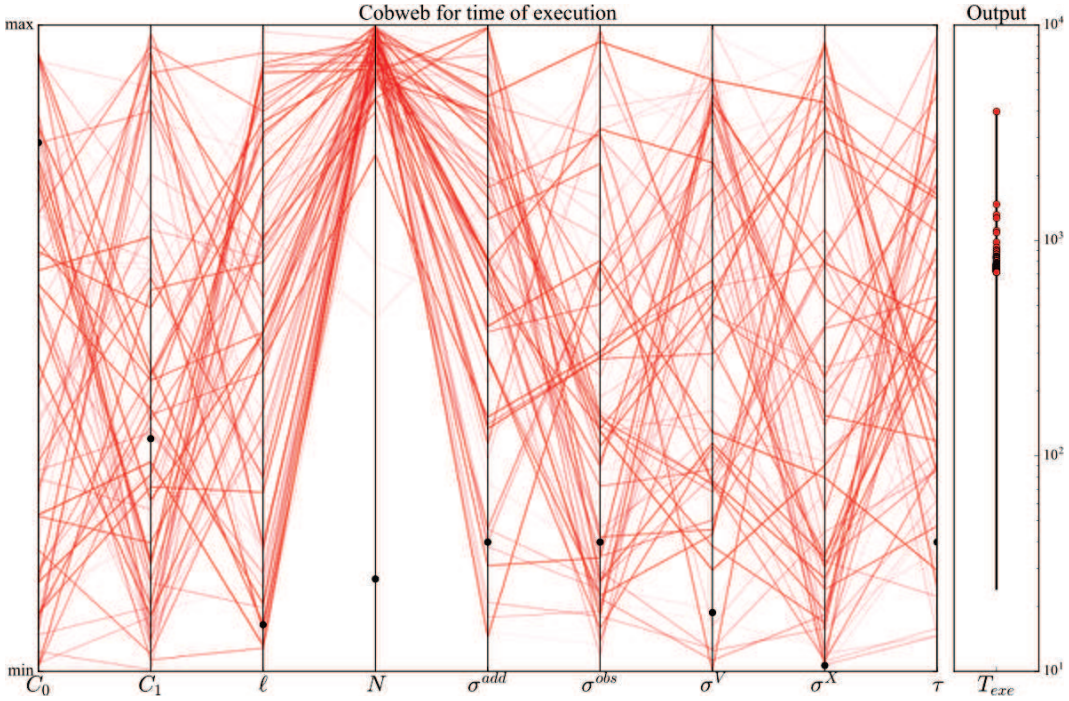


Figure 6.32 – Cobweb plot for high execution time. It is associated to high N.

6.2.6 Summary

From the separated analyses of the results for the five outputs, it is possible to highlight the most influential parameters and the share of interactions for each output. The system presented in figure 6.1, quite heavy to run, is emulated by a surrogate model (Gaussian process kriging). Sobol indices have been estimated with the estimators in table 4.1. An output can be used as a tuning output if it is poorly influenced by interactions.

The sensitivity analysis of N_{G0} shows this output is influenced mostly by pairwise interactions between inputs. Despite few inputs have a remarkable influence alone (σ^{obs} especially) it is not enough to master this output. The number of null potential is a hardly controllable output. It will not be used as a tuning output.

The wind spectrum is only affected by σ^{add}, σ^{obs}, C_0 and C_1. Their influence is mostly direct, interactions are very weak. Hence, the wind spectrum is easily controllable. It can be used to tune the inputs.

The RMSE on TKE has very complex variations. It is poorly influenced by inputs alone, but mostly by inputs in interactions, including a large part of high order interactions. As a consequence, this output is very difficult to master. It cannot be trusted to tune the inputs.

The RMSE on wind is mainly influenced by few inputs with a strong first Sobol index (σ^{add}, N, C_0 and σ^{obs}). These inputs are also responsible for the most important pairwise interactions. The sensitivity analysis reveals only few inputs are to consider to master the wind RMSE. As consequence, it is an output rather easy to control.

The execution time is very handy output, which can be controlled easily. Only the number of particles N has a influence on it. Unfortunately, it tells nothing about the quality of the reconstruction. It is only a bargain variable to plan numerical experiments.

6.3 All outputs analysis

Sensitivity analyses have been first build to assess the influence of several inputs on one output. On this system, there are 5 outputs : the number of null potential N_{G0}, the slope of wind PSD b, the RMSE on TKE r_k, the RMSE on wind r_V and the execution time T_{exe}. Hence the same sensitivity analysis has been repeated on each output. This was a choice of simplicity. Indeed, the paper of Gamboa et al. (2014) defines two indices suitable to vector output (even to functional ouput) which coincide with Sobol indices for a scalar output. Although a simpler computation, we will keep this result in mind for the comment of the results.

Having several outputs allows one to reverse the problem: how influential is a given input on all outputs? When one is moving a given input in order to get better results for a given output, what are the side-effects of this move? A first source of side-effect is the involvement of the given input in influential interactions. A second source of side-effect is the influence of the given input on other outputs. If the move improves the first output, it might degrade another one. This question is interesting because the 5 outputs are 5 scores on different aspects of the reconstruction.

6.3.1 Global analysis

In this section, we adopt a more global point of view: all outputs are taken together. The five outputs are ordered as follow:

$$Y = \begin{pmatrix} N_{G0} \\ b \\ r_k \\ r_V \\ T_{exe} \end{pmatrix}$$

The Hoeffding decomposition is valid for this vector output too.

$$Y = \sum_{\mathbf{u} \in I} f_{\mathbf{u}}(X_{\mathbf{u}})$$

As long a inputs are independent, this decomposition holds for variances. If we denote $\Sigma = V(Y)$ and $C_{\mathbf{u}} = V(f_{\mathbf{u}}(X_{\mathbf{u}}))$, we have

$$\Sigma = \sum_{\mathbf{u} \in I} C_{\mathbf{u}}$$

Then the first estimator defined in Gamboa et al. (2014) is

$$S_{\mathbf{u}}(Y) = \frac{\mathrm{tr}(C_{\mathbf{u}})}{\mathrm{tr}(\Sigma)} \tag{6.1}$$

where $\mathrm{tr}(\cdot)$ is the trace operator. In the paper, they also suggest a method of estimation that has not been implemented in this work. Nevertheless, it is possible to link the Sobol indices obtained independently to the global sensitivity index defined at equation (6.1). Without introducing new notations, we write

$$\mathrm{tr}(\Sigma) = V(N_{G0}) + V(b) + V(r_k) + V(r_V) + V(T_{exe})$$

$$\mathrm{tr}(C_{\mathbf{u}}) = V(\mathbb{E}\left[N_{G0}|X_{\mathbf{u}}\right]) + V(\mathbb{E}\left[b|X_{\mathbf{u}}\right]) + V(\mathbb{E}\left[r_k|X_{\mathbf{u}}\right]) + V(\mathbb{E}\left[r_V|X_{\mathbf{u}}\right]) + V(\mathbb{E}\left[T_{exe}|X_{\mathbf{u}}\right])$$

For each scalar output (for instance r_V) the Sobol index is the ratio $\frac{V(\mathbb{E}[r_V|X_{\mathbf{u}}])}{V(r_V)}$, thus we have the following relationship:

$$S_{\mathbf{u}}(Y) = \frac{V(N_{G0})}{\mathrm{tr}(\Sigma)} S_{\mathbf{u}}(N_{G0}) + \frac{V(b)}{\mathrm{tr}(\Sigma)} S_{\mathbf{u}}(b) + \frac{V(r_k)}{\mathrm{tr}(\Sigma)} S_{\mathbf{u}}(r_k) + \frac{V(r_V)}{\mathrm{tr}(\Sigma)} S_{\mathbf{u}}(r_V) + \frac{V(T_{exe})}{\mathrm{tr}(\Sigma)} S_{\mathbf{u}}(T_{exe}) \tag{6.2}$$

That is to say, with a ponderation by the variance of the output, the Sobol indices obtained by five separated sensitivity analyses yield to the index for vector output in Gamboa et al. (2014). This index is shown to have very nice properties:

- It sums up to 1: $\sum_{\mathbf{u} \in I} S_{\mathbf{u}}(Y) = 1$
- Invariant by isometry: $\forall O$ such that $O^T O = I_s$, $S_{\mathbf{u}}(OY) = S_{\mathbf{u}}(Y)$

Output	Min	Max	Unit
N_{G0}	0	7524	none
b	-4.716	0.772	none
r_V	0.011	1.5	$\mathrm{m{\cdot}s^{-1}}$
r_k	0.129	13.93	$\mathrm{m^2{\cdot}s^{-2}}$
T_{exe}	24.17	3968	s

Table 6.4 – Range of variations of the outputs (from original response surface).

- Invariant by scaling: $\forall \lambda \neq 0$, $S_{\mathbf{u}}(\lambda Y) = S_{\mathbf{u}}(Y)$

The drawback of this weighting is that the ranges of variation of outputs are not taken into accounts. Outputs taking large values will have a larger weight than the others. It is the case here (see table 6.4). Thus, N_{G0}'s Sobol indices accounts for 80% of the vector output Sobol indices, T_{exe} accounts for the remaining 20% and the others for nearly 0%, as shown in figure 6.33. The inputs influencing these output will be overrepresented in the final interpretation.

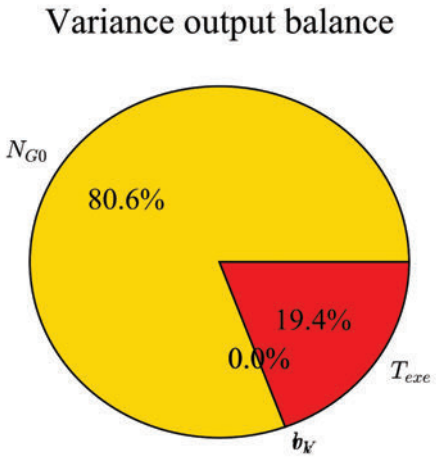


Figure 6.33 – Proportion of variance in the vector output.

To avoid this issue, in this section, all outputs are taken together with equal weight. The Sobol indices from the five sensitivity analysis have been summed. Unlike equation (6.2), the Sobol indices presented now are obtained by

$$S_{\mathbf{u}}(Y) = S_{\mathbf{u}}(N_{G0}) + S_{\mathbf{u}}(b) + S_{\mathbf{u}}(r_k) + S_{\mathbf{u}}(r_V) + S_{\mathbf{u}}(T_{exe})$$

The resulting scores does not sums up to 1 but give an overview of the influence of a given input or a given pair of inputs. They are displayed in the same way as previously.

In figure 6.34 is the pie chart of first and second order influence proportion. It tells the influence is mostly direct, with 68% of first order. Still, second order Sobol indices account for nearly 32%, which is not negligible. This figure does not tell anything about higher order

interaction because it only apportions first and second order contributions. In the figure 6.16 for example, the same pie chart is relative to a single sensitivity analysis. The higher order contribution can be estimated by the leftover after first and second orders have been removed, because the sum of all contributions is 1. This is no longer the case for the pie chart 6.34 because it sums up five independent sensitivity analysis.

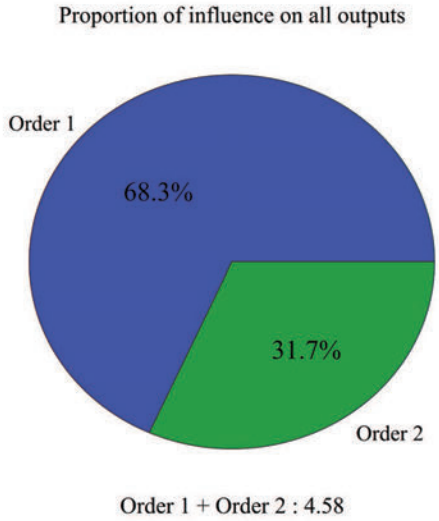


Figure 6.34 – Proportion of first and second order for all outputs.

In figure 6.35 is the tile of second order Sobol indices. Each cell is dedicated to the Sobol index of the interaction between the two inputs in abscissa and ordinate. This interaction is quantified by the sum of the second order Sobol indices for the five outputs. The more coloured is a cell in the tile, the stronger is the overall interaction between the two inputs. The tile is thus symmetric. On the diagonal, the shade is proportional to the sum of first order (main effect) Sobol index for the five outputs. This figure highlights the most influential inputs (deepest coloured cells on the diagonal): C_0, C_1, N, σ^{add}, σ^{obs}. It also gives the most important interactions (deepest coloured cells out of the diagonal): C_0 with σ^X, C_1 with σ^X, C_1 with σ^{add}, σ^{add} with σ^{obs} for the 4 major. A notable thing is that the most influential inputs alone are also doing the most influential interactions. Globally, controlling these inputs is enough to master the whole system. From this result, one could imagine the inputs ℓ, σ^V and τ can be fixed to a constant value without disturbing the system. It reduces to number of inputs and makes the system simpler.

In figure 6.36 is the graph of interaction obtained with the sum of the Sobol indices from all sensitivity analysis. We retrieve the same conclusion as with the tile 6.35 : the most influential outputs are N, σ^{add}, σ^{obs}, C_0 and C_1. The interactions between these inputs are also the strongest ones. Beside this top 5, the input σ^X interacts strongly with C_0 especially, but also with C_1. The input ℓ is also visible because of many interactions, but these interactions are weaker in the tile 6.35 than they appear on the graph 6.36. The others (σ^V, τ) are not influential alone, nor though interactions. As a consequence, they can be removed safely from the following studies on the system. If required, the next inputs to remove are ℓ first, then σ^X.

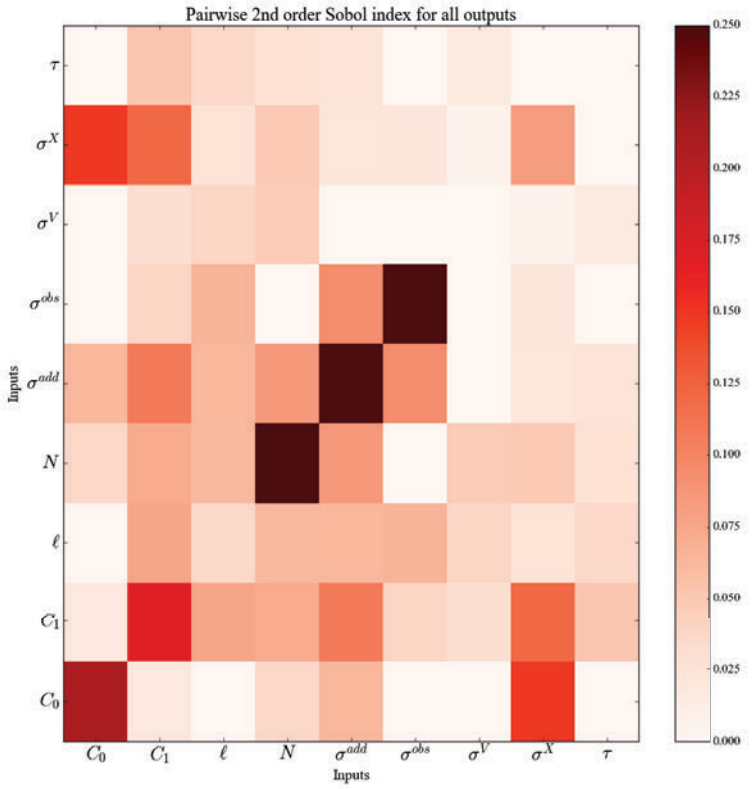


Figure 6.35 – Tile of 2nd order Sobol indices for all outputs.

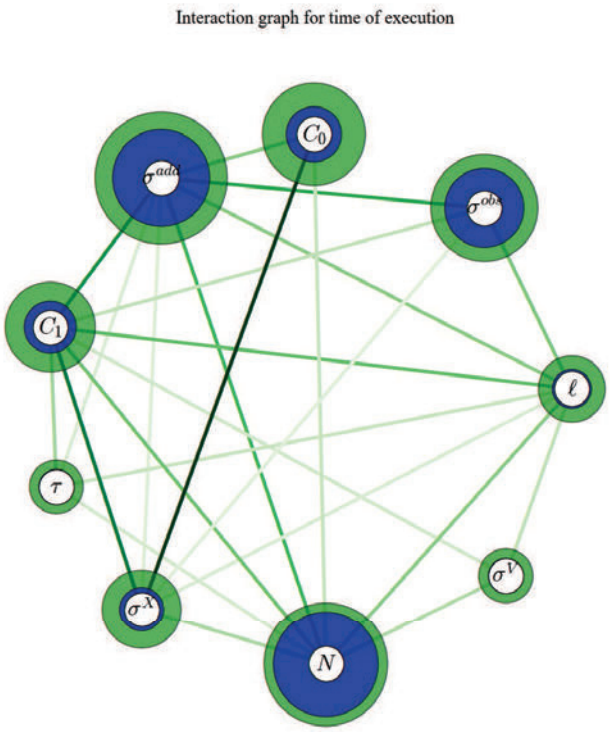


Figure 6.36 – Graph of interaction for all outputs.

6.3.2 Crossing input and output

To keep the global point of view adopted in this section, the results from the previous section have been collected in a synthetic way which crosses input and outputs. By doing so, one can see what side-effects may have the change of one input.

In figure 6.37, outputs are in abscissa and inputs are in ordinate. The shade of color is proportional to the corresponding simple first order Sobol index. We retrieve clearly the same top 3 inputs : N, σ^{add} and σ^{obs}. Because of the color scale imposed by the very high Sobol index of N on T_{exe}, the two next inputs revealed by the previous figures, C_0 and C_1, appear lightly. Moreover, C_0 and C_1 have a larger contribution in interaction, which is not displayed in this figure. Among the top 3 inputs, the figure 6.37 reveals they are not ranked equally for the five outputs. σ^{obs} has a strong influence on N_{G0} and b, but not on r_V, r_k nor T_{exe}. σ^{add} has a strong influence on r_k, r_V and b, but not on N_{G0} nor T_{exe}. N has a strong influence on T_{exe} and r_V, but not on N_{G0}, b nor r_k. Hence, the side-effect of changing several outputs by moving a single input is clearly stated here. For example, a move of N will not have any side-effect on N_{G0}, b nor r_k. This is an important information in order to make a tuning of inputs.

For the side-effect due to interaction, the move of an input does not have the expected

effect because of interactions involved. It is thus interesting to look at the strength of interactions. The figure 6.38 shows the interaction part deduced from the Sobol indices (the difference between the total and simple Sobol indices). It appears that the lines T_{exe} and b are almost empty. We have seen in the section "Output by output" that T_{exe} and b are weakly influenced by interactions. This is confirmed by this observation. It tells the moving of any variable will not imply side-effect due to interactions on T_{exe} and b. It is also visible that the columns σ^V and τ look empty as well (even ℓ to a lesser extent). It tells a variation of σ^V or τ will not have side-effects due to interactions. We have seen in the examination of figure 6.37 that σ^V or τ have no direct effect neither. Hence, they can be removed of the study without any bad return.

This subsection pointed out the inputs variables do not account equally in the system. Three outputs appear to be the main ones: N, σ^{add} and σ^{obs}. Beside them, C_0, C_1 also have a large part of influence but less direct effects. They are followed by two inputs which are not as influential but notable because of the interactions they are involved in: ℓ and σ^X. Eventually, the two last inputs have been shown to be globally not influential: σ^V and τ. To sustain this, the ranking of inputs according to their average Sobol indices[1] is displayed in table 6.5. The order slightly changes between simple and total Sobol indices, but only inside one group.

Overall influence

Alone	N	σ^{add}	σ^{obs}	C_0	C_1	σ^X	ℓ	τ	σ^V
	0.223	0.188	0.115	0.042	0.034	0.016	0.007	-0.002	-0.004
Total	σ^{add}	N	σ^{obs}	C_0	C_1	σ^X	ℓ	σ^V	τ
	0.373	0.322	0.235	0.215	0.158	0.115	0.074	0.04	0.038
	Group 1			Group 2		Group 3		Group 4	

Table 6.5 – Average first order Sobol indices for each input.

[1]It is the average of the Sobol indices provided by the five sensitivity analysis

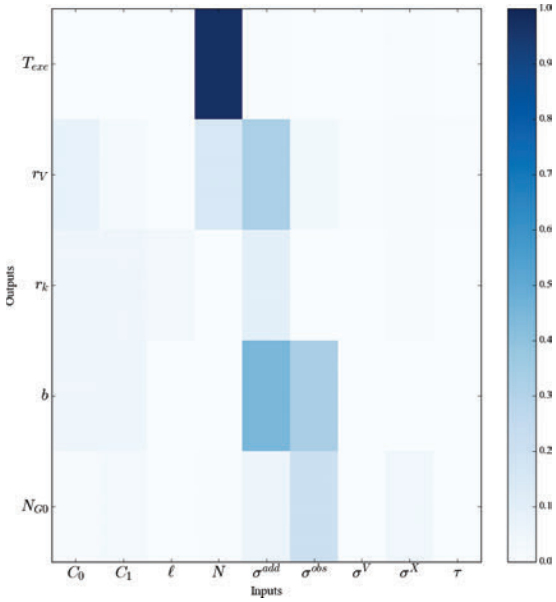


Figure 6.37 – 2D color plot of simple Sobol indices. The influence *alone* of one input on one output is proportional to the color strength.

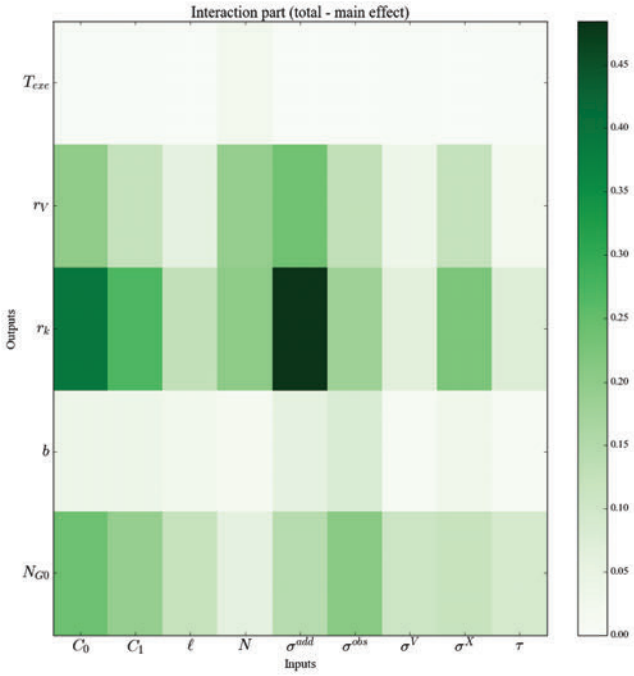


Figure 6.38 – 2D color plot of interaction part (difference between total and simple Sobol indices). The influence on one output of *interactions* in which is involved an input is proportional to the color strength.

6.3.3 Sobol indices by input

For more detailed view of potential side-effects, bar chart of Sobol indices by input are displayed. Both interaction and main effect are represented with their 95% confident interval. It avoids issues of color scale like in the tile 6.37. They are the same Sobol indices as in the section "Output by output analysis". The average value given on the figures are the average over the five outputs, as in table 6.5. The inputs are gathered by group of influence, as they appear after the examination of the figures earlier in this section. The first group gathers the top 3 inputs: N, σ^{add} and σ^{obs}. They are the predominant inputs, especially by their direct effect. The second group gathers C_0 and C_1, which have a weaker direct effect than the 3 previous but are still remarkably influential. The third group gathers ℓ and σ^X, which do not account as much as the 5 previous but still have a significant weight though interactions. The fourth and last group gathers σ^V and τ, which do not appear significant in the previous figures.

6.3.3.1 Most influential inputs: N, σ^{add} and σ^{obs}

In figure 6.39, the Sobol indices of σ^{add} show this input has a strong influence on b (wind spectrum slope) and r_V (wind RMSE), then on r_k (TKE RMSE) and N_{G0} (number of null potential) to a lesser extent. In practice, σ^{add} is not known. Only a guess of it is known: σ^{obs}. Although σ^{add} cannot be tuned, σ^{obs} can be.

The figure 6.40 shows σ^{obs} has a strong influence on b (wind spectrum slope) and N_{G0} (number of null potential), then on r_V (wind RMSE) to a lesser extent. Thus, a change of σ^{add} in order to correct the wind spectrum, for example, will have a strong side-effect on the number of null potential and a smaller side-effect on the wind RMSE.

It is interesting to notice they have a different influence while they are supposed to represent the same thing. Both influence strongly the wind spectrum. But the next output to be influenced is N_{G0} for σ^{obs}, while it is r_V for σ^{add}. An explanation for this observation is the very different use of both parameters in the system: σ^{add} is only used to add noise to the reference data, while σ^{obs} is used to remove this noise by filtering. It explains why σ^{obs} has a more direct influence on N_{G0}, which is a score specific to the selection step. On the other hand, σ^{add} directly increases wind RMSE, while the effect of σ^{obs} is diluted by the other parameters of the filtering process.

At the examination of figure 6.41, one can see the influence of N is mostly due to its overwhelming influence on the time of execution. Nevertheless, it strongly influences the RMSE on the wind. This influence is sustained by theorem 5.1. The setting of N is thus a trade-off between allocated computing time and desired error on wind retrieval.

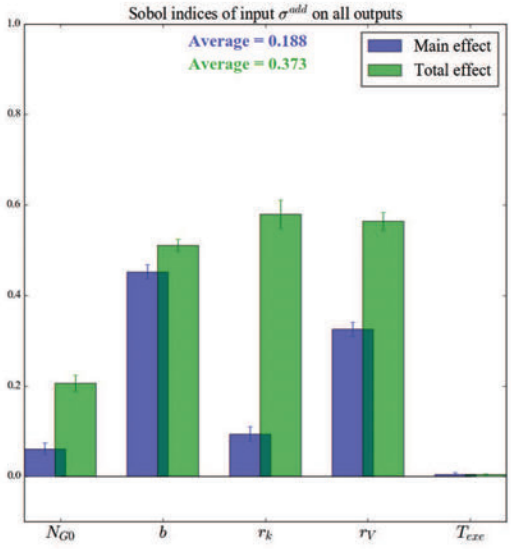


Figure 6.39 – Sobol indices : main effect (blue) and total effect (green) of true measurement noise.

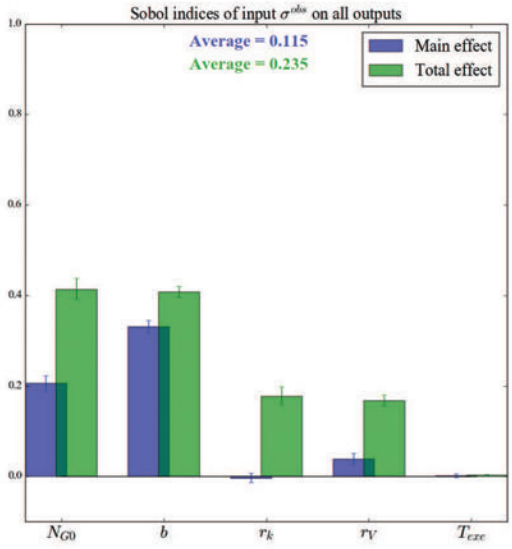


Figure 6.40 – Sobol indices : main effect (blue) and total effect (green) of measurement noise given to the filter.

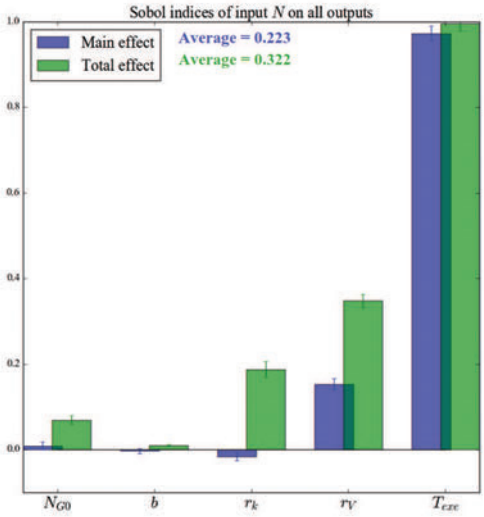


Figure 6.41 – Sobol indices : main effect (blue) and total effect (green) of number of particles.

6.3.3.2 Less influential inputs: C_0 and C_1

The parameters C_0 and C_1 are less influential than the others in the first group. Especially, the part of interaction in their influence is larger than for N, σ^{add} and σ^{obs}, as one can see in figures 6.42 and 6.43. They have also in common to appear only in the Lagrangian model, which is in use only is the mutation step. Their influence highlights the benefit (or the curse) to have a well chosen model. They have a nearly equal influence on the wind spectrum. For C_0 the first output is r_V, while for C_1, the first output is b. Turbulence modelling gives more insight about what should be their value. The sensitivity analysis points out that an error on the value of C_0 will have a side-effect on the wind RMSE, the wind spectrum and the number of null potential, in addition to other side-effects due to interactions, especially on the TKE RMSE. An error on C_1 will not have a direct side-effect on the wind RMSE, but beside this, the side-effects are the same as for C_0.

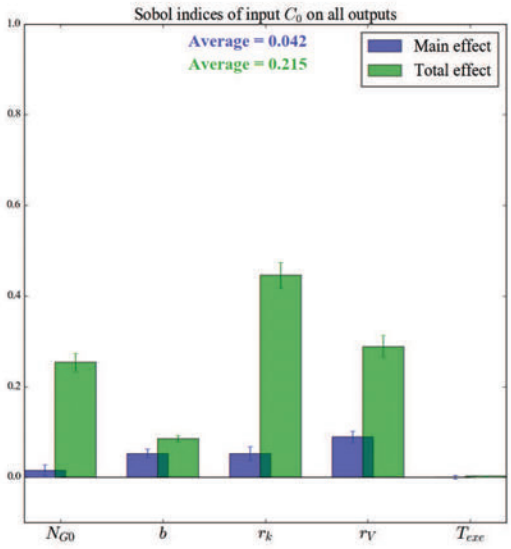


Figure 6.42 – Sobol indices : main effect (blue) and total effect (green) of Kolmogorov's constant.

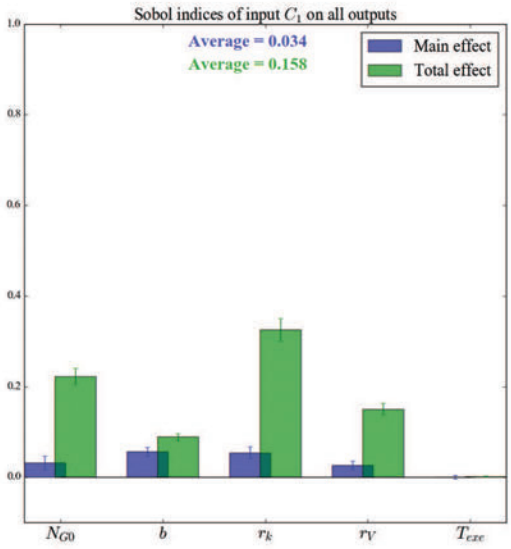


Figure 6.43 – Sobol indices : main effect (blue) and total effect (green) of fluctuation coefficient.

6.3.3.3 Involved in strong interactions: ℓ and σ^X

The two inputs ℓ and σ^X have been put in the same group because they have a similar influence. Like C_0 and C_1, the interactions are responsible for a large share of their influence. Actually, it has been seen that C_0 and σ^X have the strongest pairwise interaction overall (figure 6.35 and 6.36). But conversely to C_0 and C_1, modelling cannot help to set this parameters. They have been introduced to make the algorithm functional and there is no study of their effect. Hopefully, the sensitivity analysis show their influence is not massive. Moreover, the deep comment made here help to understand better their influence.

As one can see in the figure 6.44, ℓ influences only r_k directly. Hence, a bad setting of ℓ will have a small direct effect only on r_k. But it can also have many side-effects on N_{G0}, r_k and r_V because of interactions. It has been shown on the graph 6.36 that ℓ has pairwise interactions with σ^{add}, σ^{obs}, N and C_1. The interaction between ℓ and σ^{add} or σ^{obs} or N is probably due to the TKE estimation. The formula for the estimation of TKE involves explicitly N and ℓ, and it is an estimation of the variance of the wind, which is driven by σ^{add} and σ^{obs}. The influence between ℓ and C_1 is due to the local average in the fluctuation term.

In figure 6.45, one can see the small direct effects of σ^X affect N_{G0}, r_k and r_V. They are also the outputs affected by the weight of σ^X in interaction. The pairwise interaction of σ^X and C_0 is necessary in the mutation step: C_0 is in the dispersion of the speed equation, σ^X is in the dispersion term in the position equation. By integration, σ^X and C_0 interact. It has for consequence that N_{G0} is strongly affected by this interaction (the more particles are disturbed at the mutation, the more they are rejected at the selection) while r_V is not (because the selection step filters out the defaults of the mutation).

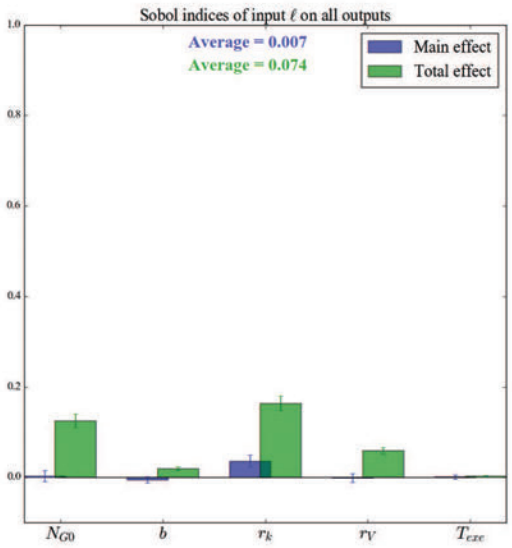


Figure 6.44 – Sobol indices : main effect (blue) and total effect (green) of interaction length.

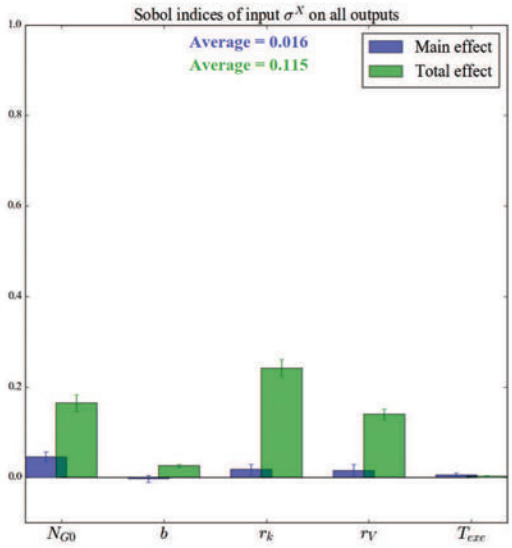


Figure 6.45 – Sobol indices : main effect (blue) and total effect (green) of standard deviation of discretisation error.

6.3.3.4 Not influential: σ^V and τ

The last group is composed of σ^V and τ which have shown no significant influence on any of the five outputs. The figures 6.46 and 6.47 confirm it. It is valuable result to reduce the dimension of the system. It also raises questions about their role in the system.

τ is only involved in the calculus of the score r_k. It has no significant influence on r_k, which tells the way the score is calculated cancel the effect of τ. It has no significant influence, but still has a non-zero Sobol index for other outputs (especially r_V and N_{G0}). This false alarm is attributed to the stochastic nature of the code. Although the range of N and the scores has been chosen in order to kill the variability of the outputs, the remaining variations might be visible for poorly influential parameters.

σ^V appears only at the conditioning step. Given the usual amount of particles affected by the conditioning step (particles out or redistributed in figure 5.20: an average of 1.9% of the total number of particles), this non-influence is not surprising. Although, during the development phase of the reconstruction algorithm, σ^V was added to avoid crashes caused by concentrations of particles at the same location. The sensitivity analysis shows that beside avoiding crashes, σ^V has no other effect, which is a good news.

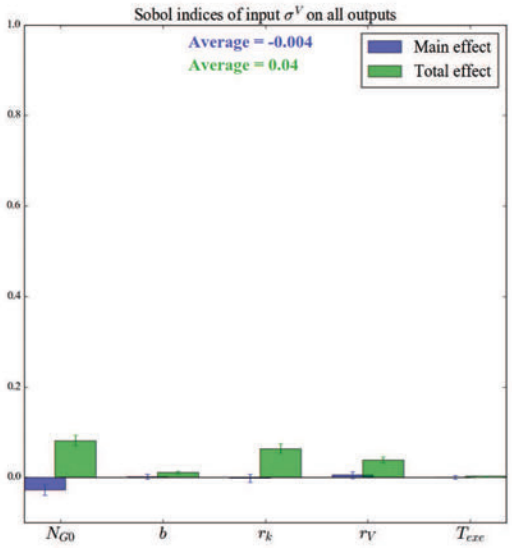


Figure 6.46 – Sobol indices : main effect (blue) and total effect (green) of default standard deviation on wind.

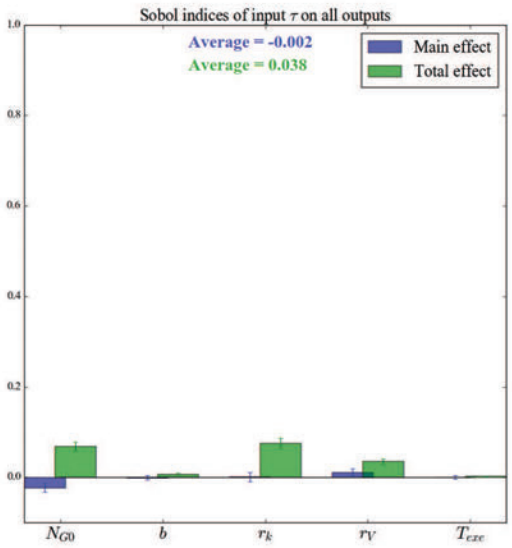


Figure 6.47 – Sobol indices : main effect (blue) and total effect (green) of integration time.

6.3.4 Summary

The influence of inputs on the overall system has been assessed by summing the Sobol indices of the five independent sensitivity analysis. It results that inputs can be gathered in four groups:

1. The most influential variable: σ^{add}, N and σ^{obs}.

   They all have a strong direct influence. σ^{add} has a strong influence on b (wind spectrum slope) and r_V (wind RMSE). N has a strong influence on T_{exe} (time of execution) and r_V (wind RMSE). σ^{obs} has a strong influence on b (wind spectrum slope) and N_{G0} (number of null potential), then on r_V (wind RMSE) to a lesser extent.

2. Less influential variable: C_0 and C_1.

   Turbulence modelling gives more insight about what should be their value. An error on C_1 impacts b (wind spectrum) and N_{G0} (number of null potential), in addition to other side-effects due to interactions, especially on r_k (TKE RMSE). An error on C_0 impacts the same outputs as C_1 plus the wind RMSE r_V.

3. Involved in strong interactions: ℓ and σ^X.

   Errors imply side-effect though interactions, mostly on r_k and N_{G0}.

4. Not influential: σ^V and τ

   Can be removed to simplify the system. Their absence of influence can be explained.

6.4 Conclusions

The results of the sensitivity analysis have been commented in two ways. First, the outputs have been analysed separately in the aim to keep the outputs the most suitable to tune the system. Second, the inputs have been ranked according to their global impact on the system. It yields to a strategy of inputs tuning which ensures the system is performing well.

First, the outputs are selected according to the interaction share in their influence and the number of inputs having a significant influence. The time of execution is the most suitable to tune the inputs. Indeed, only N is influential and it is almost not affected by interactions. But it is more a trade-off output than a real quality score. The wind spectrum slope is influenced by σ^{add} and σ^{obs}, by C_0 and C_1 next, with a very small share of interaction. The wind RMSE is influenced by σ^{add}, N, C_0 and σ^{obs}. Some interactions are influential, but they involve only the parameters which are already influenced alone. Hence, the interactions bring no additional side-effect. The number of null potential and the TKE RMSE are strongly affected by interactions involving many inputs. They are not selected to tune the system.

Second, the outputs have been gathered to study the inputs from a global point of view. The inputs are selected according to the side-effect expected when they are moved. It is scored by their average first order Sobol index and their interaction share. Four groups of inputs have been identified. The first group is composed of σ^{add}, N and σ^{obs}. They are the most influential variables, especially on the best outputs that are b and r_V. The second group is composed of C_0 and C_1. They have a large influence, but their interaction share imply side-effects. The turbulence modelling brings additional knowledge about them. The third group is composed of ℓ and σ^X. They are not very influential alone, but are involved in strong interactions. The fourth and last group is composed of σ^V and τ. They are not influential.

Putting all these conclusions together, it yields to the following strategy to ensure the system is performing well. A relevant reduction of the system would be to keep the inputs σ^{add}, N and σ^{obs} and the outputs T_{exe}, b and r_V. The inputs C_0 and C_1 can be correctly set thanks to turbulence modelling insights. The observation noise is unknown but constant. First, the output b is used to set σ^{obs}. Next, N is set by trade-off between the outputs r_V and T_{exe}.

Chapter 7

Exploration with 2-by-2 experiments

Contents

7.1 Introduction

7.1.1 What are 2-by-2 experiments ?

The idea of 2-by-2 experiments is the following. After a first work of development, a new system (denoted by a function f, such as in figure 6.1) is built and a sensitivity analysis is desired. The first phase of development usually provides a set of values for the input parameters for which the system globally works and a guess on the range in which they belong. The sensitivity analysis will explore the influence of the input parameters when they browse this range.

Sensitivity indices (such as Sobol indices) provide a quantification of the influence of an input by a single value, without any assumption on the system. But a single value is a very

short summary of the influence. The form of the function f is a much more fertile knowledge on the system because it can be interpreted.

So called "One-At-Time" (OAT) methods are very informative about the form of f. Using the nominal values and the range of variations provided by the development phase, OAT methods make one input vary while the others are set to a nominal value. By doing so, it is possible to draw the evolution of outputs when this input varies. But this method looses the effect of interactions between inputs (Saltelli and Annoni, 2010) and there is a chance that the nominal value of another input hinders the effects of the moving input.

On another hand, let all the inputs vary is the good method to catch all variations, as we have seen in sensitivity analysis. But the shape of f is not accessible when all inputs are moving. Only projections can be drawn, and they are poisoned by the variations of others parameters.

Let 2 inputs vary is a compromise to keep the drawing capacity of OAT and the possibility to take interactions into account. Such experiments are denoted as *2-by-2 experiments*. It is a way to check the main features pointed out by the sensitivity analysis. The expected result is to find the mechanism at the origin of the influence in order to control it (to reduce uncertainty or to tune the system to various situations). The total experimental plan is summarized in the table 7.1, but the resulting figures are too numerous to be all included in the main matter. To make browsing easier, hyperlinks to the appendix are inside the table 7.1. The experimental plan can also be represented by a graph: vertices are input parameters and edges exists between two inputs for which a 2-by-2 experiment has been carried out. The resulting graph of the present experimental plan is shown figure 7.1. For each edge of the graph 7.1, the five outputs have been computed. The inputs connected by the edge are the only ones to move, the others stay at their nominal value. Both nominal values and ranges of variation are recap in the table 7.2.

	C_0	C_1	ℓ	N	σ^{add}	σ^{obs}	σ^V	σ^X	τ
C_0		p.303	p.306						
C_1	p.303								
ℓ	p.306			p.312	p.315	p.318			
N			p.312		p.321	p.324			p.327
σ^{add}			p.315	p.321		p.330			p.333
σ^{obs}			p.318	p.324	p.330			p.336	p.339
σ^V								p.342	
σ^X						p.336	p.342		
τ				p.327	p.333	p.339			

Table 7.1 – Couples of inputs experimented: results are on the indicated page (hyperlink). This table is a copy of C.3.

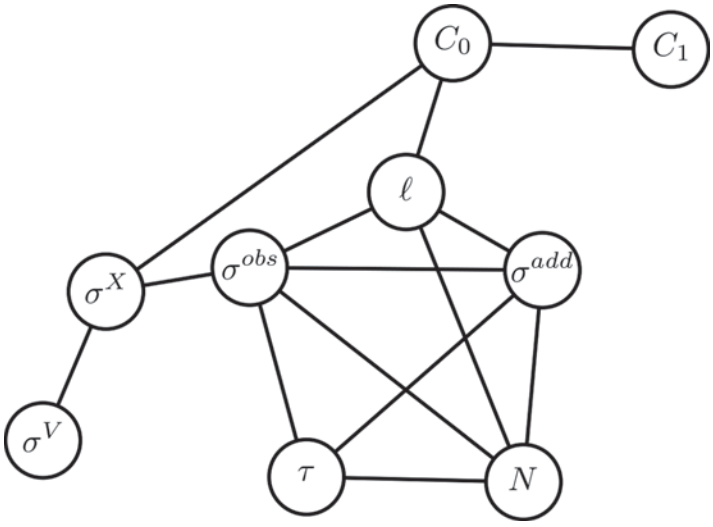


Figure 7.1 – Graph of 2-by-2 experiments: vertices are input parameters and edges exist between inputs involved in a common 2-by-2 experiment. The annotated numbers are the degree of the vertices.

Input	**Min**	**Max**	**Nominal value**	**Unit**
C_0	0.3	2.5	2.1	none
C_1	0	2.5	0.9	none
ℓ	3	100	10	m
N	400	2500	700	none
σ^{add}	0.1	2.1	0.5	$\mathrm{m{\cdot}s^{-1}}$
σ^{obs}	0.1	2.1	0.5	$\mathrm{m{\cdot}s^{-1}}$
σ^V	0	1.1	0.1	$\mathrm{m{\cdot}s^{-1}}$
σ^X	0	11.1	1.0	m
τ	5	30	10	min

Table 7.2 – Range of variation and nominal value for each input.

7.1.2 Link with global sensitivity analysis

For global sensitivity analysis, the computer code is seen as a function of all the parameters. The Hoeffding decomposition of this function allows to attribute the variance on the output to a group of parameters.

$$Y = f(X) = \sum_{\mathbf{u} \in I} f_{\mathbf{u}}(X_{\mathbf{u}}) \tag{7.1}$$

For a 2-by-2 experiment, we look at the same function but with only two parameters moving. The others are fixed to their nominal value. The 2 parameters varying are denoted X_i and X_j. It is also possible to write the Hoeffding decomposition on this new model, which

has only 3 terms :

$$\tilde{Y} = \tilde{f}(X_i, X_j) = \tilde{f}_i(X_i) + \tilde{f}_j(X_j) + \tilde{f}_{i,j}(X_i, X_j) \tag{7.2}$$

The two models are linked by the fixation of the parameters $X_{\overline{i,j}} = \{X_k,\, k \in [\![1, p]\!], k \neq i, j\}$ to their nominal value $x_{\overline{i,j}}$.

$$\mathcal{L}(\tilde{Y}) = \mathcal{L}(Y | X_{\overline{i,j}} = x_{\overline{i,j}}) \tag{7.3}$$

Because the Hoeffding decomposition is unique, one can identify the terms :

$$\tilde{f}_i(X_i) = \sum_{\mathbf{u} \in I, i \in \mathbf{u}} f_{\mathbf{u}}(X_i, x_{\mathbf{u} \setminus i}) \tag{7.4}$$

$$\tilde{f}_{i,j}(X_i, X_j) = \sum_{\mathbf{u} \in I, \{i,j\} \in \mathbf{u}} f_{\mathbf{u}}(X_i, X_j, x_{\mathbf{u} \setminus \{i,j\}}) \tag{7.5}$$

But this relationship are not exploitable without additional assumptions. The main interest of 2-by-2 experiments is to allow a visualization of the response surface and thus to infer about the shape of the function f.

7.2 Validation of the tuning strategy

The conclusions of the sensitivity analysis presented in the previous chapter sustain the possibility to reduce the system to only 3 informative outputs and 3 influential inputs. The reduced system has two degrees of freedom: σ^{add} (which is unknown in practice) and the affordable time of execution T_{exe}. It states a tuning strategy for the 3 main inputs which ensures the reconstruction is then performing well. The tuning strategy consists in setting σ^{obs} with the wind spectrum slope and then to set N by a trade-off between the affordable time of execution T_{exe} and the desired precision r_V.

7.2.1 Setting σ^{obs} with wind spectrum slope

We have seen is the chapter 1 (REF +précise), that the wind spectum has a characteristic -5/3 slope in log-log scale (see figure 5.32 for an illustration). This output has been shown to be affected by the inputs σ^{add} and σ^{obs}, mainly. It is poorly affected by interactions so that it is a useful output to tune the inputs. The input σ^{add} is the error made by the instrument, which is unknown in practice. Conversely, σ^{obs} is the guess of this error, and it is the parameter used in the algorithm instead of σ^{add}. We can expect the system to perform the best when the guess σ^{obs} is equal to the true value σ^{add}.

Figure 7.2 shows the result of the 2-by-2 experiment when σ^{obs} and σ^{add} are the only input moving. The displayed output is the wind spectrum slope b. One can see the clear influence of both variable. The red plan in the middle of the figure stands for the theoretical -5/3 value. The black line represents the equality of the two inputs. In the area where $\sigma^{obs} < \sigma^{add}$ (right hand side of the figure, where the ground is red), the wind spectrum slope is higher than the expected value. Indeed, the true observation noise (σ^{add}) is higher than its guess (σ^{obs}) thus the filter let some noise left in the estimation. Conversely, in the area where $\sigma^{obs} > \sigma^{add}$ (left hand side of the figure, where the ground is blue), the wind spectrum slope is lower than the expected value. There, the filter overestimates the amount of noise and removes to much power in the highest frequencies. The sensitivity analysis and this response surface show the wind spectrum slope is very sensitive to this setting. One can see that the response surface seems to cross the -5/3 value when σ^{obs} and σ^{add} are equal. This feeling is confirmed by a look at the cross-sections in figures 7.3 and 7.4.

Figure 7.3 shows the evolution of the output b against σ^{add}, for different values of σ^{obs}. Each solid curve corresponds to a different value of σ^{obs} (precised in the key). The -5/3 value is the horizontal dashed line. The vertical dashed lines are where σ^{add} is equal to one of the value of σ^{obs} displayed. One can see that the solid lines cross the horizontal dashed line when $\sigma^{obs} = \sigma^{add}$ (it is less clear for small values of σ^{obs}). The observation is the same for the figure 7.4 which displays the evolution of the output b against σ^{obs}, for different values of σ^{add}. It sustains that the output b has the expected value when $\sigma^{obs} = \sigma^{add}$.

It also confirms the tuning strategy. For a given instrument, σ^{add} is fixed. The parameter σ^{obs} should be set the same value. The strategy consists in drawing the output b against σ^{obs}

and to pick the value of σ^{obs} which gives the wind spectrum slope the closest to -5/3.

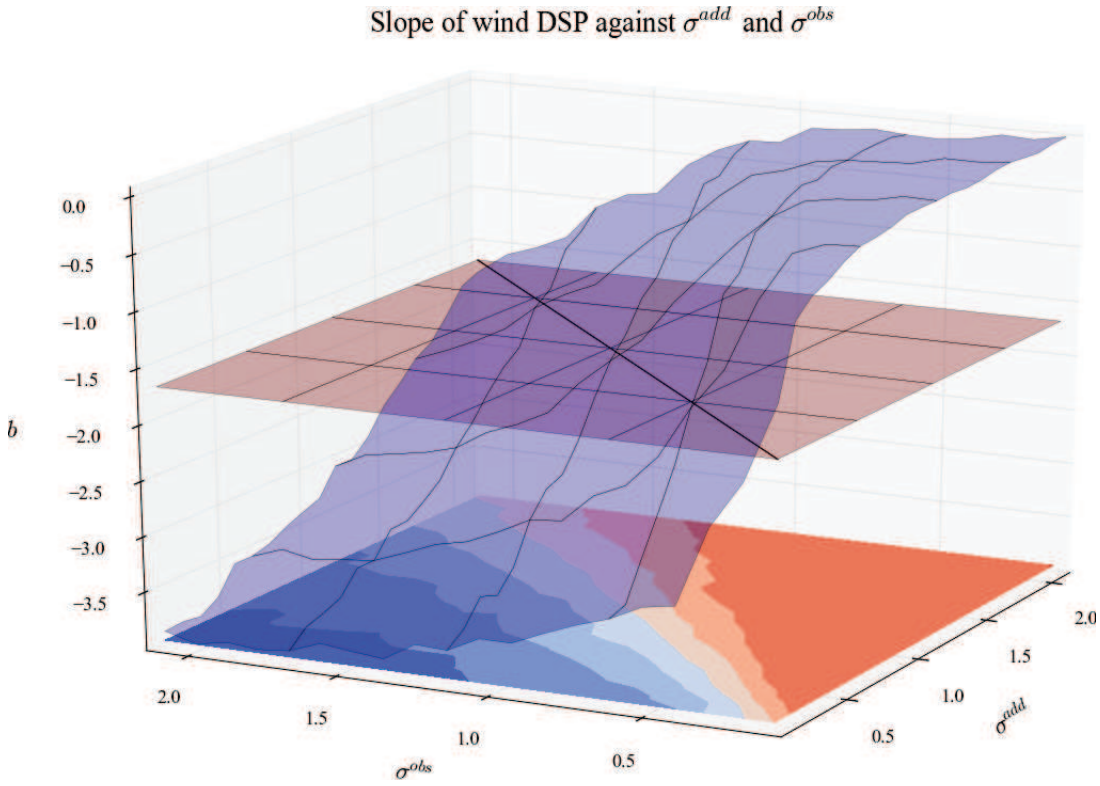


Figure 7.2 – Evolution of b when only σ^{add} and σ^{obs} vary. The sampling grid has 20 values of σ^{obs} and 20 values of σ^{obs} (400 points in total). The red plan is at the level $b = -5/3$ (theoretical expected value).

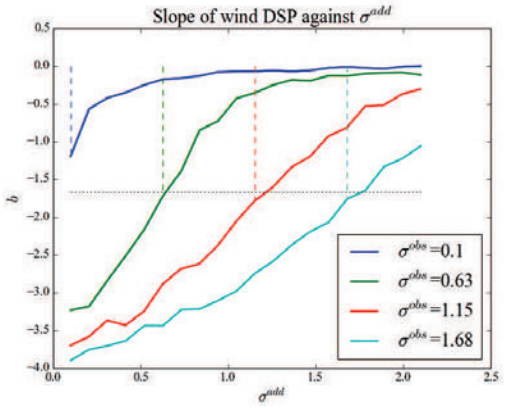


Figure 7.3 – Evolution of b when σ^{add} vary, for different values of σ^{obs}. Horizontal dotted line is $b = -5/3$. Vertical dashed lines signalize when $\sigma^{add} = \sigma^{obs}$ for each value of σ^{obs}.

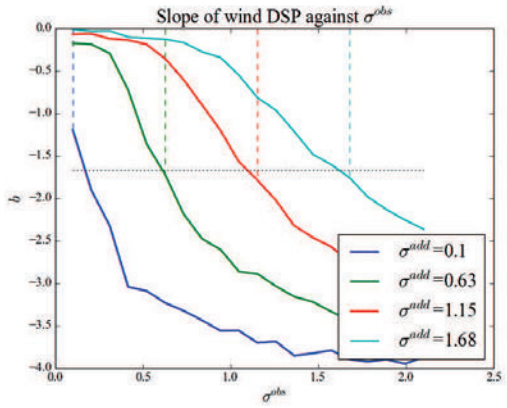


Figure 7.4 – Evolution of b when σ^{obs} vary, for different values of σ^{add}. Horizontal dotted line is $b = -5/3$. Vertical dashed lines signalize when $\sigma^{add} = \sigma^{obs}$ for each value of σ^{add}.

7.2.2 Variations of the wind RMSE against the main inputs

Among the 5 outputs, the TKE RMSE and the number of null potential have been dismissed because of their complex variations, unsuitable to a tuning strategy. The execution time depends only on N and the wind spectrum slope is exploited to set σ^{obs} equal to σ^{add}. The wind RMSE is thus the only relevant score to assess the wind retrieval with reconstruction. It has been shown that this score depends mostly on the first group of parameters: N, σ^{add} and σ^{obs}. Since it also have been shown that the influence of these inputs could be complex because of interactions, the 2-by-2 experiments will be fully exploited here. Three 2-by-2 experiments are required to completely visualize the influence and the interactions of these three main inputs. The figure 7.5 (respectively 7.6 ; 7.7) shows the variations of the wind RMSE when only N and σ^{add} (repectively N and σ^{obs} ; σ^{add} and σ^{obs}) are moving.

In the figure 7.5, one can see the wind RMSE steadily decreases with N, whatever the value of σ^{add}. The theorem 5.1 predicts a $1/\sqrt{N}$ decrease of r_V, which look confirmed here, independently from σ^{add}. The effect of σ^{add} is to increase the RMSE. But one can see that this increase is not linear. As long as $\sigma^{add} < 0.5$, the RMSE does not increase and then increases rapidly (the increasing speed depends on N). The 0.5 threshold is interesting because it is the nominal value of σ^{obs}.

In the figure 7.6, one can see the effect of N is the same as in figure 7.5. The effect of σ^{obs} is interesting. It shows a minimum around the value $\sigma^{obs} = 0.5$ which corresponds to the nominal value of σ^{add}. It sustains that the reconstruction is performing the best when $\sigma^{obs} = \sigma^{add}$.

Figure 7.7 crosses the effects of σ^{obs} and σ^{add} already observed. Two areas are to consider: when $\sigma^{obs} < \sigma^{add}$ and $\sigma^{obs} > \sigma^{add}$. As long as $\sigma^{add} < \sigma^{obs}$, there is no effect of σ^{add} on

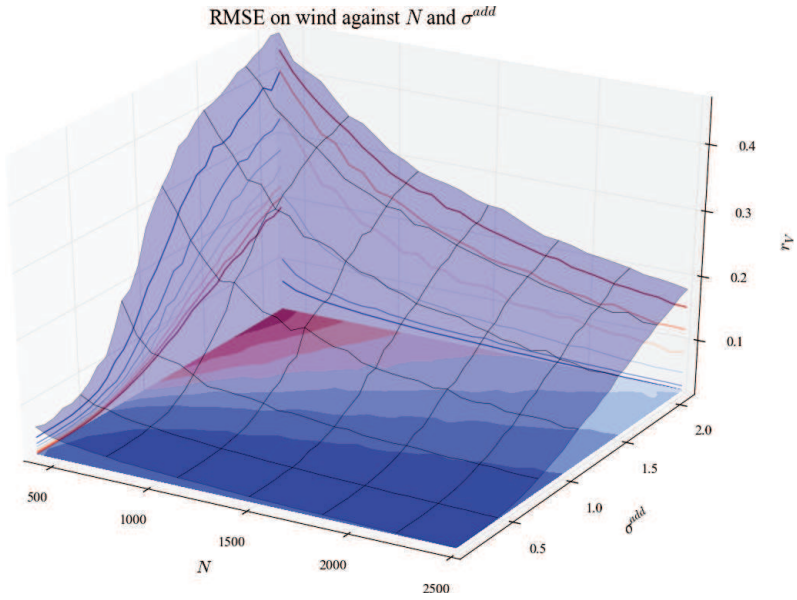


Figure 7.5 – Evolution of r_V when only N and σ^{add} vary.

the RMSE and σ^{obs} makes it increase linearly. When $\sigma^{obs} > \sigma^{add}$, the evolution of the wind RMSE is much more complex. It fastly increases with σ^{add}. The effect of σ^{obs} is in two stages: for very small σ^{obs} the RMSE increases and reaches a maximum and then decreases up to the point where $\sigma^{obs} = \sigma^{add}$ (after what it increases again, as mentioned previously). In any case, there is a low on the line $\sigma^{obs} = \sigma^{add}$.

The best choice for σ^{obs} is thus to be equal to σ^{add}. We have seen this can be obtained quite safely by the tuning with the wind spectrum slope. Once this setting has been made, one can see the evolution of wind RMSE against σ^{add} and N in figure 7.8. The RMSE increases linearly with the observation noise and decreases as $1/\sqrt{N}$, as the regressions in figure 7.9 show. Hence the wind RMSE can be approached by the relation (7.6) when the input σ^{obs} is correctly set.

$$r_V = K\frac{\sigma^{add}}{\sqrt{N}} \tag{7.6}$$

The constant K is estimated by ordinary least squares on the points of the surface 7.8 The resulting value is $K = 2.33$. One can see in the figure 7.9 that the wind RMSE is always lower that the input σ^{add}, even for very low N. For example, if one has a lidar making a error $\sigma^{add} = 1.19$ m·s^{-1} (middle line in figure 7.9), the error on the wind at the output of the filter is below 0.2 m·s^{-1}, even with only 500 particles. It reduces the noise about 83% for the lowest N tested and rate raises up to 93% with $N = 2500$. It shows the efficiency of the wind retrieval with the reconstruction. Even for an instrument quite noisy, the filter lessens strongly the inaccuracy on the wind.

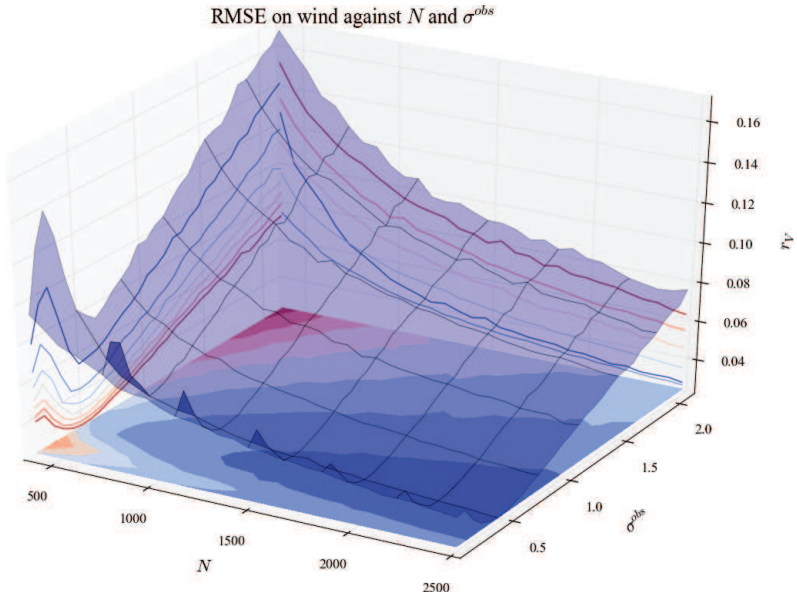


Figure 7.6 – Evolution of r_V when only N and σ^{obs} vary.

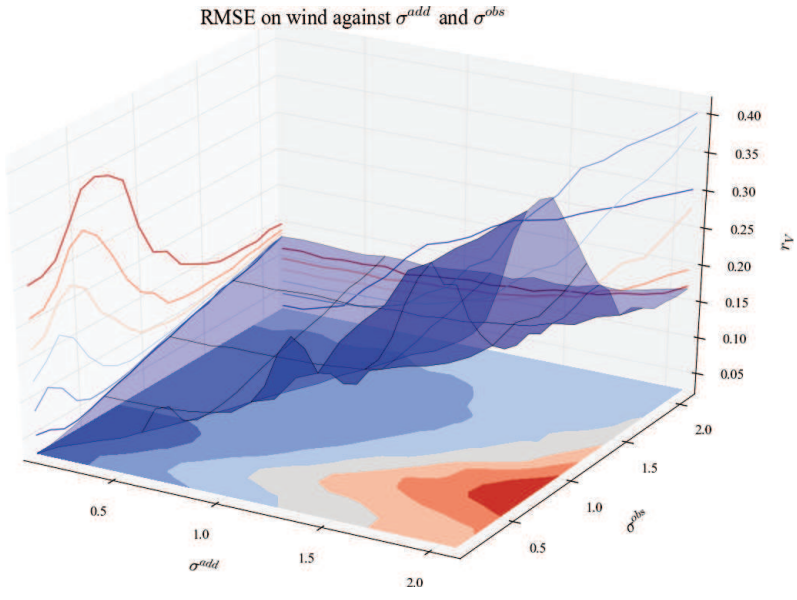


Figure 7.7 – Evolution of r_V when only σ^{add} and σ^{obs} vary.

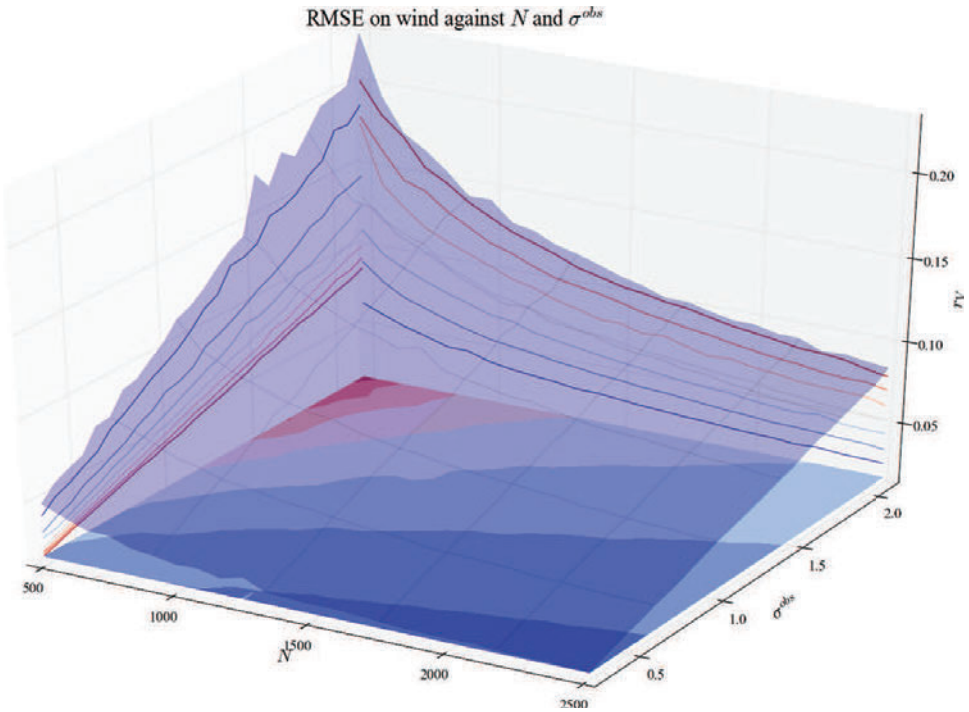


Figure 7.8 – Well set case: evolution of r_V when only N and σ^{obs} vary with $\sigma^{obs} = \sigma^{add}$.

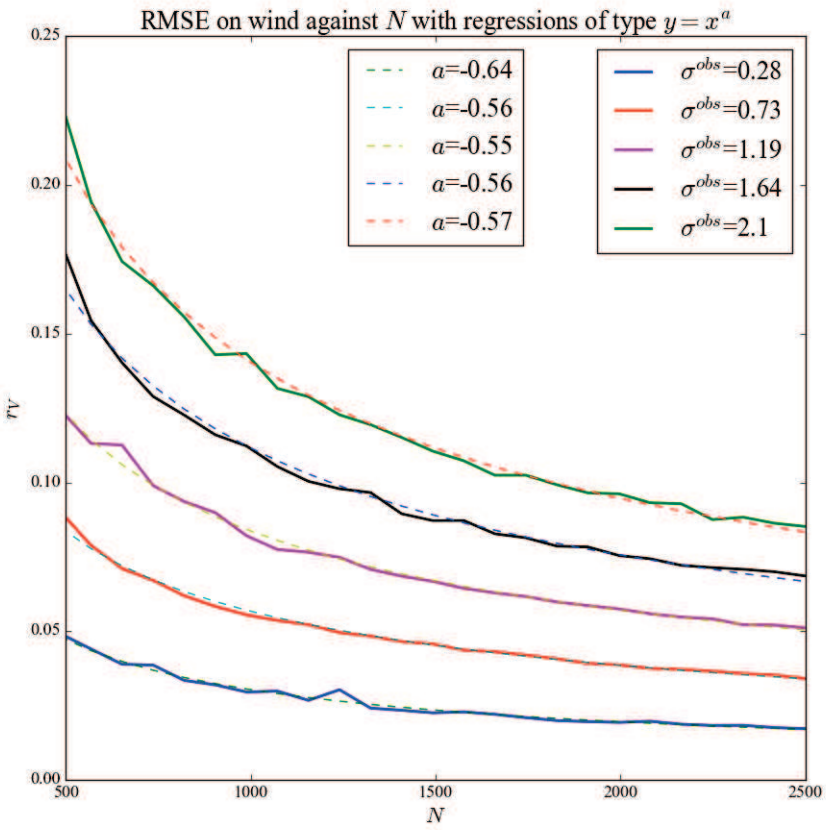


Figure 7.9 – Evolution of r_V with N when $\sigma^{obs} = \sigma^{add}$. Regressions (dashed lines) show the observed decrease is close the square root, as predicted by the theory.

7.2.3 Effect of N on the execution time

The number of particles is the only parameter to have an influence on the execution time T_{exe}. The 2-by-2 experiments confirm this claim (see the results for T_{exe} in the appendix, page 361). By regression, as shown in the figure 7.10, the rate of increase seems to be a power law:

$$T_{exe} = N^a \tag{7.7}$$

The parameter a is estimated by ordinary least squares is $a = 1.75$. According to this relation, twice the number of particles multiply by 3.36 the execution time. The other regression tested is exponential ($T_{exe} = 2^{N/a}$), but it seems less in agreement with the observations. Nevertheless, according to the exponential relationship, the time of execution should double each 515 particles. They are key figures to dimension numerical experiments.

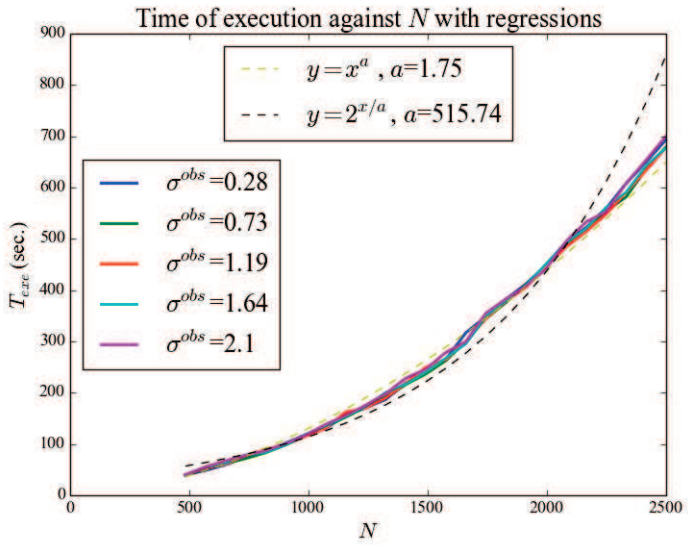


Figure 7.10 – Evolution of T_{exe} with N for different values of σ^{obs}. Regressions (dashed lines) show the best fit is a power function.

7.2.4 Tuning strategy for the reconstruction

The best wind retrieval is thus obtained with the following three inputs values:

- σ^{add} the smallest,
- σ^{obs} equal to σ^{add},
- N the highest.

Although σ^{add} is fixed by the instrument, this result helps to assess which precision is affordable for a given instrument. Even for inaccurate instrument, the resulting error is much lower

than the raw error. The number of particles is bounded by the affordable time of execution. Examining these results, the following tuning strategy seems to be the most appropriate:

1. Set N to a low value, such that T_{exe} is really small.
2. For σ^{obs} ranging around the a priori accuracy of the instrument, calculate the wind spectrum slope b.
3. Set σ^{obs} to the value which gives b the closest to -5/3. σ^{obs} is then almost equal to σ^{add}.
4. Set N to the maximum affordable value. The error on wind retrieval is now minimum, estimated by $K\frac{\sigma^{obs}}{\sqrt{N}}$ with $K = 2.33$

7.3 Other interesting results

7.3.1 On the bad construction of the score r_k

After the sensitivity analysis, the output r_k (TKE RMSE) have been shown to have very complicated behaviour, including high order interactions. As a consequence, it cannot be used to tune the output. We will see at the examination of the 2-by-2 experiments that the interpretation of the score r_k itself is also complex, which makes it an irrelevant score.

Figure 7.11 depicts the surface response when only ℓ and σ^{add} vary. σ^{add} is the most influential input. Although its interaction with ℓ does not have a high Sobol index, one can see in the figure 7.11 that they are clearly interacting. The effect of σ^{add} depends on the value of ℓ. The shape of the response surface is complex and no clear interpretation can explain it. When ℓ is large, the local average is no longer local and converges to a spatial average. When ℓ is large, the RMSE r_k reaches a minimum when $\sigma^{add} = 0.9$. The value of σ^{add} where this minimum is reached increases when ℓ decreases. One of the unclear interpretations about this could be that RMSE on TKE is the lowest when the instrument has a variance of error comparable to the ambient turbulence. This could be checked by changing the reference wind in the system, but it has not been done here. The effect of ℓ is due to the local average used to compute the TKE. By using another TKE estimator, for example the STKE defined in the section 1.4, one should retrie the same evolution as ℓ is maximum.

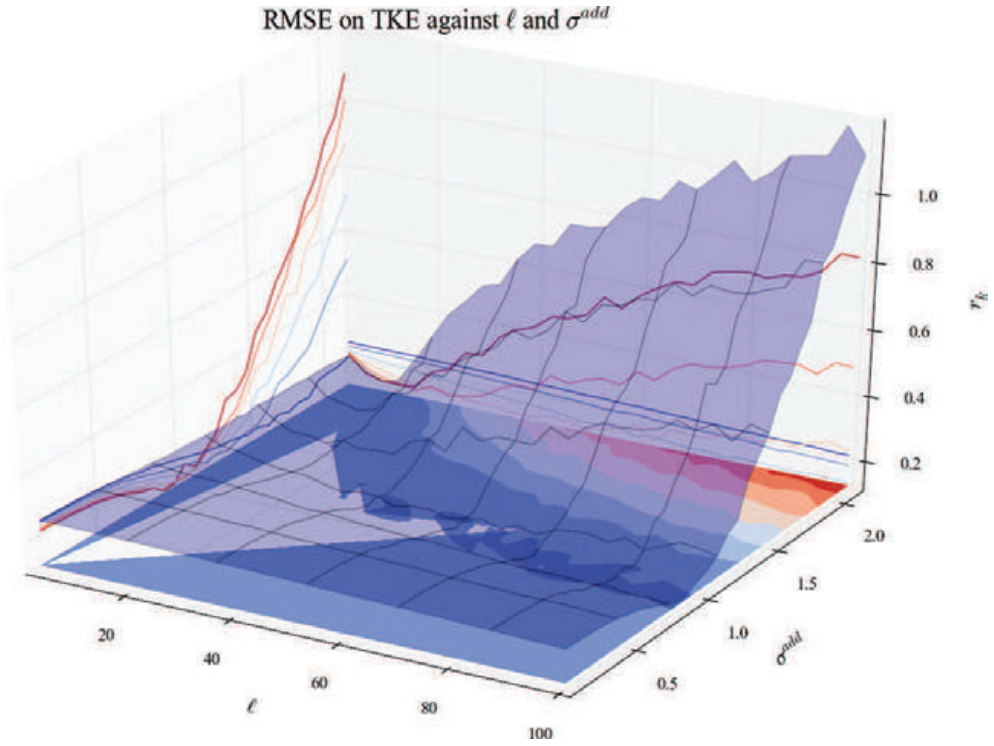


Figure 7.11 – Evolution of r_k when only ℓ and σ^{add} vary.

Figure 7.12 is the response surface when only N and τ vary. Both have quite small Sobol

indices, but τ showed an asymmetric influence on the cobweb plot 6.25. In the figure 7.12 one can see that τ does have an influence, while N has no influence at all. The non influence of N is retrieved in the others 2-by-2 experiments in the appendix. The weird effect of τ is confirmed as well by others 2-by-2 experiments in the appendix. This effect is due to the construction of the score r_k. The time average operator introduce some threshold effect that is visible here. This effect makes difficult the interpretation of the score.

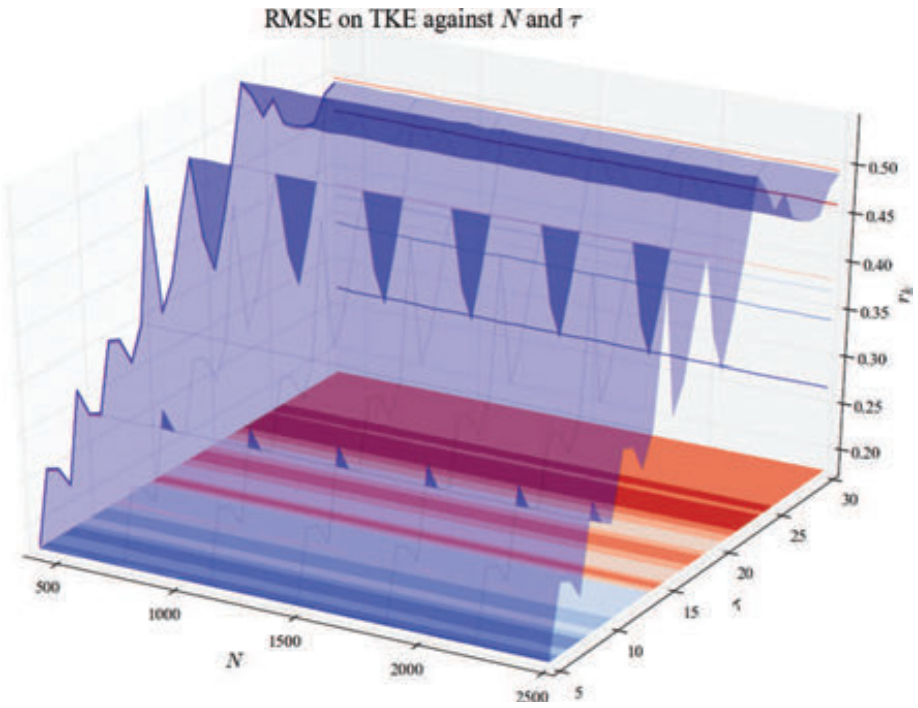


Figure 7.12 – Evolution of r_k when only N and τ vary.

Finally, the figure 7.13 shows the response surface r_k against σ^{add} and σ^{obs}. r_k is minimum when $\sigma^{add} = 1.6$ and σ^{obs} is not too small. When σ^{obs} is not too small, this minimum disappears. There is no low on the line $\sigma^{obs} = \sigma^{add}$, as for the wind RMSE. It tells that knowing perfectly the instrument error does not help to improve r_k. Whatever, is the guess of the instrument error σ^{obs}, if the real σ^{add} is not at the good value (related to ℓ as seen in 7.11 and probably related to the ambient turbulence), the TKE retrieval will not work. This is a very bad issue for this score because it depends on arbitrary parameters (ℓ) or uncontrollable (ambient turbulence) or unknown (if the previous reasoning are wrong). In any case, this output is hardly useful in practice.

The 2-by-2 experiments give more insight about the variations of r_k. But they are complex and there is no satisfying interpretations for them. Unverified interpretations could be that the score r_k depends on the balance between the instrument error σ^{add} and the ambient turbulence, and not on the balance between σ^{add} and its guess σ^{obs}. Hence, the conclusion of this analysis is that the score r_k is not well constructed and is not very informative about the system.

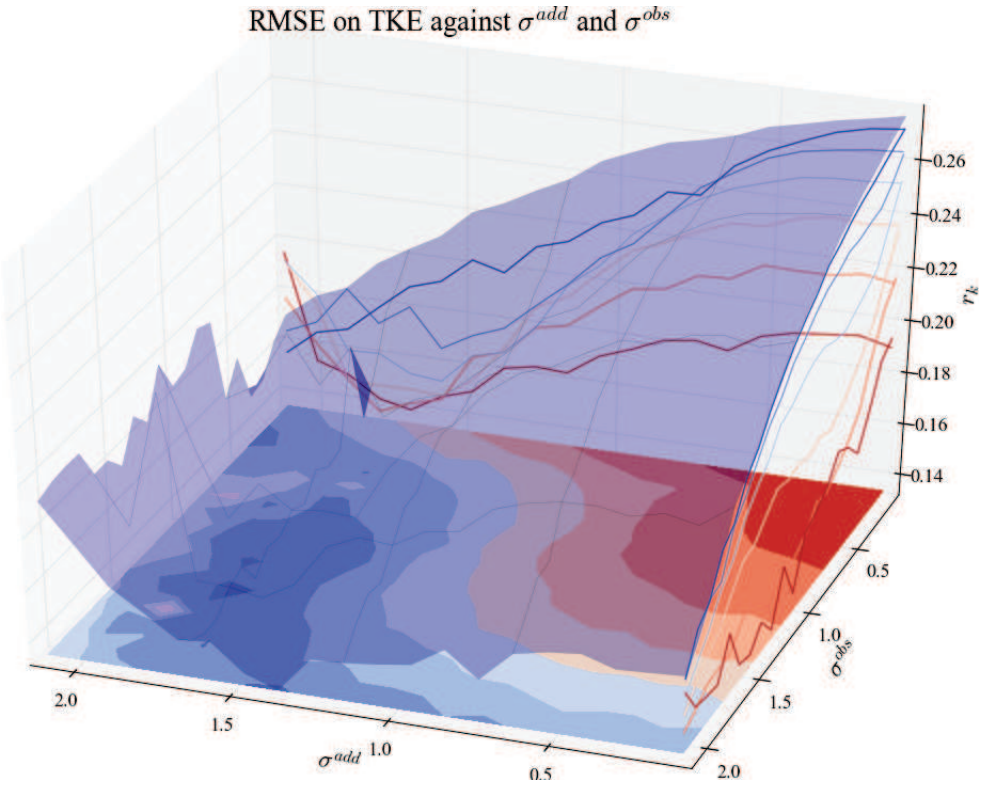


Figure 7.13 – Evolution of r_k when only σ^{add} and σ^{obs} vary.

7.3.2 Influence of C_0 and C_1

The inputs C_0 and C_1 have been put in the second group of influence in the last chapter. They have a notable influence but the interaction share was the argument to put them out of the first group. Another argument was that turbulence modelling provide estimation of them. In particular, Pope (1994) proposes the relationship $C_1 = \frac{1}{2} + \frac{3}{4}C_0$ with the value $C_0 = 2.1$. Nevertheless, the 2-by-2 experiments can give more insight about their influence and check if such relationship is visible in the outputs.

On the output b (wind spectrum slope) they were mentioned to be influential. Figure 7.14 shows the evolution of b when only C_0 and C_1 vary. One can see the response surface crosses the -5/3 value around the thick line, which has for equation $C_1 = \frac{1}{2}C_0$. The dashed line has the equation suggested by Pope $C_1 = \frac{1}{2} + \frac{3}{4}C_0$. The black dot is where the nominal values have been chosen. The equation suggested by Pope looks wrong in this figure because it has been designed for 3D flows while this one is only 1D. The nominal values are close to optimal line.

On the output N_{G0} (number of null potential), C_0 and C_1 were spot as influential (respectively second and third total Sobol index in figure 6.3). In the figure 7.15 is displayed the evolution of N_{G0} against C_0 and C_1. Although the sensitivity analysis let think that N_{G0} has complex variations when C_0 and C_1 vary (because of interactions), the response surface is

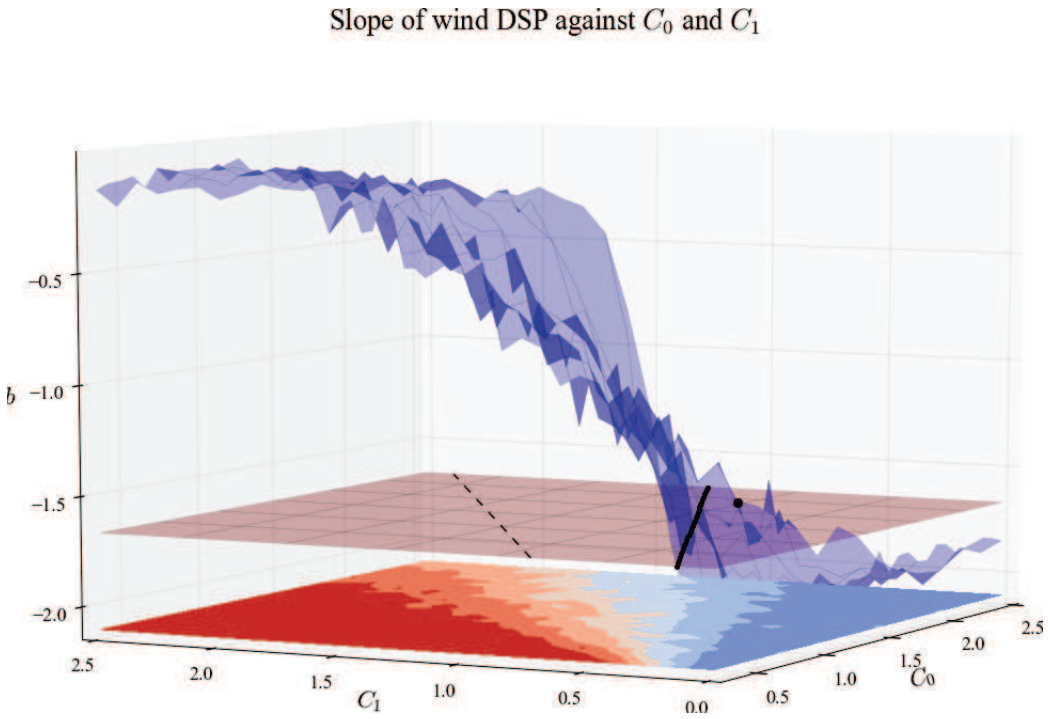


Figure 7.14 – Evolution of wind spectrum slope when only C_0 and C_1 vary. The black dotted line are the points where $C_1 = \frac{1}{2}C_0$. The dashed line has for equation $C_1 = \frac{1}{2} + \frac{3}{4}C_0$. The dot denote the nominal values.

very smooth. One can see that a range of values are admissible (they give no null potential) and another is not. The same lines as in the previous figures are plotted ($C_1 = \frac{1}{2} + \frac{3}{4}C_0$, Pope's result, is the dashed line ; $C_1 = \frac{1}{2}C_0$, the convenient equation for the wind spectrum, is solid dotted). The line that was admissible for the wind spectrum ($C_1 = \frac{1}{2}C_0$, solid dotted) is admissible here too. The line given by Pope for 3D flows ($C_1 = \frac{1}{2} + \frac{3}{4}C_0$, dashed) is not very convenient according to this score neither. The nominal values ($C_0 = 2.1$, $C_1 = 0.9$) are in the admissible area too, close to the dotted line.

On the output r_V (wind RMSE), they are also in good position in figure 6.21. The figure 7.16 displays the evolution of wind RMSE when only C_0 and C_1 vary. The same lines as in the two previous figures are plotted. As for the two previous figures, the line $C_1 = \frac{1}{2}C_0$ is admissible (it gives low RMSE) while the line $C_1 = \frac{1}{2} + \frac{3}{4}C_0$ is not. The results are thus consistent among the 3 exploitable outputs. One more thing about the figure 7.16: it shows the extreme value $C_1 = 0$ always gives the lowest RMSE, which is strange, because it suggests the model is better without the fluctuation term.

On the output r_k the inputs C_0 and C_1 have a contrary effect, as shows the figure 7.17. The line $C_1 = \frac{1}{2}C_0$ gives the highest TKE RMSE, while the line $C_1 = \frac{1}{2} + \frac{3}{4}C_0$ gives lower values (still not minimum). The influence of C_0 and C_1 on r_k is antagonist to the influence on r_V. But the sensitivity analysis raised many questions on the reliability of the score r_k. It has been shown to be influenced in a complex way. Moreover, the range of the variations due

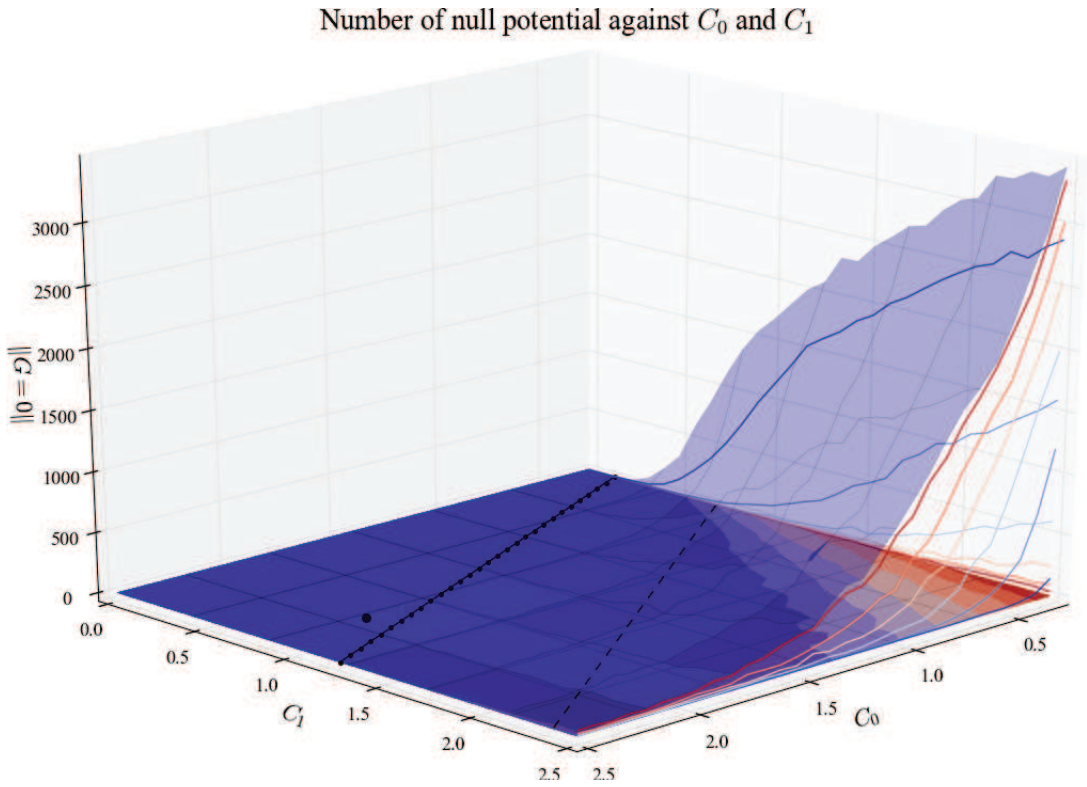


Figure 7.15 – Evolution of N_{G0} when only C_0 and C_1 vary. The black dotted line are the points where $C_1 = \frac{1}{2}C_0$. The dashed line has for equation $C_1 = \frac{1}{2} + \frac{3}{4}C_0$. The bullet denotes the nominal values.

to C_0 and C_1 is not as broad as other inputs can give. Hence, to set C_0 and C_1, the choice is made to choose the effect on r_V rather than the effect on r_k.

Further examination of the influence of parameters C_0 and C_1 with 2-by-2 experiments confirm the influence that was suspected with Sobol indices. The response surfaces are smoother than what was expected according the interactions in which they are involved. The wind spectrum gives the sharpest criterion to choose C_0 and C_1. By visual examination, the relation $C_1 = \frac{1}{2}C_0$ should be fulfilled to have a -5/3 spectrum slope. This criterion is valid for the output N_{G0} and r_V too. However, it appears to be the worst choice for the output r_k. But this output suffers from many weakness in its construction. Hence, the effect of r_k is ignored. Thus, the 2-by-2 experiment concludes that C_0 and C_1 must fulfil the relation $C_1 = \frac{1}{2}C_0$ to be well set. The nominal point ($C_0 = 2.1$, $C_1 = 0.9$) does not fulfil exactly, it should be improved easily.

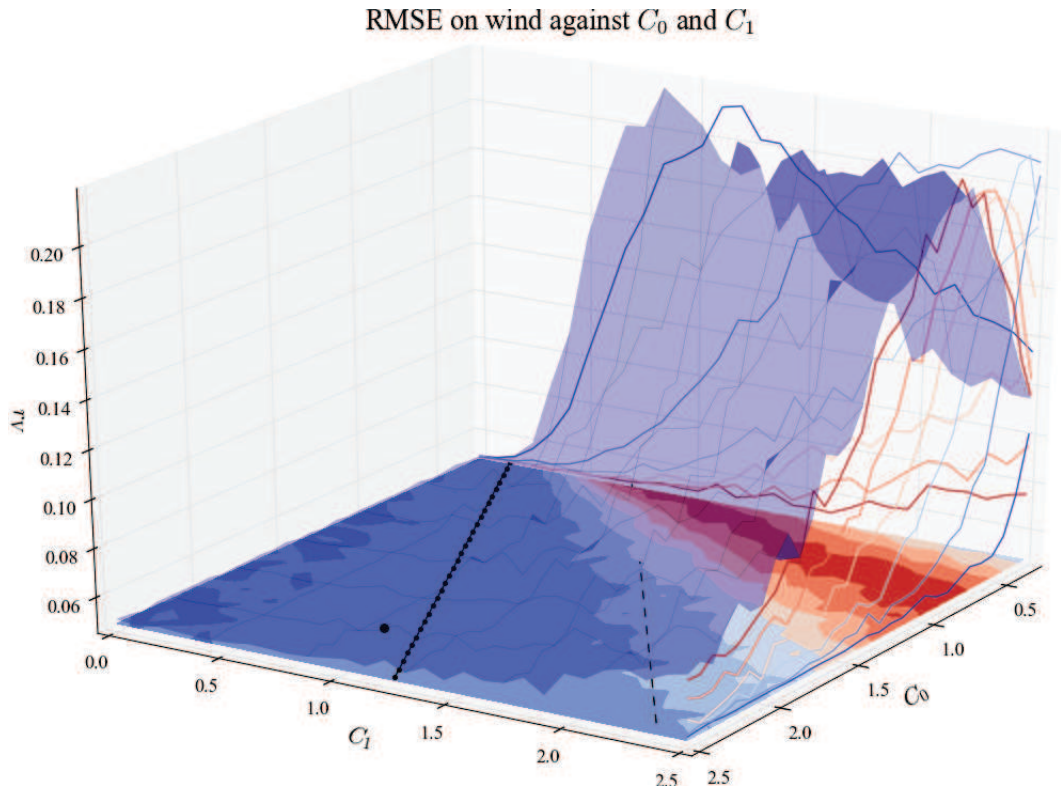


Figure 7.16 – Evolution of wind RMSE when only C_0 and C_1 vary. The black dotted line are the points where $C_1 = \frac{1}{2}C_0$. The dashed line has for equation $C_1 = \frac{1}{2} + \frac{3}{4}C_0$. The bullet denotes the nominal values.

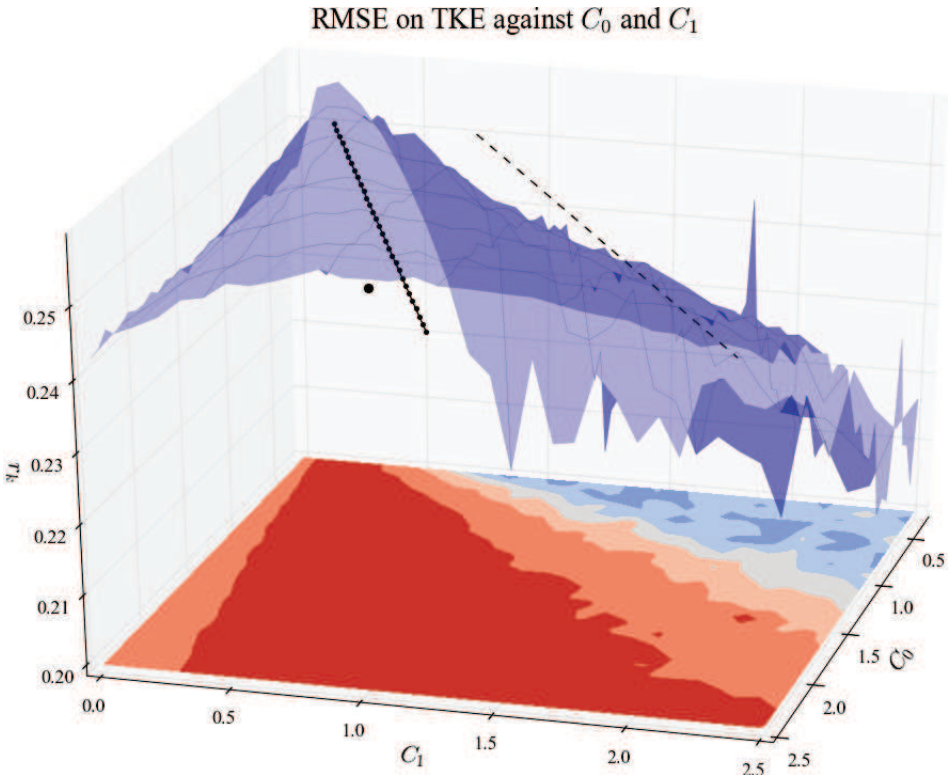


Figure 7.17 – Evolution of TKE RMSE when only C_0 and C_1 vary. The black dotted line are the points where $C_1 = \frac{1}{2}C_0$. The dashed line has for equation $C_1 = \frac{1}{2} + \frac{3}{4}C_0$. The bullet denotes the nominal values.

7.3.3 Retrieval of known variations of N_{G0}

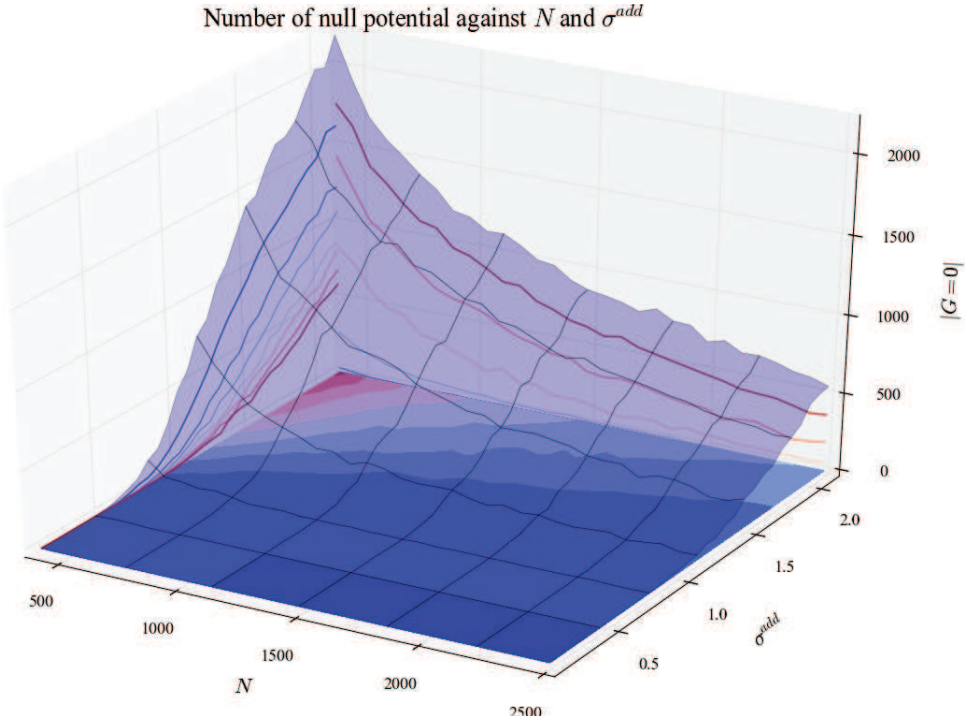


Figure 7.18 – Evolution of N_{G0} when only N and σ^{add} vary.

The influence of N, although not visible in the Sobol indices (both in figures 6.3 and 6.41), is quite clear in figures 7.18 and 7.19. It shows a regular decrease of N_{G0} as N is rising. We retrieve the behaviour predicted by the theorem 5.2 which states an exponential decrease of N_{G0} with N.

The influence of σ^{add} and σ^{obs} is described by the theorem 5.3. This theorem gives an upper bound for the average number of null potential. This upper bound is displayed in figure 7.21. One can see this bound is very low in a large corner of the figure. The actual number of null potential, obtained when only σ^{add} and σ^{obs} vary, is displayed in figure 7.20. One can see the same large corner of very low values. This corner is delimited by the black line.

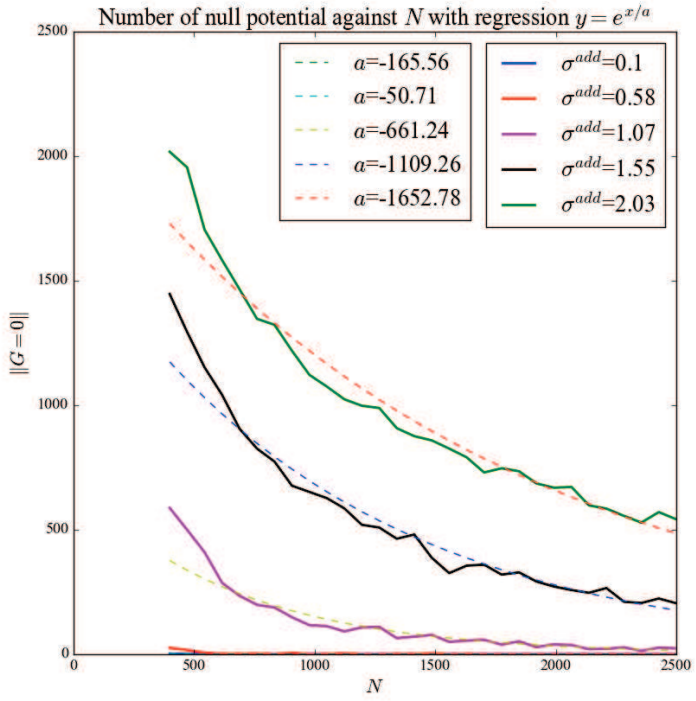


Figure 7.19 – Evolution of N_{G0} with N. Regressions show an exponential decrease.

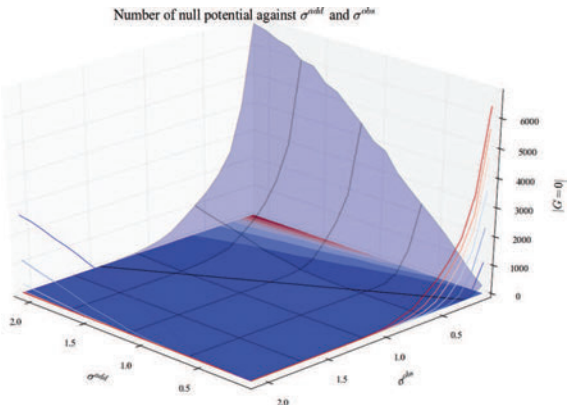


Figure 7.20 – Evolution of N_{G0} when only σ^{add} and σ^{obs} vary.

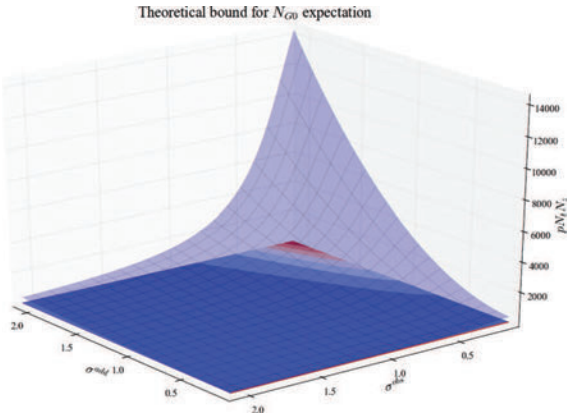


Figure 7.21 – Theoretical average for the N_{G0} output against σ^{add} and σ^{obs}.

7.4 Conclusion

So-called 2-by-2 experiments are the computation of the surface response when only 2 inputs vary and the others are kept to their nominal values. It allows visualizations of the surface response which are not possible in higher dimensions and thus help to find the shape of the response function f. Such visualizations have been used to confirm the tuning strategy coming out of the sensitivity analysis. Only few 2-by-2 experiments are necessary, thanks to the ranking of importance made by the sensitivity analysis. The surface response of wind spectrum slope and wind RMSE have been commented and confirm the following tuning strategy:

1. Set N to a low value, such that T_{exe} is really small.
2. For σ^{obs} ranging around the a priori accuracy of the instrument, calculate the wind spectrum slope b.
3. Set σ^{obs} to the value which gives b the closest to -5/3. σ^{obs} is then almost equal to σ^{add}.
4. Set N to the maximum affordable value. The error on wind retrieval is now minimum, estimated by $K\frac{\sigma^{obs}}{\sqrt{N}}$ with $K = 2.33$.

Beside, some results of the 2-by-2 experiments have been used to check remarkable points. From these additional examinations, it comes out that

- The score r_k has been shown to be not very well constructed because it is not very informative on the system.
- The inputs C_0 and C_1 have been left apart from the tuning strategy, they are explored with 2-by-2 experiments. It yields that the setting fulfilling the relation $C_1 = \frac{1}{2}C_0$ gives the best results.
- The influence of N on N_{G0} is in agreement with the theorem 5.2, although not visible in the Sobol indices.

The full results of 2-by-2 experiments have not been commented all, but they are let available to the curious reader in the appendix C, page 301.

Chapter 8

Penalised regression to estimate the Sobol indices

Contents

8.1 Sobol indices estimated by regression

8.1.1 Motivation

In the chapter 4, several estimators for Sobol indices have been presented. They are all based on a different way to estimate the average operator in the Sobol index. For example, Sobol (2001) compares 2 estimators (though denoted λ and μ in the paper, here the notations of the previous chapter are kept):

$$\widehat{D_{\mathbf{u}}}^{MC1} = \frac{1}{N}\sum_{i=1}^{N} f(X^i) f(Z^i_{\mathbf{u}}, X^i_{\bar{\mathbf{u}}}) - \left(\frac{1}{N}\sum_{i=1}^{N} f(X^i)\right)^2 \tag{8.1}$$

$$\widehat{D_{\mathbf{u}}}^{MC2} = \frac{1}{N}\sum_{i=1}^{N}\left(f(X^i) - f(Z^i_{\mathbf{u}}, X^i_{\tilde{\mathbf{u}}})\right)^2 \tag{8.2}$$

He shows that $\widehat{D_{\mathbf{u}}}^{MC2}$ has a smaller variance to estimate total Sobol indices, while $\widehat{D_{\mathbf{u}}}^{MC1}$ has a smaller variance to estimate main effect Sobol indices. Moreover, $\widehat{D_{\mathbf{u}}}^{MC2}$ is always positive, which avoid the estimation to be negative when indices are small. Saltelli et al. (2010) makes a broader comparison focused on the estimation of first order Sobol indices, and proposes another Monte Carlo estimator:

$$\widehat{D_{\mathbf{u}}}^{MC3} = \frac{1}{N}\sum_{i=1}^{N} f(X^i)\left(f(Z^i) - f(Z^i_{\mathbf{u}}, X^i_{\tilde{\mathbf{u}}})\right) \tag{8.3}$$

Owen (2013) also makes a comparison of several estimation strategies and proposes a new estimator with the concern of improving the estimation of small Sobol indices.

Among all these estimators, the estimation method of $V(\mathbb{E}[Y|X_{\mathbf{u}}])$ is modified to improve its efficiency (smaller variance, more efficient calculation) or to retrieve valuable properties (positivity, asymptotic normality). In this section, we present another type of estimation, based on a linear regression. When sensitivity analysis comes to the user, a complete set of Sobol indices is not always informative. The interpretation of the highlighted sensitivity is the final result of the sensitivity analysis. Only few coefficients are relevant to describe the contribution of variance. Poorly influential groups are not taken into account in the interpretation of the Sobol indices, even though their Sobol index is not exactly zero. Using a linear regression opens to all feature selection techniques and makes the final result easier to interpret. It also gives an alternative way to get estimators with good properties. In addition, the optimisation formulation of regression takes profit of many efficient off-the-shelves algorithms to make the minimization. Three estimators are presented and compared:

- The least squares estimator : $\widehat{S_{\mathbf{u}}}^{ls}$
- The Lasso estimator : $\widehat{S_{\mathbf{u}}}^{l1}$
- The best subset estimator : $\widehat{S_{\mathbf{u}}}^{l0}$

First, the Sobol index estimation with regression is stated in a general way. Then, the properties and the relationship among the three considered estimators are reviewed. A strategy to set the penalty is also given. Next, a small scale numerical experiment is carried out on the case of turbulent medium reconstruction.

8.1.2 Linear model statement

The notation will be the same as in the chapter 4:

- $[\![1,p]\!] = \{1,\dots,p\}$
- I is the collection of all subset of $[\![1,p]\!]$ (thus of cardinal 2^p).
- $\mathbf{u}$ is an element of I.
- $|\cdot|$ the cardinal of the set "$\cdot$". For example, $|\mathbf{u}|$ is the number of indices in $\mathbf{u}$; and $|I|$ is the number of groups of indices in I ($|I| = 2^p$).
- $I' \subset I$ is the number of subsets considered. $d = |I'| \leqslant 2^p$ is its cardinal. For example, if one is interested in first and second order Sobol indices only, one will have $I' = [\![1,p]\!] \cup \{(i,j) \in [\![1,p]\!]^2\}$.
- For all $\mathbf{u} \in I$, it is denoted $\bar{\mathbf{u}} = [\![1,p]\!] \setminus \mathbf{u} = \{i \in [\![1,p]\!], i \notin \mathbf{u}\}$
- $X = (X_1, ..., X_p)$ is the vector of random inputs (inputs are assumed independent).
- $Z = (Z_1, ..., Z_p)$ is an independent copy of X.
- $Y = f(X) = (X_1, ..., X_p)$ is the output (also random).
- $X_{\mathbf{u}} = (X_i)_{i \in \mathbf{u}}$ is the vector of random inputs in $\mathbf{u}$.
- $Z_{\mathbf{u}} = (Z_i)_{i \in \mathbf{u}}$ is an independent copy of $X_{\mathbf{u}}$.
- $Y_{\mathbf{u}} = f(Z_{\mathbf{u}}, X_{\bar{\mathbf{u}}})$ is the output when the inputs in $\mathbf{u}$ are taken from another independent realisation.

The aim is to estimate $S_{\mathbf{u}} = \frac{\mathrm{cov}(Y, Y_{\mathbf{u}})}{V(Y)}$. It is the coefficient of the slope of the linear regression between Y and $Y_{\mathbf{u}}$:

Lemme 8.1. *For any set* $\mathbf{u}$ *of indices (*$\mathbf{u} \in I'$*), when*

$$(a_{\mathbf{u}}, b_{\mathbf{u}}) = \underset{(a,b)}{arg\min} \left\{ \mathbb{E}\left[(Y - aY_{\mathbf{u}} - b)^2 \right] + \mathbb{E}\left[(Y_{\mathbf{u}} - aY - b)^2 \right] \right\} \tag{8.4}$$

then

$$a_{\mathbf{u}} = S_{\mathbf{u}}$$

Proof is in the appendix B.3.1.

The lemma 8.1 shows the best coefficient to predict the total variance from the predictor $Y_{\mathbf{u}}$ is the Sobol index $S_{\mathbf{u}}$. The coefficient $a_{\mathbf{u}}$ in the linear model of the lemma is equal to the Sobol index even if $\mathbb{E}[Y] \neq 0$. The $b_{\mathbf{u}}$ coefficient is not interesting to interpret the variance. Hence, **in the following, we will suppose that the output Y is centred**: $\mathbb{E}[Y] = 0$. The variance of the output can be explained with the linear model (8.5).

$$Y = a_{\mathbf{u}} Y_{\mathbf{u}} + \epsilon_{\mathbf{u}} \tag{8.5}$$

The statistical model (8.5) answers the following question: if the output Y has to be explained with $Y_{\mathbf{u}}$ (the same code with the input parameters $X_{\mathbf{u}}$ frozen), how much variance is it possible to explain? The best approximation of Y as a function of $Y_{\mathbf{u}}$ is $\mathbb{E}[Y|Y_{\mathbf{u}}]$ (it can be seen as the projection of Y onto $Y_{\mathbf{u}}$). The assumption stated by the linear model is that $\mathbb{E}[Y|Y_{\mathbf{u}}] = a_{\mathbf{u}}Y_{\mathbf{u}}$. As a consequence, the error ϵ of such model is a centred and the coefficient $a_{\mathbf{u}}$ is chosen to minimize its variance. The variance of $\epsilon_{\mathbf{u}}$ when it is minimum is denoted $\sigma^2_{0\mathbf{u}}$.

The statistical model (8.5) can be used to estimate only the Sobol index $S_{\mathbf{u}}$. The aim is to get all Sobol indices at once. The vector of all slopes is denoted $\mathbf{a} = (a_{\mathbf{u}}, \mathbf{u} \in I')$. The vector of all Sobol indices is denoted $\mathbf{S} = (S_{\mathbf{u}}, \mathbf{u} \in I')$. These vectors are of dimension $d = |I'|$. Finally, the minimization problem (8.4) is changed into (8.6).

$$\mathbf{S}^{ls} = \arg\min_{\mathbf{a}} \left\{ \sum_{\mathbf{u}\in I'} \mathbb{E}\left[(Y - a_{\mathbf{u}}Y_{\mathbf{u}})^2\right] + \mathbb{E}\left[(Y_{\mathbf{u}} - a_{\mathbf{u}}Y)^2\right] \right\} \tag{8.6}$$

The minimization of (8.6) benefits then from the advances in linear regression. In particular, we will focus on two penalties to select the most relevant coefficients: the L^1 penalty (problem 8.7, Lasso method) and the L^0 penalty (problem 8.8, best subset method).

$$\mathbf{S}^{l1} = \arg\min_{\mathbf{a}} \left\{ \sum_{\mathbf{u}\in I'} \mathbb{E}\left[(Y - a_{\mathbf{u}}Y_{\mathbf{u}})^2\right] + \mathbb{E}\left[(Y_{\mathbf{u}} - a_{\mathbf{u}}Y)^2\right] + \lambda\|\mathbf{a}\|_1 \right\} \tag{8.7}$$

with the L^1 norm $\|\mathbf{a}\|_1 = \sum_{\mathbf{u}\in I'} |a_{\mathbf{u}}|$.

$$\mathbf{S}^{l0} = \arg\min_{\mathbf{a}} \left\{ \sum_{\mathbf{u}\in I'} \mathbb{E}\left[(Y - a_{\mathbf{u}}Y_{\mathbf{u}})^2\right] + \mathbb{E}\left[(Y_{\mathbf{u}} - a_{\mathbf{u}}Y)^2\right] + \lambda\|\mathbf{a}\|_0 \right\} \tag{8.8}$$

with the L^0 norm $\|\mathbf{a}\|_0 = \sum_{\mathbf{u}\in I'} \mathbf{1}_{a_{\mathbf{u}}\neq 0} = |\{\mathbf{u},\, a_{\mathbf{u}} \neq 0\}|$.

8.2 Properties and links among estimators

The theoretical Sobol indices from penalized regression have been defined: least squares (8.6), Lasso (8.7), best subset (8.8). Now we focus on their estimation. The problem is to find an estimation of the coefficients in the linear model (8.5) from the following data:

$$Y = \begin{pmatrix} y^1 \\ \vdots \\ y^N \end{pmatrix} \qquad Y_{\mathbf{u}} = \begin{pmatrix} y^1_{\mathbf{u}} \\ \vdots \\ y^N_{\mathbf{u}} \end{pmatrix}$$

for all $\mathbf{u} \in I'$. The samples Y and $Y_{\mathbf{u}}$ are denoted the same way as the random variables Y and $Y_{\mathbf{u}}$ they are sampling.

Moreover, the random variables Y and $Y_{\mathbf{u}}$ are assumed centred: $\mathbb{E}[Y] = \mathbb{E}[Y_{\mathbf{u}}] = 0$ which has for consequence $\sum_{i=1}^{N} y^i = \sum_{i=1}^{N} y^i_{\mathbf{u}} = 0$. Since the random variables Y and $Y_{\mathbf{u}}$ follow the same law, we have $\mathbb{E}[Y^2] = \mathbb{E}[Y_{\mathbf{u}}^2] = \sigma^2$ and so $\sum_{i=1}^{N}(y^i)^2 = \sum_{i=1}^{N}(y^i_{\mathbf{u}})^2 = \hat{\sigma}^2 \simeq \sigma^2$. The variance of the noise in the linear model (8.5) is denoted $\sigma^2_{0\mathbf{u}} = V(Y - a_{\mathbf{u}} Y_{\mathbf{u}})$.

8.2.1 The least squares estimator

The coefficients of the linear model (8.5) can be estimated with ordinary least squares. The problem to solve in theory (8.6). In estimation, it is (8.9):

$$\widehat{\mathbf{S}}^{ls} = \arg\min_{\mathbf{a}} \left\{ \sum_{\mathbf{u}\in I'} \|Y - a_{\mathbf{u}} Y_{\mathbf{u}}\|_2^2 + \|Y_{\mathbf{u}} - a_{\mathbf{u}} Y\|_2^2 \right\} \tag{8.9}$$

For any $\mathbf{u} \in I'$, the solution is given by (8.10):

$$\widehat{S}^{ls}_{\mathbf{u}} = (Y_{\mathbf{u}}^T Y_{\mathbf{u}})^{-1} Y_{\mathbf{u}}^T Y = \frac{\sum_{i=1}^{N} y^i y^i_{\mathbf{u}}}{\sum_{i=1}^{N} (y^i_{\mathbf{u}})^2} \tag{8.10}$$

The least squares estimator is known to be the BLUE: Best Linear Unbiased Estimator. Its bias is zero and its variance is minimum among all estimators of the form AY with $A \in \mathbb{R}^{d\times N}$ (Gauss-Markov theorem, (Saporta, 2006) section 17.2).

$$\mathbb{E}\left[\widehat{S}^{ls}_{\mathbf{u}}\right] = S_{\mathbf{u}} \tag{8.11}$$

$$V\left(\widehat{S}^{ls}_{\mathbf{u}} | Y_{\mathbf{u}}\right) = (Y_{\mathbf{u}}^T Y_{\mathbf{u}})^{-1} \sigma_0^2 = \frac{\sigma_0^2}{\sigma^2} \tag{8.12}$$

Its expression is the same as the Monte Carlo estimator $\widehat{D_{\mathbf{u}}}^{MC1}$. Janon et al. (2014) already pointed out the equality of such estimators (remark 1.3 of the paper) and they proved their asymptotic normality (proposition 2.2 of the paper). Moreover, the asymptotic variance of the estimator given by Janon (denoted σ^2_{MC1}) and the variance given by Gauss-Markov theorem (equation (8.12)) are the same. Indeed, when Y is centred, σ^2_{MC1} is written

$$\begin{aligned}
\sigma^2_{MC1} &= \frac{V(Y(Y_{\mathbf{u}} - S_{\mathbf{u}} Y))}{V(Y)^2} && \text{From Janon et al. (2014)} \\
&= \frac{V(Y_{\mathbf{u}} - S_{\mathbf{u}} Y)}{V(Y)} && \text{because } \mathbb{E}[Y_{\mathbf{u}} - S_{\mathbf{u}} Y] = \mathbb{E}[Y] = 0 \text{ and } (Y_{\mathbf{u}}, Y) = (Y, Y_{\mathbf{u}}) \\
&= \frac{\sigma_0^2}{\sigma^2}
\end{aligned}$$

In conclusion, the least squares method gives the same estimator as crude Monte Carlo. This estimator is unbiased but asymptotically Gaussian: if the samples Y and $Y_{\mathbf{u}}$ are too small or for small Sobol indices, the variance is high and negative estimations are possible.

8.2.2 Lasso versus least square

The Lasso method (Least Absolute Shrinkage and Selection Operator) aims to solve a least squares problem while pushing some coefficients to be exactly 0. It is thus a valuable tool when the vector of coefficient is sparse (van de Geer, 2016). Having several coefficients equal to zero makes the statistical model more informative and more easily interpreted: predictors with a coefficient to 0 are dismissed. The Lasso method was first introduced by Tibshirani (1996). The principle is to solve the least square problem with a L^1-penalty in the function to minimize.

The use of a L^1-penalty is what make some coefficients to be exactly zero. Indeed, the admissible area has a very different shape with the L^1 norm and the L^2 norm. This is visible in the figure 8.1: an illustration with $p = 2$ is presented. On the right hand side, the admissible area for the L^2 norm is a circle. On the left hand side, the admissible area for the L^1 norm is a square with angle on the axis. The solution of the unconstrained problem (8.6) is denoted $\widehat{\beta}^{LS}$ in the figure 8.1 and the lines of equal cost are drawn in red. The solution of the constrained problem is the point in the admissible area the closest to $\widehat{\beta}^{LS}$ according to the cost lines in red. For the L^2 norm (right panel), the solution is on a circle. Both coefficients are likely to be non-zero. For the L^1 norm (left panel), the solution is likely on one of the corner of the square. On any corner, one the coefficient will be exactly zero.

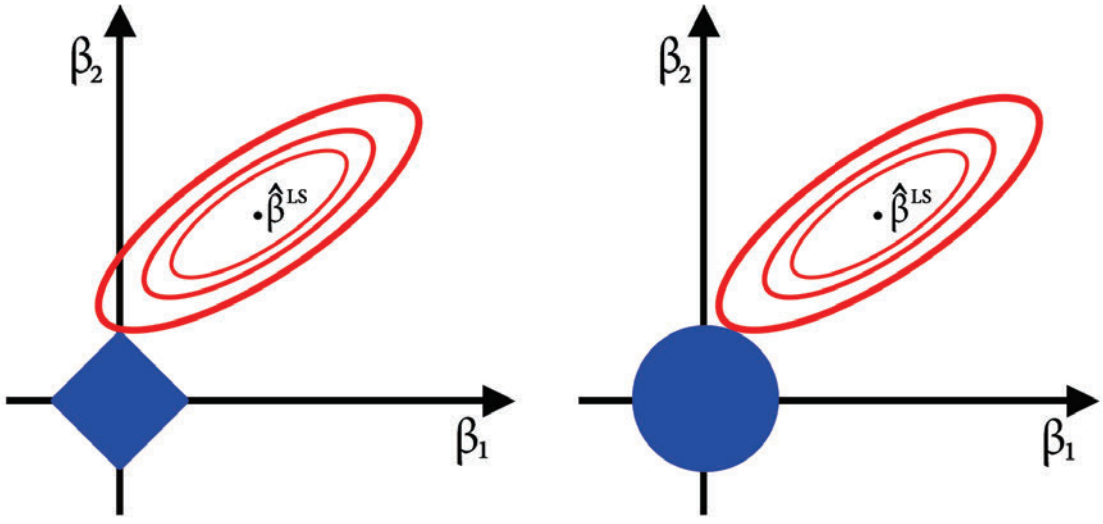


Figure 8.1 – Illustration of admissible area with L^1 and L^2 norms. The extreme points are located on an axis for the L^1 norm, thus one of the coefficient is null.
Credit: Par LaBaguette — Travail personnel, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=48816401

The Lasso is written only for the Sobol indices estimation problem. The problem to solve

is (8.13) and the solution is given by the proposition (8.1).

$$\widehat{\mathbf{S}}^{l1} = \arg\min_{\mathbf{a}} \left\{ \sum_{\mathbf{u}\in I'} \|Y - a_{\mathbf{u}} Y_{\mathbf{u}}\|_2^2 + \|Y_{\mathbf{u}} - a_{\mathbf{u}} Y\|_2^2 + \lambda_1 \|\mathbf{a}\|_1 \right\} \tag{8.13}$$

Proposition 8.1. *For any* $\mathbf{u} \in I'$*, the Lasso and least squares estimators are related according to the following formula:*

$$\widehat{S}_{\mathbf{u}}^{l1} = \max\left(\widehat{S}_{\mathbf{u}}^{ls} - \varepsilon_1, 0\right)$$

with $\varepsilon_1 = \frac{\lambda_1}{2\sigma^2}$.

Proof is in the appendix B.3.2, page 296.

The Lasso method gives an estimator that has a direct relationship with the least squares estimator. To get the Lasso estimator, the least square estimator is shrunk of $\varepsilon_1 = \lambda_1/(2\sigma^2)$. When the least squares estimator is smaller that the shrunk, the Lasso estimator is exactly zero.

From a Bayesian point of view, the penalty is equivalent to give a prior distribution to the coefficients $\mathbf{a}$. The L^1-penalty imposes an absolute exponential prior distribution: $\forall a \in \mathbf{a}$, $\mathbb{P}(a) \propto \exp(-\lambda_1 |a|)$. Maximizing the likelihood gives the least squares estimator. Maximizing the posterior probability with an absolute exponential prior gives the Lasso estimator.

The hard part to take profit of the Lasso is to correctly set the penalty λ_1. When $\lambda_1 \to 0$, the estimator tends to the ordinary least squares estimator and there is no benefit to use the Lasso. When $\lambda_1 \to +\infty$, the penalty unrealistically shrinks the coefficients to estimate. Good predictors will be dismissed, leading to too simple models.

8.2.3 Best subset versus least square

The so-called best subset method (mentioned in (Tibshirani, 1996; Breiman, 1995; Lin et al., 2010)) adds a L^0 penalty to the least square problem (equation (8.8)):

$$\widehat{\mathbf{S}}^{l0} = \arg\min_{\mathbf{a}} \left\{ \sum_{\mathbf{u}\in I'} \|Y - a_{\mathbf{u}} Y_{\mathbf{u}}\|_2^2 + \|Y_{\mathbf{u}} - a_{\mathbf{u}} Y\|_2^2 + \lambda_0 \|\mathbf{a}\|_0 \right\}$$

with the L^0 norm $\|\mathbf{a}\|_0 = |\{i,\, a_i \neq 0\}|$ denoting the number of non-zero components in $\mathbf{a}$.

Proposition 8.2. *For any* $\mathbf{u} \in I'$*, the best subset and least squares estimators are related according to the following formula:*

$$\widehat{S}^{l0}_{\mathbf{u}} = \widehat{S}^{ls}_{\mathbf{u}} \mathbf{1}_{\widehat{S}^{ls}_{\mathbf{u}} > \varepsilon_0}$$

with $\varepsilon_0 = \sqrt{\frac{\lambda_0}{\sigma^2}}$.

Proof is in the appendix B.3.3, page 298.

The best subset estimator gives also an estimator directly linked with the least square estimator. When the least square estimator is smaller than the threshold $\varepsilon_0 = \sqrt{\lambda_0}/\sigma$, the best subset estimator is exactly zero. Otherwise, least square and best subset estimators are equal. The shrinkage is not systematic, conversely to the Lasso. As for the Lasso, the difficulty is to correctly choose the penalty λ_0.

The best subset estimator is optimal in terms of information loss. Indeed, it minimizes the Akaike information criterion. The Akaike information criterion (AIC) is a metric of the information loss due to the model (original publication in 1973, republished in the collection (Akaike, 1998)). The model $Y = a_{\mathbf{u}} Y_{\mathbf{u}} + \epsilon_{\mathbf{u}}$ is imperfect and the AIC quantifies the benefit of adding a new predictor. It is defined by

$$AIC = 2k - 2\log(L(\mathbf{a}))$$

where k is the number of coefficients to estimate and L is the likelihood function. In our case, k is the number of non-zero Sobol indices and the likelihood function is given by

$$\log(L(\mathbf{a})) = \log(\mathbb{P}(Y|\mathbf{a})) \propto \sum_{\mathbf{u} \in I} \|Y - a_{\mathbf{u}} Y_{\mathbf{u}}\|_2^2$$

Hence, the AIC for our problem is exactly the function of $\mathbf{a}$ to minimize in the best subset problem (8.8).

Unfortunately, the minimization with a L^0 penalty is a NP-hard problem (Natarajan, 1995). As a consequence, only greedy algorithms can perform the minimisation of (8.8). For example, in (Tibshirani, 1996), they are estimated using the so-called *leaps and bounds* procedure (Furnival and Wilson, 1974).

8.2.4 Lasso versus best subset

Lasso and best subset estimators are both related to the least squares estimator. Transitively, they are related to each other. The property 8.1 tells the Lasso is *soft threshold* of the least square estimator, while property 8.2 tells the best subset is a *hard threshold* of the least square estimator. The threshold of Lasso is said soft because it is continuous, while the hard threshold is not. Shapes of both threshold functions are displayed in figures (8.2) and (8.3).

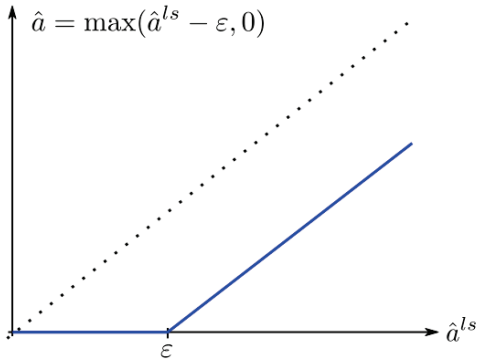


Figure 8.2 – Soft threshold: link between the Lasso estimator and the least square estimator.

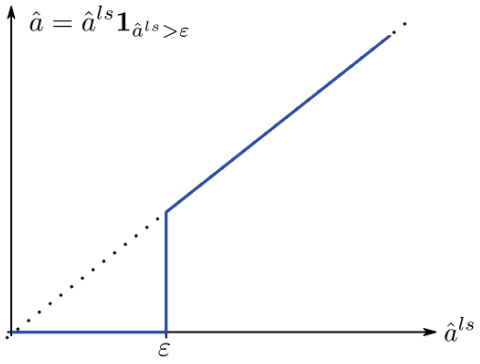


Figure 8.3 – Hard threshold: link between the best subset estimator and the least square estimator.

To have the same threshold $\varepsilon = \varepsilon_1 = \varepsilon_0$, we need to have the following relation between the penalty:

$$\lambda_1 = 2\sigma\sqrt{\lambda_0} \tag{8.14}$$

In this case, the relationship between $\widehat{S}^{l0}_{\mathbf{u}}$ and $\widehat{S}^{l1}_{\mathbf{u}}$ is easy to write and valid everywhere but at the discontinuity (when $\widehat{S}^{ls}_{\mathbf{u}} = \varepsilon$).

$$\widehat{S}^{l0}_{\mathbf{u}} = \frac{\widehat{S}^{ls}_{\mathbf{u}}}{\widehat{S}^{ls}_{\mathbf{u}} - \varepsilon}\widehat{S}^{l1}_{\mathbf{u}} \tag{8.15}$$

When the thresholds are different, one need to distinguish the case $\varepsilon_1 < \varepsilon_0$ and $\varepsilon_1 > \varepsilon_0$ and then the sub-cases $\widehat{S}^{ls}_{\mathbf{u}} < \min(\varepsilon_1, \varepsilon_0)$, $\widehat{S}^{ls}_{\mathbf{u}} \in]\varepsilon_1, \varepsilon_0[$ and $\widehat{S}^{ls}_{\mathbf{u}} > \max(\varepsilon_1, \varepsilon_0)$. The final result does not feed the comment. One can see that $\widehat{S}^{l0}_{\mathbf{u}} \to \widehat{S}^{l1}_{\mathbf{u}}$ either when $\varepsilon \to 0$ (both converge toward $\widehat{S}^{ls}_{\mathbf{u}}$ when the penalty decreases) either when $\widehat{S}^{ls}_{\mathbf{u}} \to +\infty$. It highlights that the use of L^1 or L^0 penalty is only relevant for small Sobol indices. For large Sobol indices, $\widehat{S}^{ls}_{\mathbf{u}}$ has the advantage to be unbiased.

Even with an expression of $\widehat{S}^{l0}_{\mathbf{u}}$ and $\widehat{S}^{l1}_{\mathbf{u}}$ as a function of $\widehat{S}^{ls}_{\mathbf{u}}$, for which the asymptotic behaviour is known, the asymptotic law of the penalized estimators are not straightforward. The Delta method does not apply because neither soft nor hard threshold functions are differentiable.

Lin et al. (2010, 2008) argue in favour of the L^0 penalty. They compare L^1 and L^0 penalties according to the predictive risk function $R(\beta, \widehat{\beta}) = \mathbb{E}\left[\|X\beta - X\widehat{\beta}\|_2^2\right]$ and show that the risk ratio of L^0 over L^1 is bounded while the risk ratio of L^1 over L^0 is not. The final estimate of L^0 is better, but the algorithm to get it are not efficient. Indeed, Natarajan (1995) proved that the L^0-penalized least squares is a NP-hard problem. Hence, even if the L^0 estimate is better, more efficient algorithms exist for the L^1 estimate.

8.3 Choice of penalty by cross-validation

8.3.1 General principle of cross-validation

Given a linear model $Y = X\beta + \epsilon$, one wants to estimate β and the uncertainty on the estimation. The principle of cross-validation is to split the sample X in two part: a part dedicated to the estimation of the coefficient in the regression, another part dedicated to the prediction of the model with the estimated sample. On the part dedicated to the prediction, one has both the reference value (given in the sample) and the predicted value. Hence, one can have a score of error on the model.

The sample (Y, X) where $Y \in \mathbb{R}^N$ and $X \in \mathbb{R}^{N \times d}$ is divided into one sample (Y_A, A) used for the estimation of the coefficient (the *training* sample), and a sample (Y_B, B) used for the prediction and error estimation (the *testing* sample). We denote N_A the size of the learning sample.

$$Y = \begin{pmatrix} y^1 \\ \vdots \\ y^{N_A} \\ y^{N_A+1} \\ \vdots \\ y^N \end{pmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} Y_A \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} Y_B \end{matrix}$$

$$X = \begin{pmatrix} X_1^1 & \dots & X_d^1 \\ \vdots & & \vdots \\ X^{N_A} & \dots & X_d^{N_A} \\ X^{N_A+1} & \dots & X_d^{N_A+1} \\ \vdots & & \vdots \\ X^N & \dots & X_d^N \end{pmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} A \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} B \end{matrix}$$

The training phase provides the estimated coefficients $\widehat{\beta}$ in the linear model $Y = \beta X + \epsilon$. The regression (with or without penalty) is performed on the sample (Y_A, A). For instance, for the ordinary least squares, we have the formula

$$\widehat{\beta} = (A^T A)^{-1} A^T Y_A$$

Using this estimated coefficient, the output is predicted with the predictors of the second sample. The comparison with the observed output provides an estimation of the uncertainty of prediction:

$$\epsilon \simeq Y_B - B\widehat{\beta}$$

In particular, one can check the bias, variance and mean squared error. In our case, we are interested in the uncertainty of estimation. The uncertainty of estimation has to be derived from the uncertainty of prediction.

8.3.2 Uncertainty of prediction and uncertainty of estimation

The uncertainty on prediction quantifies the error made by the model when it is applied on new data. The uncertainty on estimation quantifies the error made in the estimation by comparing to a new dataset. By splitting the sample into a training sample and a testing sample, one can access the uncertainty of prediction. But the uncertainty of estimation is not directly accessible and has to be derived from the uncertainty of prediction. If we consider a linear model $Y = X\beta + \epsilon$, with $Y \in \mathbb{R}$, $X \in \mathbb{R}^d$, $\beta \in \mathbb{R}^d$ and we assume we have an estimation $\widehat{\beta}$ of the coefficient which is independent of X.

The uncertainty of the prediction $\widehat{Y} = X\widehat{\beta}$ of the variable Y is described by its bias, variance, and mean-squared error (MSE).

$$\mathsf{Bi}_Y = \mathbb{E}\left[\widehat{Y} - Y\right] \tag{8.16}$$

$$\mathsf{Var}_Y = V\left(\widehat{Y}\right) = \mathbb{E}\left[(\widehat{Y} - \mathbb{E}\left[\widehat{Y}\right])^2\right] \tag{8.17}$$

$$\mathsf{MSE}_Y = \mathbb{E}\left[(\widehat{Y} - Y)^2\right] \tag{8.18}$$

These statistics describe the uncertainty of prediction. But for this application, we are more interested on the error of estimation: bias, variance and MSE for the estimated coefficients β.

$$\mathsf{Bi}_\beta = \mathbb{E}\left[\widehat{\beta}\right] - \beta \tag{8.19}$$

$$\mathsf{Var}_\beta = V\left(\widehat{\beta}\right) = \mathbb{E}\left[(\widehat{\beta} - \mathbb{E}\left[\widehat{\beta}\right])(\widehat{\beta} - \mathbb{E}\left[\widehat{\beta}\right])^T\right] \tag{8.20}$$

$$\mathsf{MSE}_\beta = \mathbb{E}\left[(\widehat{\beta} - \beta)^T(\widehat{\beta} - \beta)\right] \tag{8.21}$$

Note that they all are of different dimensions: $\mathsf{Bi}_\beta \in \mathbb{R}^d$, $\mathsf{Var}_\beta \in \mathbb{R}^{d\times d}$ and $\mathsf{MSE}_\beta \in \mathbb{R}$. Although, they are linked by the following relation:

$$\mathsf{MSE}_\beta = \mathsf{Bi}_\beta^T \mathsf{Bi}_\beta + \mathrm{tr}(\mathsf{Var}_\beta) \tag{8.22}$$

where $\mathrm{tr}(\cdot)$ is the trace operator.

The uncertainty of prediction and estimation are linked with the following relations when $\widehat{\beta}$ is independent from X:

$$\mathsf{Bi}_Y = \mathbb{E}\left[X\right] \mathsf{Bi}_\beta \tag{8.23}$$

$$\mathsf{Var}_Y = \mathbb{E}\left[X\mathsf{Var}_\beta X^T\right] + V\left(X\mathbb{E}\left[\widehat{\beta}\right]\right) \tag{8.24}$$

$$\mathsf{MSE}_Y = \mathsf{Bi}_\beta^T \mathbb{E}\left[X^T X\right] \mathsf{Bi}_\beta + \mathbb{E}\left[\mathrm{tr}(X\mathsf{Var}_\beta X^T)\right] + \sigma_0^2 \tag{8.25}$$

Proof is in the appendix B.3.4.

In the case of the Lasso method, the cross-validation is repeated for different values of penalty. As the estimator given by the ordinary least squares is unbiased, the bias of esti-

mation should decrease with the penalty. Conversely, a large penalty shrinks the coefficients and thus reduces the variance of the estimator but its bias grows. In the middle, we expect the mean squared error to reach a minimum.

$$\lim_{\lambda \to 0} \mathsf{Bi}_\beta(\lambda) = 0 \quad \text{and} \quad \lim_{\lambda \to +\infty} \mathsf{Var}_\beta(\lambda) = 0$$

With the testing sample, one can estimate the bias, variance and mean squared error of prediction (equations (8.16),(8.17),(8.18)). They can be linked to the same statistics for estimation (equations (8.19),(8.20),(8.21)) through the formulae (8.23), (8.24) and (8.25). The penalty is then chosen to minimise the error of prediction $\mathsf{MSE}_Y(\lambda)$. When this minimum is reached, the chosen penalty makes a good comprise between bias and variance of estimation.

8.3.3 Application to Sobol indices estimation

For our particular problem, the linear model is not of the form $Y = \mathbf{X}\beta + \epsilon$. Instead, we have d linear models of the form $Y = a_{\mathbf{u}} Y_{\mathbf{u}} + \epsilon_{\mathbf{u}}$. There is only one predictor which verifies $\mathbb{E}[Y_{\mathbf{u}}] = 0$ and $\mathbb{E}[Y_{\mathbf{u}}^2] = \sigma^2$, for any $\mathbf{u}$. The error $\epsilon_{\mathbf{u}}$ has no reason to be the same for each model: $\mathbb{E}[\epsilon_{\mathbf{u}}] = 0$ and $\mathbb{E}[\epsilon_{\mathbf{u}}^2] = \sigma_{0\mathbf{u}}^2$. Applying the formulae (8.23), (8.24) and (8.25) we have:

$$\mathsf{Bi}_Y^{\mathbf{u}} = \overbrace{\mathbb{E}[Y_{\mathbf{u}}]}^{=0} \mathsf{Bi}_a^{\mathbf{u}} = 0 \tag{8.26}$$

$$\mathsf{Var}_Y^{\mathbf{u}} = \sigma^2 (\mathsf{Var}_a^{\mathbf{u}} + \mathbb{E}[\widehat{a_{\mathbf{u}}}]^2) \tag{8.27}$$

$$\mathsf{MSE}_Y^{\mathbf{u}} = \sigma^2 (\mathsf{Bi}_a^{\mathbf{u}})^2 + \sigma^2 \mathsf{Var}_a^{\mathbf{u}} + \sigma_{0\mathbf{u}}^2 = \sigma^2 \mathsf{MSE}_a^{\mathbf{u}} + \sigma_{0\mathbf{u}}^2 (1 - \sigma^2) \tag{8.28}$$

In practice, the estimators of bias, variance and mean squared error of prediction are accessible through the formulae:

$$\widehat{\mathsf{Bi}_Y^{\mathbf{u}}} = \frac{1}{N} \sum_{i=1}^{N} (y_{\mathbf{u}}^i \widehat{a_{\mathbf{u}}} - y^i) \tag{8.29}$$

$$\widehat{\mathsf{Var}_Y^{\mathbf{u}}} = \frac{1}{N-1} \sum_{i=1}^{N} \left(y_{\mathbf{u}}^i \widehat{a_{\mathbf{u}}} - \frac{1}{N} \sum_{i=1}^{N} y_{\mathbf{u}}^i \widehat{a_{\mathbf{u}}} \right)^2 \tag{8.30}$$

$$\widehat{\mathsf{MSE}_Y^{\mathbf{u}}} = \frac{1}{N} \sum_{i=1}^{N} (y^i - y_{\mathbf{u}}^i \widehat{a_{\mathbf{u}}})^2 \tag{8.31}$$

On the two last series of equation, one can see that the estimator for the bias is useless. Indeed, despite the link (8.23) between bias of prediction and bias of estimation, despite the fact that $\mathsf{Bi}_a(\lambda)$ is expected to increase with λ, the estimator (8.29) cannot be used to retrieve that trend because of the relation (8.26).

For the variance and mean-squared error, we would rather have a score for all the coefficients, not one for each $\mathbf{u}$. Applying the definition of the MSE for any vector of parameters β (equation 8.21) to the case $\beta = \mathbf{a}$ and $\beta = (a_{\mathbf{u}} Y_{\mathbf{u}} + \epsilon_{\mathbf{u}}, \mathbf{u} \in I')$, the relevant global MSE is the sum of MSE for each $\mathbf{u}$.

$$\mathsf{MSE}_Y = \sum_{\mathbf{u} \in I'} \mathsf{MSE}_Y^{\mathbf{u}} \quad \text{and} \quad \mathsf{MSE}_a = \sum_{\mathbf{u} \in I'} \mathsf{MSE}_a^{\mathbf{u}}$$

$$\mathrm{tr}(\mathsf{Var}_Y) = \sum_{\mathbf{u} \in I'} \mathsf{Var}_Y^{\mathbf{u}} \quad \text{and} \quad \mathrm{tr}(\mathsf{Var}_a) = \sum_{\mathbf{u} \in I'} \mathsf{Var}_a^{\mathbf{u}}$$

Finally,

$$\mathrm{tr}(\mathsf{Var}_Y) = \sigma^2 \left(\mathrm{tr}(\mathsf{Var}_a) + \sum_{\mathbf{u} \in I'} \mathbb{E}\left[\widehat{a_{\mathbf{u}}}\right]^2 \right)$$

$$\mathsf{MSE}_Y = \sigma^2 \mathsf{MSE}_a + (1 - \sigma^2) \sum_{\mathbf{u} \in I'} \sigma_{0\mathbf{u}}^2$$

In the equation of variance, the terms $\mathbb{E}\left[\widehat{a_{\mathbf{u}}}\right]^2$ vary with penalty (they are linked to the bias which is not accessible). The variations of $\mathrm{tr}(\mathsf{Var}_Y)$ against λ are thus not equal to the variations of $\mathrm{tr}(\mathsf{Var}_a)$. In the equation of MSE, $\sigma^2 = V(Y)$ is a constant, $\sigma_{0\mathbf{u}}^2 = V(\epsilon_{\mathbf{u}})$ is the minimum variance of the least squares problem. It is also a constant (it depends only on $\mathbf{u}$). The variations of MSE_Y are the same as MSE_a. Hence, the error of estimation is minimum when the error of prediction is minimum.

In conclusion, the penalty will be chosen to minimize the error of prediction. In the particular case of centred design, the bias of prediction is longer linked to the bias of estimation. The variance of prediction rely on additional terms that are still to be estimated to complete the link between prediction and estimation. As a consequence, only mean-squared error plot will be used.

8.3.4 Results of numerical experiments

The three estimators presented in the last section will be experimented on the application case of turbulence reconstruction. The sample of inputs X and Z are generated with latin hypercubes. For both X and Z, 4000 values of each input are generated. The split between samples for cross-validation have been made randomly, with a proportion of 65% for the training sample (2400 values) and 35% for the test sample (1600 values). Once the sample has been split, the random state of the splitting is saved in case the experiment is repeated in the same conditions. The experiment have been carried out with the meta-model of the output r_V (the root-mean squared error on the wind) because it is easy to interpret. The algorithm used to minimize the cost functions (least squares and Lasso) is the conjugated gradient.

The figures (8.4) and (8.5) show the evolution of the estimated mean-squared error $\widehat{\mathsf{MSE}_Y}$ of the Lasso for different penalty values. On the left (figure 8.4) the Lasso estimate have

calculated from the Monte-Carlo estimate, on which the soft-threshold function has been applied, using the proposition 8.1. On the right (figure 8.5) the Lasso estimate have calculated by the minimization of the cost function defined by (8.7) (L^1 penalized least squares). One can see a clear minimum on both. Although, the slope is much more regular when the soft threshold is applied. Indeed, it avoids the weaknesses of the minimization: do not reach the exact minimum but something close enough, get stuck at a local minimum... Overall, the comparison of both figures corroborates the minimization is trustworthy. But in the objective to apply another minimum finder to such curve, the soft threshold one is recommended. Moreover, the soft threshold one is much faster to compute.

As a conclusion, to find the good penalty value by cross-validation, we recommend to do first an ordinary least squares regression on the training sample, then to get the Lasso estimate by soft thresholding for all the penalty values and eventually to compute the mean squared error on the testing sample. This recommendation holds only for problems which have a result similar to the proposition 8.1.

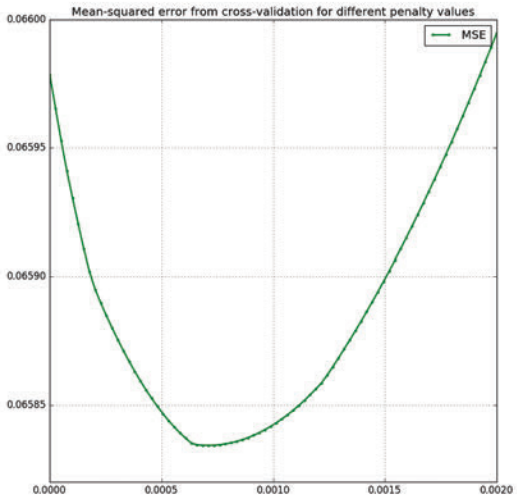


Figure 8.4 – Estimated mean squared error against the penalty in Lasso estimation. Obtained with soft thresholding of the Monte Carlo estimate.

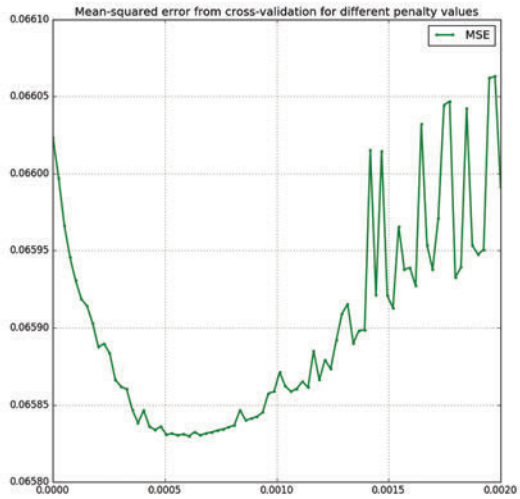


Figure 8.5 – Estimated mean squared error against the penalty in Lasso estimation. Obtained with the minimization of the cost function.

This methodology can be extended to the L^0 penalty. The best subset estimator resulting from L^0 penalized regression has been shown to be accessible through a hard thresholding of the least squares estimator (proposition 8.2). This was the only method of estimation tested here since only greedy algorithm can solve the L^0 penalized minimization. As a consequence, the parameter to check will not be the penalty λ_0 but the threshold $\varepsilon_0 = \sqrt{\frac{\lambda_0}{\sigma^2}}$. The Monte Carlo estimate is calculated on the training sample. For each threshold value, the mean squared error is estimated on the testing sample. It results the curve in figure 8.6.

One can see the step-like shape of the curve: the threshold does not have influence until it reaches the next coefficient value. As a consequence, the minimum is not unique. For this example, it is reached for a threshold around 0.1, which a large value. It let only two non-zero indices. It points out the method of selection is not perfect and might be too selective. In (Fruth et al., 2011) the value of 0.02 is used to threshold second order Sobol indices (figure 2 in the 2011 version on HAL). This value will be used here as well.

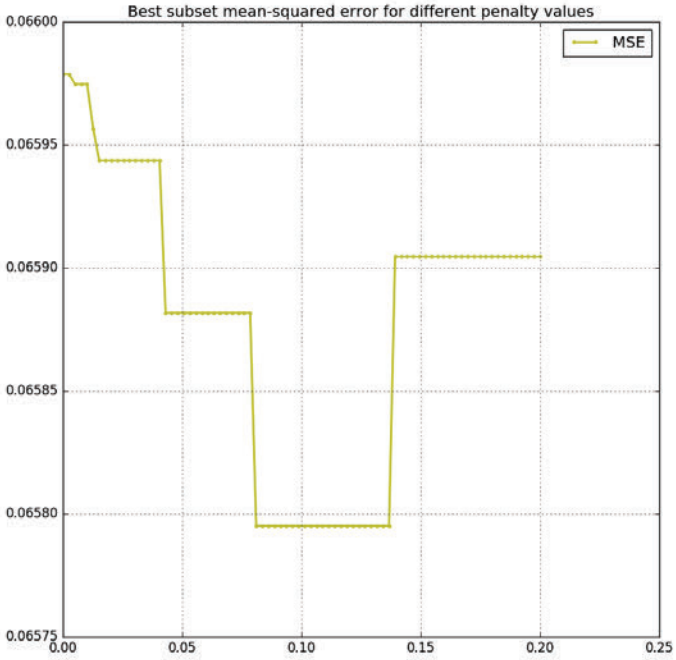


Figure 8.6 – Estimated mean squared error against the threshold in best subset estimation.

The figure 8.7 shows the evolution of the coefficients with the penalty. On the x-axis is the value of the penalty in logarithmic scale. On the y-axis is the value of the estimated coefficients. One can see that for a penalty almost null, the coefficients are all non-zeros. Actually they are equal to their ordinary least squares estimate. For a very large penalty, all coefficients are null. When the penalty decreases, they raise one after another. The introduction of a new coefficient can sometimes influence the curve of another coefficient. This is the sign of a correlation between them (as in (Hesterberg et al., 2008), figure 2). No such feature is visible in the figure. This is not surprising since the inputs of the sensitivity analysis have been simulated independently. Although it is good to be confirmed by this observation.

Four estimators have been tested: the straightforward Monte Carlo, the ordinary least squares, the Lasso and the best subset. They are displayed in figure (8.8). As expected, the least squares (blue) and the Monte Carlo (black dashed line) are two realisations of the same

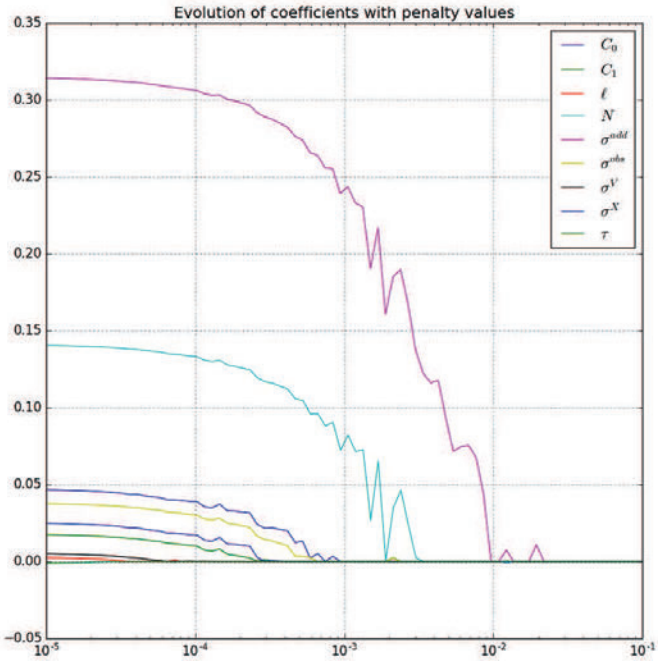


Figure 8.7 – Value of the coefficients estimated by Lasso regression against the penalty (in log-scale). The coefficients are all at 0 for large penalty and raise in order of importance up to their value as obtained with ordinary least square.

estimator. The Lasso (green) shrinks all the coefficients up to 0. It results that only the main indices are kept non-zero and the negative estimation of small indices are filtered. The best subset (yellow) also selects the major coefficients, but it does not shrink them. The hard threshold was set to the value $\varepsilon_0 = 0.02$.

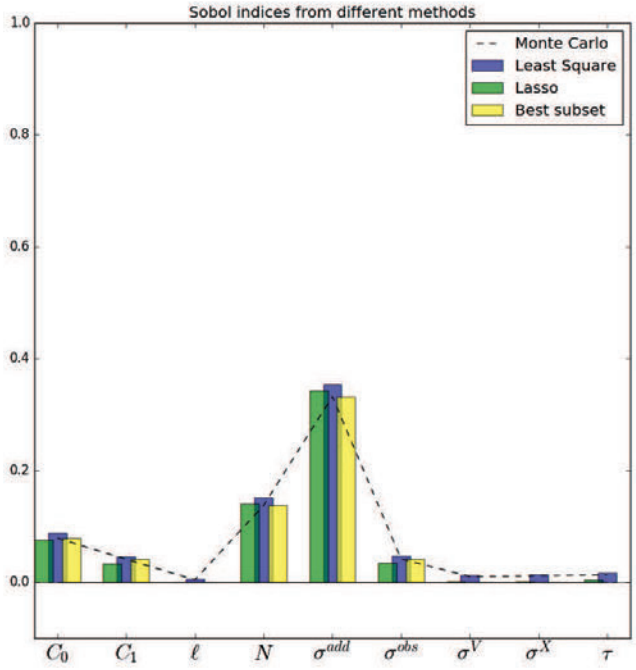


Figure 8.8 – Three estimators ensuing from regression (coloured bars) are compared to the Monte Carlo estimator (black dashed line). One can see that the ordinary least square (blue) gives the same estimation (more or less some randomness due to estimation). The best subset is exactly equal to the Monte Carlo, excepted for indices smaller than the threshold, which are 0. The Lasso estimator gives a shrunk estimation bounded to 0.

8.4 Conclusion

Sobol indices summarize the influence of a group of parameters with a real number in $[0, 1]$. But the number of groups grows exponentially with the number of parameters (if the code has p parameters, there are 2^p groups of parameters). In practice, only few of them are really influencing the code. Moreover, the interpretation of the Sobol indices will focus only on the main ones. That is to say, a good estimation of Sobol indices is not necessarily an unbiased estimation.

Penalized regressions offer biased estimators with lower variance such that the total error is lower. From the initial remark that Sobol indices can be seen as the estimated parameter in a linear model, three regression types have been tested. The ordinary least squares give the same estimate as with Monte Carlo. The L^1 penalized least squares give the Lasso estimate. The L^0 penalized least squares give the best subset estimate. The Lasso shrinks all coefficients and set the smallest ones to exactly zero. The best subset does not shrink the coefficients but set the smallest ones to exactly zero. They are linked to the least squares estimate with a soft threshold and a hard threshold, respectively.

To set the penalty objectively, a cross-validation has been carried out for each penalty value in a given set. The penalty which gives the lowest mean squared error is chosen to

perform the final estimation. The application of this methodology on a single example has shown good results, but more experimentations are needed to assess if it can be repeated and trusted.

General conclusion

The present thesis analyses the sensitivity of the reconstruction method to filter wind measurements. This is important to improve the ability of lidar to perform turbulence measurements (in wind farms or in airport, for example). The presentation was divided in two parts: a first part dedicated to the state of the art of the different fields used in the reconstruction method, a second part dedicated to the applications and the results. This conclusion summarizes the original contributions of this work.

Major contributions

Tuning strategy

The results of the sensitivity analysis presented at the chapter 6 underlined key inputs parameters and relevant output scores. The key parameters are the number of particles, N, the true observation noise, σ^{add} and the given observation noise, σ^{obs}. The relevant output are the wind spectrum slope, b, the wind RMSE, r_V and the execution time T_{exe}. Additional experiments of the chapter 7 show how the key inputs influence the relevant outputs. From this knowledge, we highlight a tuning strategy to set the most influential inputs. The tuning strategy is the following:

1. Set N to a low value, such that T_{exe} is really small.
2. For σ^{obs} ranging around the *a priori* accuracy of the instrument, calculate the wind spectrum slope b.
3. Set σ^{obs} to the value which gives b the closest to -5/3. σ^{obs} is then almost equal to σ^{add}.
4. Set N to the maximum affordable value. The error on wind retrieval is now minimum, estimated by $K\frac{\sigma^{obs}}{\sqrt{N}}$ with $K = 2.33$.

Thanks to the tuning strategy coming out of this work, the reconstruction method should be easier to deploy on new dataset. The benefit of the tuning strategy could be assessed on other dataset from BLLAST. Beyond, automatic application of the tuning strategy would help making the reconstruction more autonomous.

Reconstruction system details and behaviour

The reconstruction method was described in previous publications (Baehr, 2008, 2010; Baehr et al., 2011; Suzat et al., 2011; Rottner and Baehr, 2014; Rottner, 2015). Such publications

were highlighting the originality of the method in comparison to existing methods and giving theorems about convergence. Deeper details could not be published because of the patenting process, which came out to pending in December, 2016 (Baehr et al., 2016). Publications are now authorized and this thesis is probably the most complete. Another paper, Rottner and Baehr (2017), was submitted in June 2017, but cannot be as detailed as this document. The level of details given in the chapter 5 is an important result of this PhD.

In addition, the validation scores built for the sensitivity analysis have been studied and some theoretical results about their behaviour are demonstrated. The theorem 5.2 links the number of null potential to the number of particles. It is based on a result from Del Moral (2004), itself coming from Azuma-Hoeffding's inequality. The theorem 5.3 links the number of null potential to the observation noise (true and given). This is an orginal result in agreement with the numerical experiments of the chapter 7. However, the numerical experiment does not test the theory. Dedicated experiments shall be carried out before claiming the theoretical result is matched by the experiment.

Sobol indices estimation

The chapter 8 tests an innovative way of estimation of Sobol indices. The Sobol indices expressed as a solution of a least squares problem was already known (Janon et al., 2014). But it has never been combined with penalties, as it is done in optimization under constaints. The idea is that only the few highest Sobol indices are useful in pratice. To make small Sobol indices go to exactly zero, a penalty is added to the function to minimize in the least square problem. The tested penalties are L^1 (the sum of absolute value of Sobol indices) and L^0 (the number of non-zero Sobol indices). The L^1 penalty shrinks all coefficients and set the smallest ones to exactly zero. The L^0 penalty does not shrink the coefficients but set the smallest ones to exactly zero. To set the penalty objectively, a cross-validation has been carried out for each penalty value in a given set. The penalty which gives the lowest mean squared error is chosen to perform the final estimation. The application of this methodology on a single example has shown good results, but more experimentations are needed to assess if it can be repeated and trusted.

Improvements in the implementation

The need of many runs to perform a sensitivity analysis motived the recoding of the reconstruction code from Scilab to Fortran 90. Indeed, Scilab is a high level interpreted language, while Fortran is a low level compiled language. The coding with a low level language is much harder than high level. Software engineering took a large part of the time in this PhD. It was the opportunity to re-think the algorithms. The conditioning step of the reconstruction especially, was improved (economy of one loop) The algorithm was presented in the chapter 5. Overall, the computing time was reduced by a factor 150. The 1D code used here processes 2 hours of data in 70 seconds (for 1400 particles). The previous code was processing it in about 3 hours. This result must be tempered by the fact that the Scilab code performs 3D

reconstruction.

Prospects

Influence of the meta-model

It is a legitimate to wonder to what extend the presented results and conclusions are dependent from the choice of meta-model. Although a rigorous assessment (systematic cross-validation with different kernel types for example) was not possible within the allocated time, the whole sensitivity analysis was repeated with a Matérn 3/2 variogram instead of the Gaussian one. The results, in figures 8.9 to 8.12, are presented in the same way that figures 6.37 and 6.38: the tile has inputs in abscissa, outputs in ordinate and the Sobol indices of a given input for a given output is depicted in shade of color. First order simple Sobol indices (direct influence share) are in blue (figures 8.9 and 8.10). The interaction part (difference between total and simple first order indices) is in green (figures 8.11 and 8.12). The only visible difference is the line of the output N_{G0}: the interaction share is much smaller with the Matèrn variogram (as a consequence, direct effect is stronger). This difference is not clear to interpret and it does not contest the conclusions of this work. Therefore, this experiment is agreeing that the conclusions are not dependent from the choice of the meta-model.

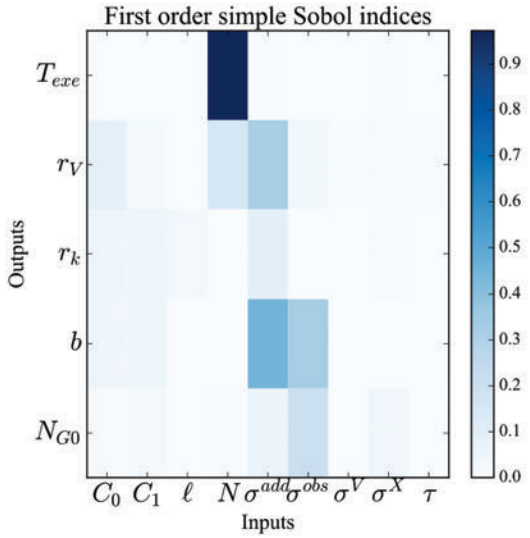


Figure 8.9 – First order Sobol indices with **Gaussian** variogram.

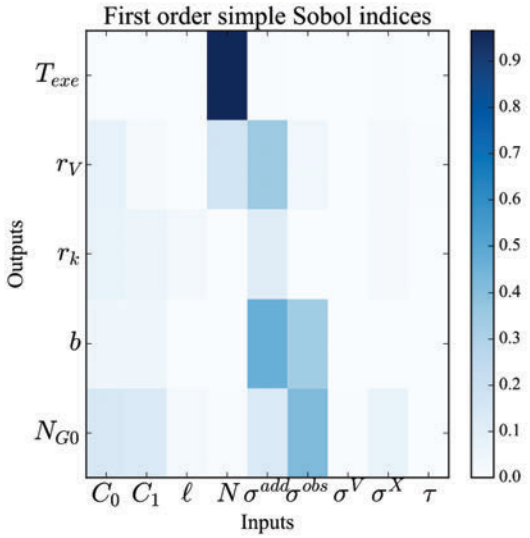


Figure 8.10 – First order Sobol indices with **Matérn 3/2** variogram.

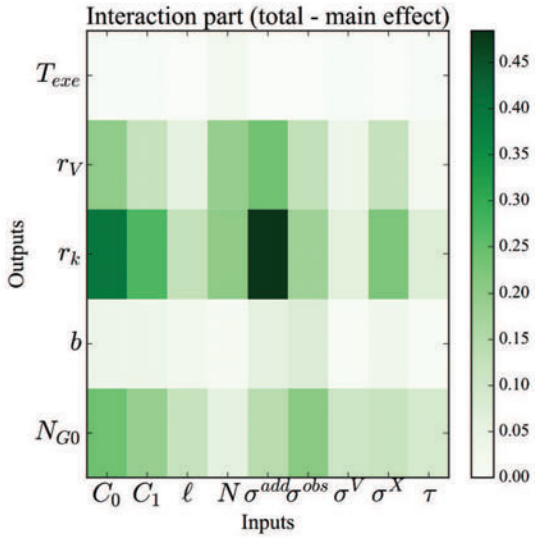


Figure 8.11 – Interaction part with **Gaussian** variogram.

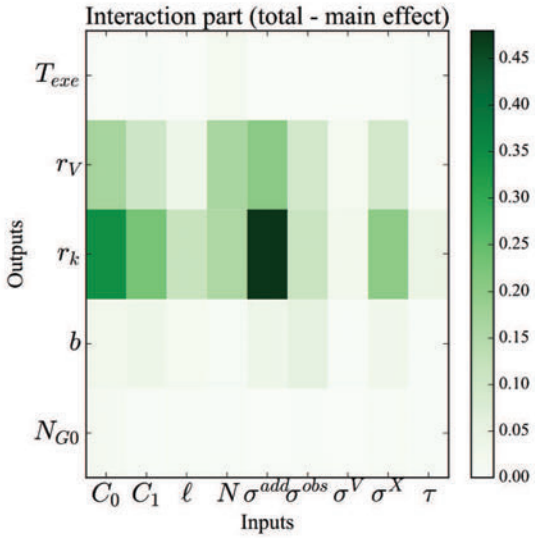


Figure 8.12 – Interaction part with **Matérn 3/2** variogram.

Estimation with penalised regression

The results presented in this thesis about the estimation of Sobol indices with penalised regression are only a demonstrative example. The method should be tested on a complete real case example, and compared to other well proven estimation methods. Mathematical properties (invariances, convergence...) of such estimators are still to be explored.

Sensitivity analysis feedback

The sensitivity analysis carried out here was the first ever done in the reconstruction. Strong of this first experiment, we can do recommendations for a second study.

The scores can be discussed. Although the initial idea of the score RMSE on TKE was worth to try, it appears to be not as informative as expected. The effects of the integration time, τ, are compensated in Sobol indices estimation. At the chapter 1, it was told that spatial and time variances cannot be compared for vertical velocity. The value of this score is not interpretable, nor are its trends. Instead, an average value of LSKTE would have been informative. The number of particles rejected at the selection is also a good indicator of malfunctions in the reconstruction. It could be a good score.

As a first attempt, the stochastic nature of the code was not taken into account (the argument is that N is always large enough to ensure convergence). In a second experiment, this aspect should be included. Likewise, instead of estimating a multidimensional Sobol index, Sobol indices have been estimated independently on each component. Although the difference of variance (due to units) was an argument to justify the average Sobol indices used

in the chapter 6, multidimensional Sobol index could be worth to be estimated the next time.

Reconstruction method

The present study was restricted to the 1-dimensional case, for two reasons: the recoding in Fortran is time-consuming and the 1D code runs faster. Given that the target of this thesis was to make the sensitivity analysis, the 1D code was enough. However, turbulence is very different in 3D (see chapter 1 and the comparison between TKE estimators) and lidar measurements are limited by the Cyclop's dilemma (while they are not in 1D). Thus, the extension to 3-dimensional case is a very interesting prospect, even if some major results (like the tuning strategy) might hold.

Known limitations of the reconstruction method have been recalled in the chapter 5. One of them is that it does not work in stable conditions. Further work on the Lagrangian model must be done in order to get a suitable model. Given the differences of nature between turbulent and stable cases, switching model techniques might be helpful to solve this issue.

The dataset used as a reference was taken in fair weather, well developed turbulence. A prospect would be to make the same study with another reference. It could tell if the present study is confirmed whatever the reference is. It is likely that the conclusion will be different if the reference is taken in sable atmosphere. Although the Lagrangian model is not expected to work in stable cases, the sensitivity analysis might points out what are the parameters to change in order to adapt the model to such cases.

Part III

Appendices

Appendix A

Theoretical background

A.1 Probability

This section intends to homogenize the knowledge common to most of readers, including those not practising probability on a daily basis. There exists already myriad of books on probability where more details (such as demonstrations) and more examples can be found. Some of them have been used to write this section, as Barbe and Ledoux (1998) (in French) and Gardiner (2009); Øksendal (2013) (in English) in addition to many roaming lectures.

A.1.1 Elements of measure theory

Let E be a non-empty space in which belong the object to measure. For example, if one wants to measure lengths, E will be the set of real numbers. The space E will be called the *state* space. The set of all parts of E is denoted by $\mathcal{P}(E)$. Parts of the state space are accessible for measure only if they are in a **σ-algebra**.

Definition A.1 (σ-algebra)**.**

With words *A σ-algebra is a subset of $\mathcal{P}(E)$ stable by complementarity and countable union.*

Formally *$\mathcal{E}$ is a σ-algebra if it satisfies the 3 following statements:*

- $E \in \mathcal{E}$
- $\forall A \in \mathcal{E},\ A^{\mathsf{c}} \in \mathcal{E}$
- $\forall n \in \mathbb{N},\ A_n \in \mathcal{E},\ \bigcup_{n\in\mathbb{N}} A_n \in \mathcal{E}$

In a set E, there are many σ-algebras. As an example, $\{\emptyset, E\}$ is a σ-algebra of E (it is called the *coarse* σ-algebra). $\mathcal{P}(E)$ is a σ-algebra of E (it is called the *discrete* σ-algebra). For any subset $A \subset E$, $\{\emptyset, A, A^{\mathsf{c}}, E\}$ is a σ-algebra of E (it is called the σ-algebra *generated by* A).

A very common σ-algebra is the one corresponding to lengths, surfaces or volumes. On $\mathbb{R}$, it is the σ-algebra generated by the intervals, denoted by $\mathcal{B}(\mathbb{R})$. On a more general space E, it is the σ-algebra generated by the open sets of E, denoted by $\mathcal{B}(E)$. Lengths, surfaces or volumes are then given by the **measure** of the elements in the σ-algebra.

Definition A.2 (Measure)**.**

With words *A measure μ is an application associating to any measurable element a positive real number, such that*

- *the measure of the empty set is zero.*
- *the measure of the union of disjoint sets is the sum of each set measure.*

Formally

$$\mu : \begin{array}{c} \mathcal{E} \longrightarrow [0, +\infty] \\ A \longmapsto \mu(A) \end{array}$$

and satisfies the 2 following statements:

- $\mu(\emptyset) = 0$
- $\forall n \in \mathbb{N}$, $A_n \in \mathcal{E}$ *such that* $\forall i \neq j$, $A_i \cap A_j = \emptyset$, $\quad \mu\left(\bigcup_{n\in\mathbb{N}} A_n\right) = \sum_{n\in\mathbb{N}} \mu(A_n)$

Intuitively, what in this definition makes it a measure? First, the fact that it is a mapping that associates to each measurable set a real number. The measure "summarizes" the set by a real number (for example its size). Second, the additivity: the size of disjoint sets is the addition of the size of all sets.

The measure defined on $\mathcal{B}(E)$ that correspond to lengths, surfaces and volumes is called the Borel measure λ. When this measure is extended to null subsets[1], it is called the Lebesgue measure and denoted Λ. [2]

Probability are measures on another space than the state space. Let space Ω be a space endowed with a σ-algebra $\mathcal{F}$. A measurable element of the probability space $A \in \mathcal{F}$ is a called an **event**. The space Ω is called the **universe**.

[1] a set N is null for the measure λ if $\exists A \in \mathcal{B}$, $N \subset A$ and $\lambda(A) = 0$

[2] Non measurable elements exist only if one accepts the axiom of choice. But this axiom is controversial because it yields to paradoxical results, such as Banach-Tarski theorem (well explained in this Youtube video). In summary any subset of $\mathbb{R}$, unless very exotic, is measurable.

Definition A.3 (Probability)**.**

With words *A probability is a measure with maximum value 1.*

Formally

$$\mathbb{P} : \begin{array}{c}(\Omega, \mathcal{F}) \longrightarrow [0,1] \\ A \in \mathcal{F} \longmapsto \mathbb{P}(A)\end{array}$$

with $\mathbb{P}(\Omega) = 1$.

Hence, the probability for an event to occur is the "size" of the event, as measured by $\mathbb{P}$.

A.1.2 Random variables

Measure theory is the starting point to study random variables. By default, random variables are real valued, for the sake of simplicity (and because the differences are most of the time an extension of the notations to complex or multi-dimensional case).

A.1.2.1 Definition

Definition A.4 (Random variable)**.**

With words *A random variable X is a measurable function from a probability space (hidden) to a state space (accessible by measurement).*

Formally

$$X : \begin{array}{c}(\Omega, \mathcal{F}, \mathbb{P}) \longrightarrow (E, \mathcal{E}) \\ \omega \longmapsto X(\omega)\end{array}$$

where X verifies

$$\forall A \in \mathcal{E},\ X^{-1}(A) \in \mathcal{F}.$$

with $\quad X^{-1}(A) = \{\omega \in \Omega : X(\omega) \in A\}$.

Given $\omega \in (\Omega, \mathcal{F}, \mathbb{P})$, the value $X(\omega)$ is called a ***realization*** *of X. The figure A.1 gives a visual sight of the notation.*

The probability space $(\Omega, \mathcal{F}, \mathbb{P})$ is considered as hidden. It is the source of randomness: nothing is known about Ω but the σ-algebra $\mathcal{F}$ and the probability $\mathbb{P}$ exist. In a coin toss, for example, Ω could be the $\mathbb{R}^3 \times \mathbb{R}^3$ space of the initial position and speed of coin... or it could be $\mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^3$ space of the initial position and speed of coin and wind speed... or even more complex. Instead of trying to set what could be influencing the coin, we settle that it is

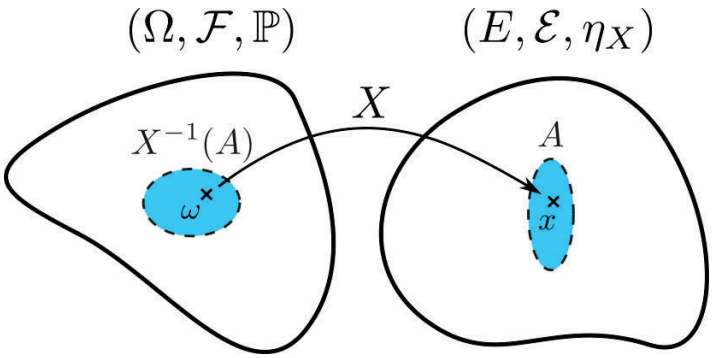


Figure A.1 – Diagram to introduce random variables: the random variable X is a measurable function from an unknown departure space Ω to a measurable arrival space E.

unknown, and the unknown part is the source of randomness. Informally, randomness is an unknown variable.

A.1.2.2 Probability law

Given a random variable X (definition A.4), the departure space $(\Omega, \mathcal{F}, \mathbb{P})$ is equipped with the measure $\mathbb{P}$. The arrival space $(E, \mathcal{E})$ is measurable. With this ingredients, one can define a measure on the arrival space: the probability law of X.

Definition A.5 (Law of a random variable)**.**

With words *The law η_X of the random variable X is the pushforward image of the probability $\mathbb{P}$ on the state space.*

Formally

$$\forall A \in \mathcal{E},\ \eta_X(A) = \mathbb{P}\left(X^{-1}(A)\right)$$

with $X^{-1}(A) = \{\omega \in \Omega,\ X(\omega) \in A\}$.

To ensure the pre-image $X^{-1}(A)$ is measurable by $\mathbb{P}$, the fact the X is a measurable function is essential. As $\mathbb{P}$ is a probability, η_X arrival values are in $[0, 1]$:

$$\eta_X : \begin{array}{ccc} \mathcal{E} & \longrightarrow & [0,1] \\ A & \longmapsto & \eta_X(A) = \mathbb{P}\left(X^{-1}(A)\right) \end{array}$$

As a consequence, η_X is also a probability on $(E, \mathcal{E})$:

$$\eta_X(E) = \mathbb{P}(\Omega) = 1 \tag{A.1}$$

A.1.2.3 Probability density function

Usually, in problems with Euclidian geometry, the state space $(E, \mathcal{E})$ is equipped with the Lebesgue measure Λ (the measure which correspond to length, areas and volumes, including negligible elements). It is very informative to know how the probability law η_X of the random variable X is different from the Lebesgue measure Λ. Both η_X and Λ are measures on $(E, \mathcal{E})$. Given a measurable element $A \in \mathcal{E}$, we want to compare $\eta_X(A) = \int_{x \in A} d\eta_X(x)$ and $\Lambda(A) = \int_{x \in A} d\Lambda(x)$. In particular, is there a weight function f_X (with which properties?) such that

$$\int_{x \in A} d\eta_X(x) = \int_{x \in A} f_X(x) d\Lambda(x)$$

The answer of this question is given by the theorem of Radon-Nikodym.

Theorem A.1 (Radon-Nikodym)**.**
Let ν and μ be two positive and σ-finite measures on the measurable space $(E, \mathcal{E})$ such that μ is absolutely continuous with respect to ν ($\forall A \in \mathcal{E}, \mu(A) = 0 \Rightarrow \nu(A) = 0$).
Then there exists a function h such that

$$h : E \to [0, +\infty[, \ / \ \forall A \in \mathcal{E}, \quad \mu(A) = \int_A h d\nu$$

.
The function h is called the Radon derivative of μ with respect to ν (or "density of μ w.r.t ν") and we denote

$$h = \frac{d\mu}{d\nu}$$

Notice that the function h is almost unique. The ambiguity relies on negligible sets for μ. The **probability density function** (or PDF) is the Radon derivative of the probability law, η_X, with respect to a measure of interest, ν.

Definition A.6 (Probability density function)**.**

With words *The probability density function of a random variable X is the weight function to compare the probability law with a measure of interest ν.*

Formally

$$f_X : \begin{array}{ccc} E & \to & [0, +\infty[\\ x & \mapsto & f_X(x) \end{array}$$

such that

$$\forall A \in \mathcal{E}, \quad \int_{x \in A} d\eta_X(x) = \int_{x \in A} f_X(x) d\nu(x)$$

To be fully exact, one should always precise the measure of interest ν to which the PDF refers. In our case, the measure of interest is the Lebesgue measure (corresponding to lengths,

surfaces and volumes): $\nu = \Lambda$. It follows that f_X is Lebesgue-integrable with integral 1.

$$\int_{x \in E} f_X(x) d\Lambda(x) = \eta_X(E) = 1 \tag{A.2}$$

The primitive of the probability function with 0 for limit when $x \to -\infty$, denoted by $F_X(x)$, is called the **cumulative distribution function** or CDF. For 1-dimentional real-valued random variable[3], it is defined by equation A.3.

$$F_X(x) = \int_{-\infty}^{x} f_X(y) dy = \eta_X(]-\infty, x]) = \mathbb{P}(X \leqslant x) \tag{A.3}$$

The last expression allows to define the CDF even when the random variable has no PDF. For example, the random variable with the Cantor function as cumulative density function has no probability density function. Indeed, the law of such a random variable is not absolutely continuous with respect to Lebesgue measure[4]. Hence the Radon-Nikodym theorem does not apply. It gives an example of a random variable that have a continuous CDF but no PDF.

A.1.2.4 Expected value

A random variable (function from unknown space) is fully described by its probability law (measure) or its probability density function (real positive valued function on the state space), when it exists. The expected value describes the central trend of the random variable by a scalar[5] denoted $\mathbb{E}[X]$. It requires additional regularity of the random variable, expressed with L^p spaces.

Definition A.7 ($\mathcal{L}^p$ space)**.**

With words *The $\mathcal{L}^p$ space (shortcut for $\mathcal{L}^p(E, F)$) is the ensemble of functions from E to F, with $(E, \mathcal{E}, \mu)$ a measured space, which are integrable at order p for the measure μ.*

Formally

$$\mathcal{L}^p(E, F) = \left\{ f \,:\, E \to F \;\middle/\; \int |f(t)|^p d\mu(t) < +\infty \right\}$$

The application $\|f\|_p = (\int |f(t)|^p d\mu(t))^{\frac{1}{p}}$ is a semi-norm for $\mathcal{L}^p$ because $\|f\|_p = 0 \nRightarrow f = 0$ (take $f = \mathbf{1}_{x=0}(x)$ for instance). Hence, the space $\mathcal{L}^p$ is modified in order to make $\|f\|_p$ a norm. The modification consists in merging all functions equal almost everywhere, according to μ.

[3] When $E \subset \mathbb{R}^n$, the cumulative distribution function is defined by integration on the cuboid $\otimes_{i=1}^{n}]-\infty, x_i]$.
[4] if K_3 is the Cantor set, $\eta_X([0,1] \setminus K_3) = 0$, while $\Lambda([0,1] \setminus K_3) = 1$
[5] a single element of the state space. If E is multi-dimensional, $\mathbb{E}[X]$ will be multi-dimensional as well.

Definition A.8 (L^p space)**.**

With words *The L^p space is the $\mathcal{L}^p$.*

Formally

$$L^p(E,F) = \{[f],\ f \in \mathcal{L}^p(E,F)\}$$

with $[f]$ any element of $\{g \in \mathcal{L}^p(E,F),\ \mu(\{x, f(x) \neq g(x)\}) = 0\}$.

Applied to random variables, the previous definition with $p = 1$ is an assumption to define the expected value. Let $X \in L^1(\Omega, E)$ in the following definition.

Definition A.9 (Expected value)**.**

With words *The expected value of a random variable X is the average value of X considering every element ω in the universe Ω with the measure $\mathbb{P}$.*

Formally

$$\mathbb{E}[X] = \int_{\omega\in\Omega} X(\omega) d\mathbb{P}(\omega) = \int_{x\in E} x\, d\eta_X(x)$$

The equality $\int_{\omega\in\Omega} X(\omega)d\mathbb{P}(\omega) = \int_{x\in E} x\, d\eta_X(x)$ come from the definition of η_X and the substitution $x = X(\omega)$.

When X has a probability density function, the integral can be written with the Lebesgue measure, which is easier to estimate.

$$\mathbb{E}[X] = \int_{x\in E} x\, d\eta_X(x) = \int_{x\in E} x f_X(x)\, d\Lambda(x) \tag{A.4}$$

However, the existence of $\mathbb{E}[X]$ is not guaranteed. For example, the Cauchy distribution has no expected value. Let Z be a random variable with the following PDF:

$$f_Z(z) = \frac{1}{\pi}\frac{1}{1+z^2} \tag{A.5}$$

Then $z f_Z(z) \underset{z\to\infty}{\sim} 1/(\pi z)$, thus the integral $\int z f_Z(z) dz$ does not converge. The Cauchy distribution has no expected value.

A.1.2.5 Momenta

The definition of momenta involves the expected value $\mathbb{E}[X]$. To do so, it is denoted for any test function φ such that $\int \varphi(x) f_X(x) dx$ converges[6],

$$\mathbb{E}[\varphi(X)] = \int_{x\in E} \varphi(x)\, d\eta_X(x) = \int_{x\in E} \varphi(x) f_X(x)\, d\Lambda(x) \tag{A.6}$$

[6]The equality A.6 also defines the distribution associated to X.

Momenta are the scalars obtained by choosing φ of the form $x \mapsto x^n$ with $n \in \mathbb{N}$. The integer n is called the order of the momentum. They are defined for L^n random variables.

Definition A.10 (Momenta)**.**

With words *The n-order momentum of a random variable X is the mathematical expectation of X^n.*

Formally

$$\mathbb{E}\left[X^n\right] = \int_{x \in E} x^n \, d\eta_X(x)$$

Centered momentum are often used :

$$\mathbb{E}\left[(X - \mathbb{E}\left[X\right])^n\right] = \int_{x \in E} (x - \mathbb{E}\left[X\right])^n \, d\eta_X(x) \tag{A.7}$$

In particular, the 2-order centred momentum is the variance of X :

$$V(X) = \mathbb{E}\left[(X - \mathbb{E}\left[X\right])^2\right] = \mathbb{E}\left[X^2\right] - \mathbb{E}\left[X\right]^2 \tag{A.8}$$

It is of particular interested because it is easy to interpret as the discrepancy of the random variable around the mean value. For a Gaussian random variable, mean and variance are the only parameters needed to know everything about the random variable.

A.1.2.6 Conditional probability

Briefly, we recall some useful results for conditional probability such as the law of total probability or Bayes' theorem. Let A and B be two events, with $\mathbb{P}(B) \neq 0$. The conditional probability of A given B is denoted $\mathbb{P}(A|B)$.

Definition A.11 (Conditional probability)**.**

With words *The conditional probability of A given B is the ratio between $\mathbb{P}(A \cap B)$ and $\mathbb{P}(B)$.*

Formally

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)}$$

The conditional probability is the probability that A and B occur, considering the event B happens for sure. The events A and B are independent if and only if $\mathbb{P}(A|B) = \mathbb{P}(A)$. Noticing that $\mathbb{P}(A|B)\,\mathbb{P}(B) = \mathbb{P}(A \cap B) = \mathbb{P}(B|A)\,\mathbb{P}(A)$ one gets Bayes' formula:

Theorem A.2 (Bayes)**.** *Let A and B be two event of $(\Omega, \mathcal{F}, \mathbb{P})$ with $\mathbb{P}(B) \neq 0$*

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(B|A)\,\mathbb{P}(A)}{\mathbb{P}(B)}$$

Bayes' formula gave its name to Bayesian statistics and probability, in opposition to the historical frequentist approach. The adjective *Bayesian* has now become synonym of *using a prior knowledge*. The names of the terms in the theorem also are commonly used in a broader sense.

- $\mathbb{P}(A|B)$ is the *posterior*, usually the output of Bayes' theorem. It tells what happens after the event B has been taken into account.
- $\mathbb{P}(A)$ is the *prior*, the external knowledge one wants to exploit. It can be a theoretical insight, some expert advice or the information of a model... The choice of the prior influences widely the result. The prior knowledge has to be trustworthy to take benefit of Bayesian formulation.
- $\mathbb{P}(B|A)$ is the *likelihood*, the information usually coming from the observations. The ratio $\mathbb{P}(B|A)/\mathbb{P}(B)$ is the normalized likelihood.

Some probability laws are known only conditionally to some events. A useful tool to exploit such knowledge is the formula of total probability:

Theorem A.3 (Law of total probability)**.** *Let A be an event of $(\Omega, \mathcal{F}, \mathbb{P})$ and $(B_n)_{n\in\mathbb{N}}$ a partition of Ω and such that $\mathbb{P}(B_n) \neq 0$.*

$$\mathbb{P}(A) = \sum_{n\in\mathbb{N}} \mathbb{P}(A|B_n)\,\mathbb{P}(B_n)$$

A.2 Simulation of random variables

As presented in the previous section, random variables are "measurable functions from a probability space to a state space" (definition A.4). They induce several objects such as probability law (def. A.5), probability density function (def. A.6), cumulative distribution function (eq. A.3)... But in practice, one has to transpose these objects into numerical equivalents, so that actual manipulations on them become possible. This section intents to present how random variables are generated on a computer. It is important because the quality of the pseudo-random generator is critical for the filtering method. It also helps to understand the non-linear filtering algorithm.

A.2.1 Monte Carlo approximation

Monte Carlo approximation consists in using independent copies of the target random variable, X, to estimate valuable information about X (expected value or PDF shape for example). As in the previous section, the state space is denoted $(E, \mathcal{E}, \eta_X)$ and the probability space is denoted $(\Omega, \mathcal{F}, \mathbb{P})$. They are both measured space with respective measures η_X (probability law of random variable X) and $\mathbb{P}$. Let us consider N independent copies of X, denoted by $X^1, ..., X^N$. Throughout the manuscript, the key character "$\rightsquigarrow$" will mean "follows the law". As an example, $X \rightsquigarrow \mathcal{U}(0,1)$ means "X follows the law $\mathcal{U}(0,1)$" (uniform between 0 and 1). Thus, for any $i \in [\![1, N]\!]$, $X^i \rightsquigarrow \eta_X$ and all X^i are independent.

The expected value, $\mathbb{E}[X]$, (when it exists) is approached by an ensemble average with an almost sure convergence (strong law of large numbers). The ensemble average is an unbiased estimator of the expected value. For any test function φ, we also have almost sure convergence.

$$\mathbb{E}[\varphi(X)] = \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} \varphi(X^i)$$

As the previous equality holds for any test function φ, the empirical distribution η_X^N (defined by $< \eta_X^N, \varphi >= \frac{1}{N}\sum_{i=1}^{N} \varphi(X^i)$) converges almost surely toward the target distribution η_X (defined by $< \eta_X, \varphi >= \mathbb{E}[\varphi(X)]$). The empirical distribution is expressed with Dirac distribution (δ_a is defined by $< \delta_a, \varphi >= \varphi(a)$).

$$\eta_X = \lim_{N\to\infty} \eta_X^N = \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} \delta_{X^i} \tag{A.9}$$

The probability density function, $f_X(x)$, (when it exists) is approached by a histogram. The probability for X to be in a neighbourhood of a point x (let say $V_x = [x - \Delta x/2, x + \Delta x/2]$) is expressed in two ways: one involve the probability density function, the other involve the

Mathematical object (def)	**Numerical equivalent**	**Link**
Probability law (A.5)	Monte Carlo approximation	$\eta_X = \lim_{N\to\infty} \eta_X^N = \frac{1}{N}\sum_{i=1}^{N} \delta_{X^i}$
Probability density function (A.6)	Histogram	$\int_{x\in V_x} f_X(x')dx' = \lim_{N\to\infty} \frac{\lvert\{X^i \in V_x\}\rvert}{N}$
Expected value (A.9)	Ensemble average	$\mathbb{E}\left[\varphi(X)\right] = \lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} \varphi(X^i)$

Table A.1 – Correspondence between theoretical objects and their numerical equivalent with Monte Carlo method.

histogram.

$$\begin{aligned} \eta_X([x-\Delta x/2, x+\Delta x/2]) &= \int_{x-\Delta x/2}^{x+\Delta x/2} d\eta_X(x') \\ &= \int_{x-\Delta x/2}^{x+\Delta x/2} f_X(x')dx' \\ &= f_X(x)\Delta x + o(\Delta x) \end{aligned} \tag{A.10}$$

$$\begin{aligned} \eta_X^N([x-\Delta x/2, x+\Delta x/2]) &= \int_{x-\Delta x/2}^{x+\Delta x/2} \frac{1}{N}\sum_{i=1}^{N} \delta_{X^i}(dx') \\ &= \frac{1}{N}\sum_{i=1}^{N} \underbrace{\int_{x-\Delta x/2}^{x+\Delta x/2} \delta_{X^i}(dx')}_{=1 \text{ if } X^i\in V_x, 0 \text{ else}} \\ &= \frac{1}{N}\lvert\{i,\, X^i \in [x-\Delta x/2, x+\Delta x/2]\}\rvert \end{aligned} \tag{A.11}$$

The probability density function appears in A.10 (approximation $\Delta x \to 0$). The equation A.11 leads to the proportion of sample present in $[x-\Delta x/2, x+\Delta x/2]$. Hence, PDF are numerically visualized with histograms.

A.2.2 Pseudo-random generators

All valuable approximations in table A.1 rely on the assumption that any X^i is an independent copy of X. How to obtain such a sample numerically? The main problem is that computer are deterministic machine. Perfect randomness is thus not accessible, but it is possible to build pseudo-random generator of numbers.

Pseudo-random generators are recursive sequences chosen to vary rapidly, but still dwell inside [0,1]. Thus, pseudo-random generators provide approximation of uniform-distributed samples. The generator of Fortran 90 and Scilab languages are tested, because both have been used to implement the reconstruction algorithm. Pseudo-random generators are of the

form of the equation A.12 (Hull and Dobell, 1962).

$$\begin{cases} x_{n+1} & = & ax_n + c \ [\text{mod } m] \\ x_0 & & \text{given} \end{cases} \tag{A.12}$$

The starting point x_0 is called the **random seed**. Two sequences of same length generated with the same seed will be equal, even though they look random. This a useful tool to make repeatable random experiments. The seed can also be modified manually to ensures more randomness.

A random generator is said good when it fulfils the best the next 4 qualities (by order of importance):

1. The generated sample follows is uniformly distributed (its discrepancy is low).
2. Individuals in the sample are independent.
3. The period is large.
4. The sample is quickly generated.

The associated tests are recap in the table A.2 and further details are given below.

Tested criterion	Law $= \mathcal{U}(0,1)$		Independence	Period	Speed
Test	Kolmogorov-Smirnov	χ^2	Autocorrelation	Theory	Timing

Table A.2 – Recap of criteria and tests used for the verification of the pseudo-random numbers generators

The first property is assessed with two statistical tests. The Kolmogorov-Smirnov test compares the empirical cumulative distribution function with the theoretical CDF of $\mathcal{U}(0,1)$. The statistic of the test is $t = \sup_x |F_X^N(x) - F_X(x)|$ where $F_X^N(x)$ is the empirical CDF and $F_X(x)$ is the theoretical CDF. The χ^2 test compares the empirical histogram with the theoretical one, given a number of classes. The statistic of the test is $t = \sum_{j=1}^{J} \frac{(N\hat{p_j} - Np_j)^2}{Np_j}$ where j is a class, N is the size of the sample, $\hat{p_j}$ is the empirical probability for an individual to be in the class j and p_j its theoretical equivalent. For both tests, the null hypothesis is "the sample follows a $\mathcal{U}(0,1)$". The conclusions of these tests are shown in table A.3 and A.4.

The second property is checked by having a look on the auto-correlation. In figure A.2 (resp. A.3) is shown the autocorrelation of the Scilab (resp. Fortran) sample. On both graphics, the blue dashed line draw the punctual 95% prediction interval for an autocorrelation of 0. As long as the autocorrelation remains in between the two blue dashed lines, it can be neglected. Except few overshoots, the auto-correlation is within the prediction bound of 0. Both generators can be assumed to provide independent realisations every time.

$$\chi^2 \textbf{ test : } t = \sum_{j=1}^{J} \frac{(N\hat{p_j} - Np_j)^2}{Np_j}$$

	Score t	p-value	**Conclusion**
Scilab	10.954	$0.279 > \alpha$	H_0 not rejected
Fortran	9.05	$0.4327 > \alpha$	H_0 not rejected

Table A.3 – Results of χ^2 test with $J = 10$ classes and a sample of 10^4 individuals. The first type error threshold is chosen to $\alpha = 0.05$. The null hypothesis "the sample follows a $\mathcal{U}(0, 1)$" is accepted for both Scilab and Fortran generators.

$$\textbf{Kolmogorov-Smirnov test : } t = \sup_x |F_N(x) - F(x)|$$

	Score t	p-value	**Conclusion**
Scilab	0.0093	$0.3473 > \alpha$	H_0 not rejected
Fortran	0.0084	$0.4829 > \alpha$	H_0 not rejected

Table A.4 – Results of Kolmogorov-Smirnov test with a sample of 10^4 individuals. The first type error threshold is chosen to $\alpha = 0.05$. The null hypothesis "the sample follows a $\mathcal{U}(0, 1)$" is accepted for both Scilab and Fortran generators.

The third criterion (the periodicity) is given by the parameters in the equation A.12. For Fortran, the value of $2^{32} - 1$ is given in Marsaglia et al. (2003) for 1-dimensional samples. For Scilab, the documentation of the `rand` function gives the value of 2^{31}. Both are of order of magnitude 10^9. If the program needs more random values than the period, the generated sequence of random value will start over within a run of code. Having a large period ensures the pseudo-random generators are "random" until the end. To avoid this shortcoming, the random seed is reset to a new value at different moments in the program.

The last criterion (speed) is tested by timing the generation of the 10^4 values.

The four criteria and their test are given in the table A.2. The table A.5 gives the conclusions of these tests : both are good enough pseudo-random generators, Fortran's one is better for both speed and period.

Langage	**Time to generate 10^4 values (s)**	**Period**	**Independence**	**Uniformity**
Fortran	$2.3 \cdot 10^{-4}$	$2^{32} - 1$	OK	OK
Scilab	$3 \cdot 10^{-2}$	2^{31}	OK	OK

Table A.5 – Conclusions of all tests carried out on both generators. Both are good for use, but Fortran's one is way faster.

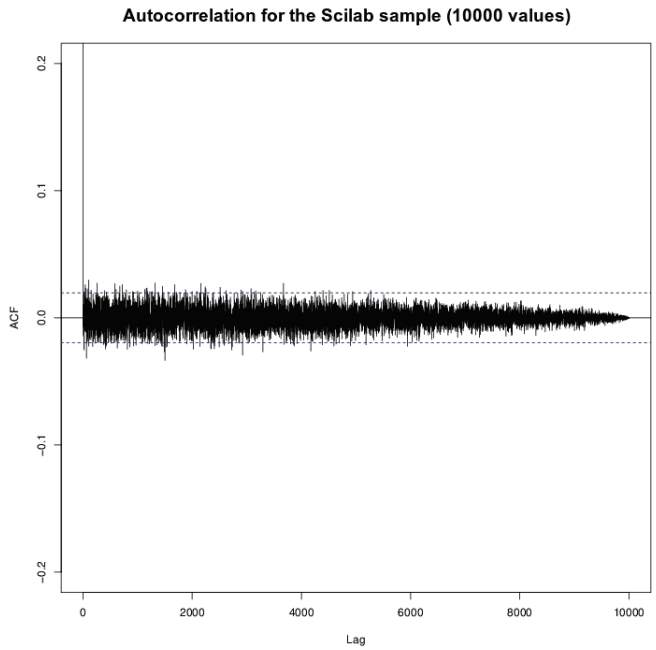


Figure A.2 – Autocorrelation of Scilab sample of 10000 values. Values inside the two dashed blue lines are in the punctual 95% prediction interval of a 0 correlation estimation.

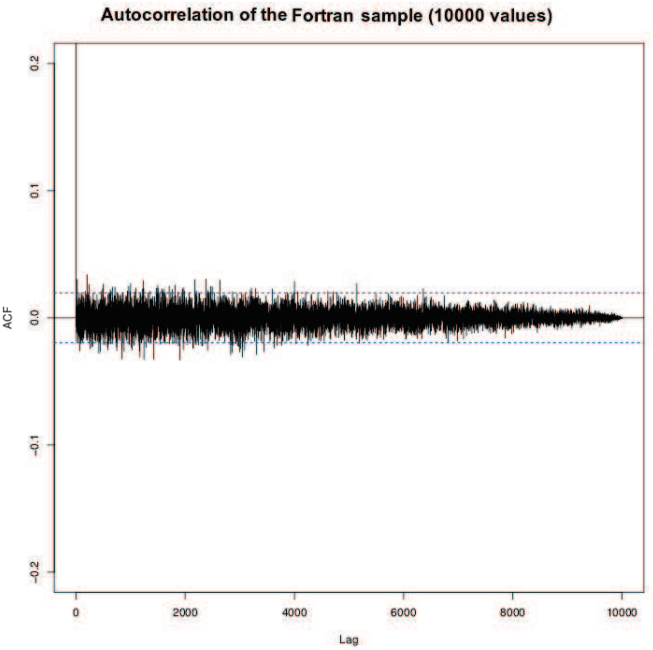


Figure A.3 – Autocorrelation of Fortran sample of 10000 values. Values inside the two dashed blue lines are in the punctual 95% prediction interval of a 0 correlation estimation.

A.2.3 Non-uniform sampling

As we have seen, pseudo-random generators provide good approximation of uniform-distributed samples. Others distributions are generated from uniform-distributed samples. Specific solutions exist to generate Gaussian-distributed sample (such as Box-Muller algorithm (Box et al., 1958)). We are interested in the simulation of *any* distribution. The distribution transform is used in the filtering method, to generate the posterior from the prior and the likelihood.

To describe the target distribution, the simplest way is to have a PDF (possibly unnormed). Let $G(x)$ a function on E that we call the **potential**. The potential associates to any x its weight in the target distribution. The interpretation of $G(x)$ is similar to $f_X(x)$. As a consequence, G have to be a real positive and integrable function.

$$G : \begin{array}{ccc} \mathbb{R} & \to & [0, +\infty[\\ x & \mapsto & G(x) \end{array} \qquad \text{with } \int_{-\infty}^{+\infty} G(y)dy < +\infty \tag{A.13}$$

Let $(u^i)_{i\in[\![1,N]\!]}$ be an uniform-distributed Monte Carlo sample. From this sample, we want to build a G-distributed sample $(x^i)_{i\in[\![1,N]\!]}$. As an example, we will take G the function drawn in blue on A.4 (sum of 2 Gaussian because bi-modal and simple). The potential G is an unnormed PDF. To get an actual PDF, one must use $G/\int G(y)dy$. Its associated normed

CDF is denoted w and is defined by equation A.14.

$$\forall x \in \mathbb{R}, \quad w(x) = \frac{\int_{-\infty}^{x} G(y)dy}{\int_{-\infty}^{+\infty} G(y)dy} \tag{A.14}$$

On figure A.4, an example of G is drawn in blue line and the corresponding w is drawn in red.

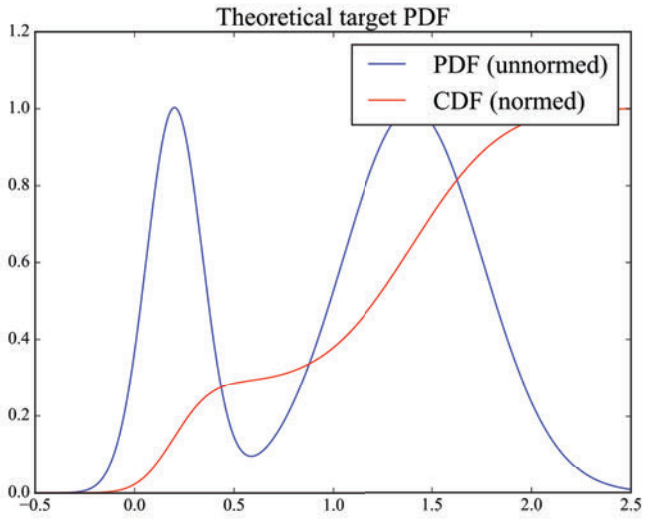


Figure A.4 – Example of unnormed PDF (or potential) G in blue (sum of 2 Gaussian) and its associated normed CDF w.

The cumulative density function goes from the range of variation of X to $[0, 1]$ and is steeper where G is high. An uniform-distributed sample is converted into a G-distributed sample by inverting the cumulative density function. Hence, one can build the G-distributed sample by applying the inverse of w to every $(u^i)_{i\in[\![1,N]\!]}$ (A.15).

$$\forall i \in [\![1, N]\!], \; x^i = \inf\{x, \, w(x) > u^i\} \tag{A.15}$$

In figure A.5, one can see an uniform-distributed sample $(u^i)_{i\in[\![1,N]\!]}$ on the y-axis, for which every element is inverted with the w function (same example as in figure A.4). Thus, on the x-axis is shown the resulting sample $(x^i)_{i\in[\![1,N]\!]}$ which is build according the relation A.15.

Theoretically, we have seen how to generate G-distributed random variable, for any real positive integrable function G. But in practice, computer do not use analogic functions but arrays of data. Hence, it is necessary to convert the previous reasoning in the discrete case in order to get an algorithm.

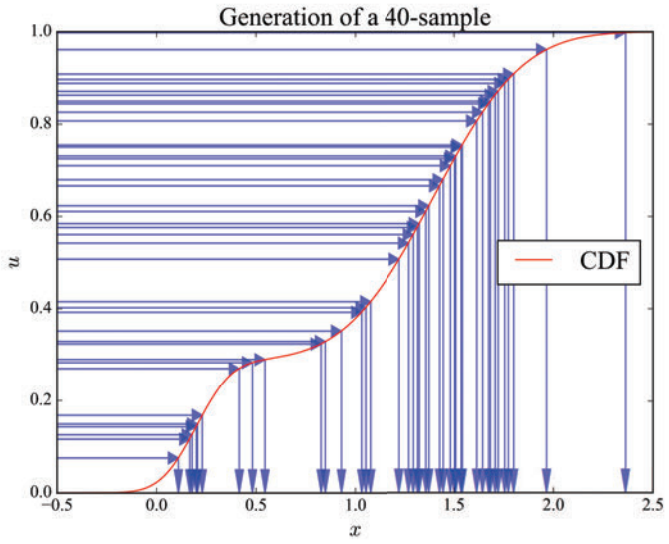


Figure A.5 – An uniform sample is changed into the target law sample using the inversion of CDF. On the y-axis, arrows are uniform-distributed, on the x-axis they are G-distributed.

Now $\mathbf{G}$ denotes an array of size M, ($\mathbf{G} = (G_1, ..., G_M) \in \mathbb{R}^M$) which gives the potential on a discrete set of points $\mathbf{x} = (x_1, ..., x_M) \in \mathbb{R}^M$[7]. For any $k \in [\![1, M]\!]$, $G_k = G(x_k)$. In this framework, $\mathbf{w} = (w_1, ..., w_M) \in \mathbb{R}^M$ is the discrete CDF, and its definition equation of w (A.14) is changed into (A.16):

$$\forall k \in [\![1, M]\!], \quad w_k = \frac{\sum_{m=1}^{k} G_m}{\sum_{m=1}^{M} G_m} \tag{A.16}$$

The inversion of w is modified as well, because the infimum has to belong to $\mathbf{x}$. Using the fact that $\mathbf{w}$ is a sorted array, the operation (A.15) is slightly modified into (A.17).

$$\forall i \in [\![1, N]\!], \; x^i = x_k \;\; \text{with } k = \min_{m \in [\![1,M]\!]} \{w_m > u^i\} \tag{A.17}$$

Gathering all the previous elements together, the algorithm A.1 is able to generate sample according to the potential G. This algorithm has been applied to the example given in figures A.4 and A.5 with $M = 1000$ (number of points in the discretization) and $N = 5000$ (size of the sample). The result is shown in figure A.6. One can see that the normed histogram (Monte Carlo equivalent of PDF) is closely comparable to the target PDF (normed potential G). It proves that the algorithm A.1 correctly simulate G-distributed random variable. This algorithm will be used many times to produce samples according to an empirical probability law, only described by arrays of number.

[7]Beware: x_k is a point on which $\mathbf{G}$ is known, x^i is a element of generated sample.

Algorithm A.1 Generate G-distributed samples : $X = \text{sampling}(\mathbf{G}, \mathbf{x})$

Input: $\mathbf{G} = (G_1, ..., G_M)$, $\mathbf{x} = (x_1, ..., x_M)$ # with $G_k = G(x_k)$
Output: $X = (x^1, ..., x^N) \rightsquigarrow G$
  $U = (u^1, ..., u^N) \rightsquigarrow \mathcal{U}(0, 1)$
  $\mathbf{w} = \text{cumsum}(\mathbf{G})/\text{sum}(\mathbf{G})$
  for $i \in [\![1, N]\!]$ **do**
    $k = \min_{m, \in [\![1,M]\!]}\{w_m > u^i\}$
    $x^i = x_k$
  end for
  return $X = (x^1, ..., x^N)$

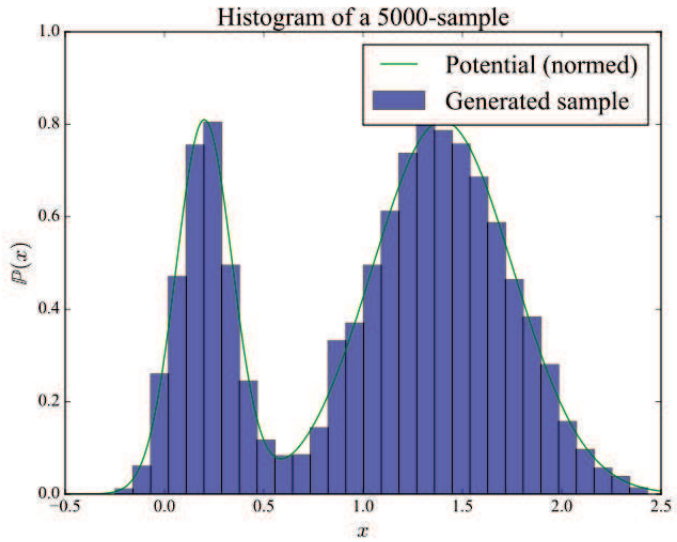


Figure A.6 – Example of generation of samples with the algorithm A.1. Histogram of the sample follows the shape of the potential G given.

A.3 Stochastic processes

The random variables used to describe physical phenomenon might be changing with time. Hence we introduce stochastic processes ("stochastic" has exactly the same sense as "random", but "stochastic" is the word in use). Introduction to stochastic processes can be found in Øksendal (2013) and Gardiner (2009).

Definition A.12 (Stochastic process)**.**

With words *A stochastic process is a family of random variable.*

Formally $(X_t)_{t\in[0,T]}$ *is a stochastic process means that*

$$\forall t \in [0,T],\ X_t : \begin{array}{c}(\Omega, \mathcal{F}, \mathbb{P}) \longrightarrow (E, \mathcal{E}) \\ \omega \longmapsto X_t(\omega)\end{array} \text{ is a random variable.}$$

with $T > 0$.
However, the trajectory of X_t *(the function* $t \mapsto X_t(\omega)$, $\forall \omega \in \Omega$*) is deterministic.*

A stochastic process is thus a function which depends on an random event: a "random function". Although, unlike a function, there is no representative curve for a stochastic process. Instead, one can draw its momentum or few trajectories. The average is a deterministic function ($t \mapsto \mathbb{E}[X_t]$). The variance is a deterministic function ($t \mapsto V(X_t) = \mathbb{E}\left[(X_t - \mathbb{E}[X_t])^2\right]$). Momenta provide only a partial information on the stochastic process. However, they help to get a mental image of a stochastic process. This is now illustrated on two famous examples: the Brownian motion, and the solution of Langevin equation.

A.3.1 The Brownian motion

In figure A.7 is shown an example of stochastic process with $\mathbb{E}[X_t] = 0$ and $\text{cov}(X_t, X_s) = \min(t, s)$. Some trajectories are drawn in red. The ensemble of trajectories expend around the mean following the increasing variance (dashed blue lines). From a single trajectory it is not obvious to see that expansion (figure A.8). This is why plotting several trajectories is a way to visualize the "random dimension": some features of the trajectory are explained by the global trend (the expansion), and other are just due to randomness (the side of expansion, local variations). For example, if one would have only the average value and a single realization (as in figure A.8) one would conclude that the realisation is in disagreement with the average, although it is simulated in the exact same way as in figure A.7.

We underline here a common problem in signal processing : how to infer about stochastic process from a single realisation? This point will be addressed in section A.4 (page 273).

The stochastic process shown figure A.7 is a really famous stochastic process: the Brownian motion. Its name comes from the biologist Robert Brown that was the first to observe it in the movement of particles inside a fluid in 1827. In 1905, Albert Einstein describes quantitatively this random movement of particles and links it with the diffusion equation (Einstein (1905), §4). The formal construction of Brownian motion as a stochastic process was done by Norbert Wiener in 1923. Hence the Brownian motion is also called a Wiener process. Its formal definition is the following:

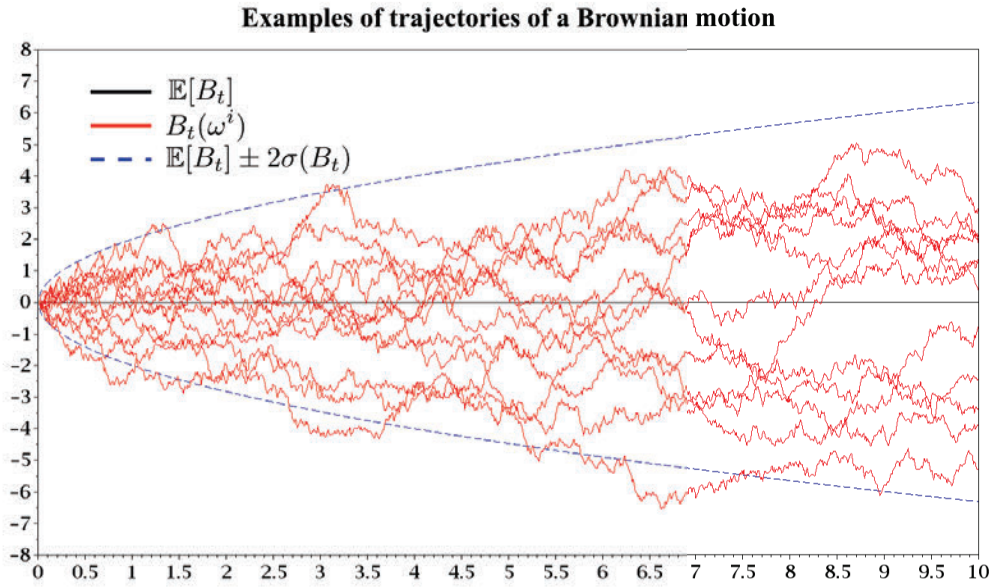


Figure A.7 – Example of stochastic process (Brownian motion): its average value (black) and its standard variation (dashed blue) give hints on the behaviour of realizations (red).

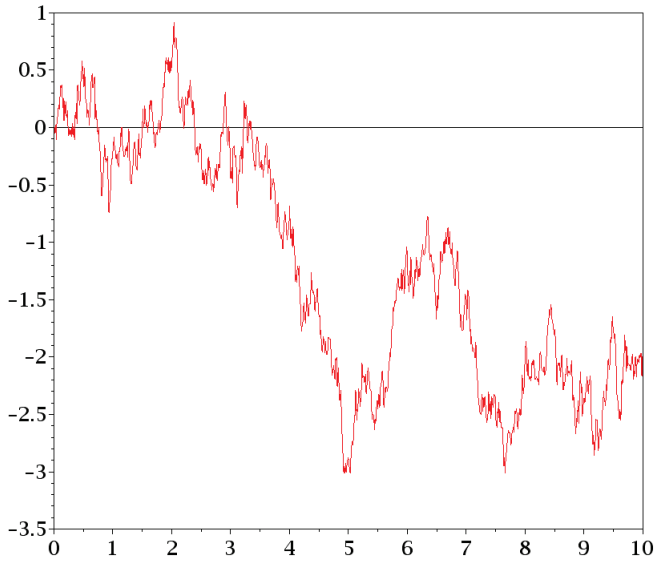


Figure A.8 – Same stochastic process (Brownian motion) with only its average value (black) and one realization (red).

Definition A.13 (Brownian motion)**.**

With words *We call a Brownian motion a Gaussian process $(B_t)_{t\in[0,T]}$ with mean zero and with covariance equal to the smallest time.*

Formally *Given $T > 0$,*

$$\forall t, s \in [0,T],\ \mathbb{E}[B_t] = \mathbb{E}[B_s] = 0,\ \mathrm{cov}(B_t, B_s) = \min(s,t)$$

where a *Gaussian process* is

Definition A.14 (Gaussian process)**.**

With words *A stochastic process $(X_t)_{t\in[0,T]}$ is Gaussian when every sample of it form a Gaussian vector (i.e. all linear combination of samples is Gaussian).*

Formally

$$\forall n \in \mathbb{N},\ \forall t_1, ..., t_n \in [0,T],\ \forall \theta \in \mathbb{R}^n,\ \sum_{i=1}^{n} \theta_i X_{t_i} \text{ is Gaussian}$$

In particular we have $B_t \rightsquigarrow \mathcal{N}(0, |t|)$ Hence its mean, variance and auto-covariance are given by

- $\mathbb{E}[B_t] = 0$
- $V(B_t) = |t|$
- $\mathrm{cov}(B_t, B_s) = \min(t, s)$

A.3.2 The Langevin equation

Another example is the Ornstein-Uhlenbeck process. It is the solution of the Langevin equation, which was historically made to describe the movement of a "large particle" dropped in a viscous fluid.

$$dV_t = -\frac{1}{\tau} V_t dt + \sigma dB_t \tag{A.18}$$

The equation A.18 is of different nature than usual partial differential equation: it is a stochastic differential equation (i.e. a differential equation on stochastic processes). Fundations of stochastic differential equations can be found in any course on stochastic calculus, and also in the book of Øksendal (Øksendal, 2013) and Gardiner (Gardiner, 2009). Stochastic differential equation relies on Ito integral (integral of a stochastic process along a stochastic process). The complete form of equation A.18 using Ito integral is

$$V_t - V_0 = \int_0^t -\frac{1}{\tau} V_s ds + \int_0^t \sigma dB_s \tag{A.19}$$

but the "differential" form is often preferred because it is shorter.

Using several theorem based on Ito integral, it is possible to solve the equation A.18 (see Gardiner (2009) p.103, Øksendal (2013) p. 71).

$$V_t = V_0 e^{-\frac{t}{\tau}} + \sigma \int_0^t e^{-\frac{t-s}{\tau}} dB_s \tag{A.20}$$

In particular, it gives an analytic expression of the mean, the variance and the auto-covariance of the solution process (Gardiner (2009) p. 104):

- $\mathbb{E}[V_t] = \mathbb{E}[V_0]\, e^{-\frac{t}{\tau}}$
- $V(V_t) = \sigma^2 \frac{\tau}{2} + \left(V(V_0) - \sigma^2 \frac{\tau}{2} \right) e^{-2\frac{t}{\tau}}$
- $\mathrm{cov}(V_t, V_s) = \left(V(V_0) - \sigma^2 \frac{\tau}{2} \right) e^{-\frac{t+s}{\tau}} + \sigma^2 \frac{\tau}{2} e^{\frac{t-s}{\tau}}$

Numerically, it is possible to simulate some trajectories (see figure A.9). One can see in

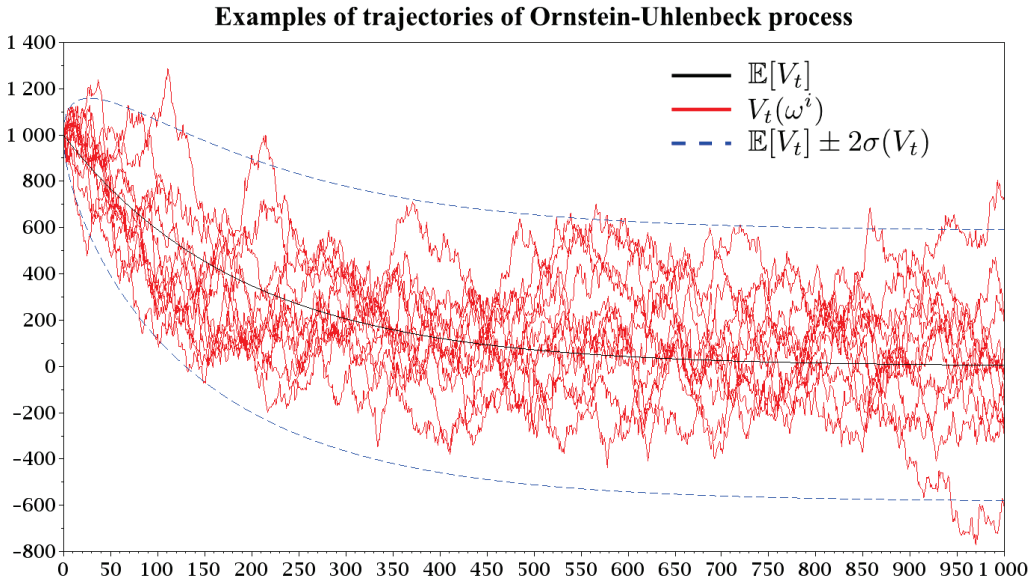


Figure A.9 – Same Ornstein-Uhlenbeck process (solution of Langevin equation): average (black) framed by standard variation (dashed blue) and few realizations.

figure A.9 that the behaviour of Ornstein-Uhlenbeck process is really different from the Brownian motion. While the average of Brownian motion is always equal to zero, the average of Ornstein-Uhlenbeck process is a converging exponential. While Brownian motion is infinitely expanding, the variance of Ornstein-Uhlenbeck process converges toward a constant value.

Physically, Langevin equation is a model for the movement of a "large particle" dropped in a viscous fluid. From figure A.9, we can tell that the particle will globally slow down in a first phase (until time-step 400) and get stabilized around a constant speed in a second phase.

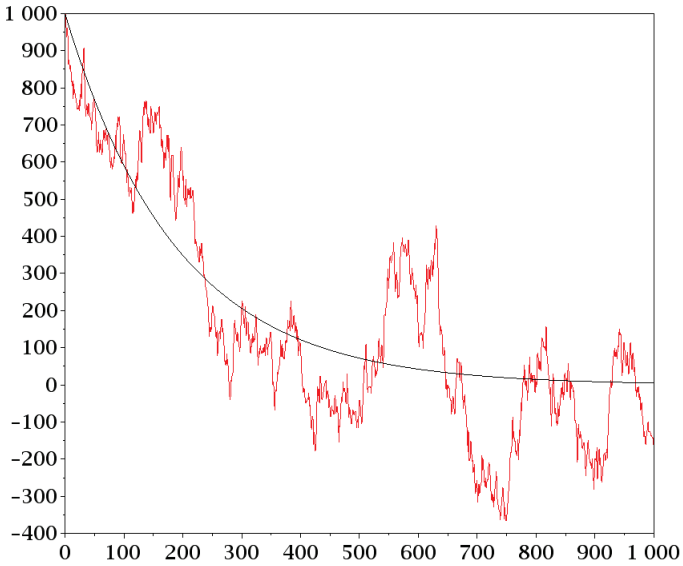


Figure A.10 – The Ornstein-Uhlenbeck process with only its average value (black) and one realization (red).

In this case, a single realization (as in figure A.10) will conserve the most important features of global process: first phase of decreasing, second phase around the average. The single realisation is more informative about the whole process.

A.4 Signal processing

Wind measurements are stochastic processes (because of random turbulence and random instrumental noise). What the instrument provides is a single realisation of the stochastic process "wind". This section aims to clarify how the realisation helps to know the stochastic process from which it is produced.

A.4.1 Stationarity and ergodicity

With time series of a single realization, we can compute statistics to describe the probability law of the signal. We consider only real-valued signals indexed by $t \in [0, T]$ with $T > 0$. The first assumption to do so is that the probability law of the signal does not change while we compute the statistics. This assumption is called **stationarity** :

Definition A.15 (Stationarity)**.**

With words *A signal is said stationary if the joint law of a time-sample does not change by translation in time*

Formally

$$\forall d \in \mathbb{N},\, \forall t_1, ..., t_d,\, \forall \tau,\, Law(X_{t_1}, ..., X_{t_d}) = Law(X_{t_1+\tau}, ..., X_{t_d+\tau})$$

This definition is stronger than just "the probability law of the signal does not change with time". Indeed, if a signal is stationary in the sense given by definition A.15, then $\forall t_1, t_2,\, Law(X_{t_1}) = Law(X_{t_2})$ (if we choose $\tau = t_2 - t_1$). So it is correct to say that if a signal is stationary, its probability law does not change with time. But the definition A.15 avoid also other problematic cases.

A weaker definition of stationarity is limited to the invariance of the p first moments (from Priestley (1981), def. 3.2.2).

Definition A.16 (Stationarity at order p)**.**

With words *A signal is said stationary at order p if the momenta up to p of the joint law of a time-sample does not change by translation in time.*

Formally

$$\forall d \in \mathbb{N},\, \forall t_1, ..., t_d,\, \forall \tau,\, \forall k \leqslant p,\, \mathbb{E}\left[(X_{t_1}^k, ..., X_{t_d}^k)\right] = \mathbb{E}\left[(X_{t_1+\tau}^k, ..., X_{t_d+\tau}^k)\right]$$

where $\mathbb{E}[\cdot]$ refers to the joint law.

In particular, the stationarity at order p ensures that the momenta only depends on the time

difference. For example, $\text{cov}(X_{t_1}, X_{t_2})$ depends only on $|t_1 - t_2|$.

$$V((X_{t_1}, X_{t_2})) = \begin{pmatrix} V(X_{t_1}) & \text{cov}(X_{t_1}, X_{t_2}) \\ \text{cov}(X_{t_1}, X_{t_2}) & V(X_{t_1}) \end{pmatrix} = V((X_0, X_{t_2 - t_1}))$$

if one takes $t_2 > t_1$ and $\tau = t_1$ in the definition.

As Gaussian processes are naturally very widespread, the process stationary at order 2 are common. Hence, it is sometimes said **weak-sense stationary**, which refers to process stationary at order 2. This definition is sufficient for most of applications.

Some applications (like kriging) do not require weak stationarity but only stationarity on the increments of the process. Such class of process are called **intrinsic**.

Definition A.17 (Intrinsic process)**.**

With words *A random signal X_t is intrinsic when its increments are centred with a variance depending only on the difference of time.*

Formally *X_t is intrinsic* $\iff$

- $\forall t, s,\ \mathbb{E}\left[X_{t+s} - X_t\right] = 0$
- $\exists \gamma : [0, +\infty[\to \mathbb{R},\ \forall t, s,\ V(X_{t+s} - X_t) = 2\gamma(s)$

Intrinsic processes are not necessarily stationary (not even at order one). For example, the Brownian motion is intrinsic but it is not stationary, as we will see later. Conversely, processes stationary at order 2 are intrinsic. Intrinsic process is the minimum property to introduce the tools that are commonly used in signal processing, such as:

- The mean :
$$\mu_X = \mathbb{E}\left[X_t\right]$$
- Autocovariance :
$$C_X(s) = \text{cov}(X_{t+s}, X_t)$$
- Autocorrelation :
$$\rho_X(s) = \frac{\text{cov}(X_{t+s}, X_t)}{\sqrt{V(X_t)\, V(X_{t+s})}}$$
- Variogram :
$$\gamma_X(s) = \frac{1}{2} V(X_{t+s} - X_t)$$

Autocovariance, autocorrelation and variogram are related in the case of stationary process at order 2. In this case, the mean μ_X and the variance σ_X^2 are constant.

$$V(X_{t+s} - X_t) = V(X_{t+s}) + V(X_t) - 2\text{cov}(X_{t+s}, X_t) = 2\sigma_X^2 - 2\text{cov}(X_{t+s}, X_t) \qquad \text{(A.21)}$$

Hence

$$\gamma_X(s) = \sigma_X^2 - C_X(s) \tag{A.22}$$

These probabilistic parameters are interesting because they give an intuitive interpretation of the process. Intuitively, the larger is s, the less X_{t+s} is linked to X_t. Hence, the form of C_X give an information about the "memory" of the process. A process with "long range memory" (i.e. with C_X decreasing slowly) does not need to be observed very often, because it is predictable. A process with "short range memory" (i.e. with C_X decreasing quickly) requires many observations to be described.

Stationarity is useful to define statistics invariant with time and thus common for a whole time series. But it is not enough to estimate these statistics from a single realization $x(t)$ of X_t. Such a realisation must be representative of the whole process: explore the same area of values, have the same variability. This property is called **ergodicity**. More precisely, under which condition is the time average is comparable to the expected value?

Let X_t be a stochastic process stationary at order 2. The time average is the stochastic process given by

$$M_\tau = \frac{1}{\tau}\int_0^\tau X_t dt \tag{A.23}$$

Since X_t is stationary at order 2, its average is constant and its auto-covariance depends only on the time gap ($\mathbb{E}[X_t] = \mu_X$ and $\mathbb{E}[X_{t_1}X_{t_2}] - \mu_X^2 = C_X(t_2 - t_1)$). Mean and variance of M_τ are expressed with these quantities.

$$\mathbb{E}[M_\tau] = \frac{1}{\tau}\int_0^\tau \mathbb{E}[X_t]\, dt = \mu_X \tag{A.24}$$

$$\begin{aligned}
V(M_\tau) &= \mathbb{E}\left[\left(\int_0^\tau X_t dt\right)^2\right] - \mu_X^2 \\
&= \frac{1}{\tau^2}\mathbb{E}\left[\int_{[0,\tau]^2} X_{t_1}X_{t_2} dt_1 dt_2\right] - \mu_X^2 \\
&= \frac{1}{\tau^2}\int_{[0,\tau]^2} C_X(t_2 - t_1) dt_1 dt_2 \\
&= \frac{1}{\tau^2}\int_{-\tau}^{\tau} dt \int_{-\tau}^{\tau} C_X(s) ds \qquad \text{with } s = t_2 - t_1, t = t_1 \\
&= \frac{2}{\tau}\int_0^\tau C_X(s) ds \qquad \text{because } C_X \text{ is even}
\end{aligned} \tag{A.25}$$

The time average M_τ (stochastic process) will be comparable to the expectation μ_X (scalar) if its variance tends to 0. This requires $\int_0^\tau C_X(s)ds = o\left(\frac{1}{\tau}\right)$. In particular, having $\int_0^{+\infty} C_X(s)ds < \infty$ is enough. Under this condition, the time average approximates correctly the expectation.

$$\text{For any } x(t) \rightsquigarrow X_t, \qquad \lim_{\tau\to\infty}\frac{1}{\tau}\int_0^\tau x(t)dt = \mu_X \tag{A.26}$$

For stochastic process less regular than stationary at order 2, it is still possible to define ergodicity but the criteria are readable. Let X_t be a non-stationary process, one can define the measure corresponding to the time average:

$$\nu_\tau : A \in \mathcal{E} \mapsto \frac{1}{\tau}\int_0^\tau \mathbb{P}\left(X_t \in A\right) dt \tag{A.27}$$

$\nu_\tau(A)$ is the ratio between the time X_t has spent in A versus the total time τ. If the random variable M_τ follows the law ν_τ, its expectation is given by $\mathbb{E}\left[M_\tau\right] = \int_E x\nu_\tau(dx) = \frac{1}{\tau}\int_0^\tau \int_E x\mathbb{P}\left(X_t \in dx\right) dt = \frac{1}{\tau}\int_0^\tau \mathbb{E}\left[X_t\right] dt$. Its variance is given by $V(M_\tau) = \frac{1}{\tau}\int_0^\tau V(X_t) + \frac{1}{\tau}\int_0^\tau \mathbb{E}\left[X_t\right]^2 - \frac{1}{\tau^2}\left(\int_0^\tau \mathbb{E}\left[X_t\right]\right)^2$. When it converges, the limit of ν_τ when $\tau \to +\infty$ is the equilibrium measure ν. A random variable following ν would be a "infinite time average". But the link between the expectation of the equilibrium measure and the expectation of X_t is subject to additional hypothesis. In consequence this manuscript will only focus on processes stationary at order 2, as ergodicity is not the core of the problem.

Definition A.18 (Ergodicity)**.**

With words *A random signal X_t is ergodic when its mathematical expectation can be approached by a time average. In particular, processes stationary at order 2 are ergodic their auto-covariance decreases faster than $1/\tau$.*

Formally

$$\lim_{\tau\to\infty} \mathbb{E}\left[\frac{1}{\tau}\int_0^\tau X_t dt\right] = \mu_X \text{ and } \lim_{\tau\to\infty} V\left(\frac{1}{\tau}\int_0^\tau X_t dt\right) = 0$$

If X_t is stationary at order 2, it is equivalent to

$$\int_0^\tau C_X(s) ds = o\left(\frac{1}{\tau}\right)$$

There is thus a distinction to make between theoretical quantities and reachable approximation of them. The link between them relies on these 2 strong assumptions : stationarity and ergodicity. Although there are strong, we usually assume they are verified because it is the only way to make signal processing. Theoretical and approached mean and auto-covariance are summarized in table A.6.

	Theoretical	**Approached**
Mean	$\mu_X = \mathbb{E}\left[X_t\right]$	$m_x(\tau) = \frac{1}{\tau}\int_0^\tau x(t)dt$
Auto-covariance	$C_X(s) + \mu_X^2 = \mathbb{E}\left[X_t X_{t+s}\right]$	$R_{xx}(\tau, s) = \frac{1}{\tau}\int_0^\tau x(t)x(t+s)dt$

Table A.6 – Theoretical and approached expression of mean and auto-covariance under stationarity and ergodicity assumption.

For stationary and ergodic processes, these statistics are the best approximation one can

have :

$$\mu_X = \lim_{\tau \to \infty} m_x(\tau) \tag{A.28}$$

$$C_X(s) + \mu_X^2 = \lim_{\tau \to \infty} R_{xx}(\tau, s) \tag{A.29}$$

Stationarity and ergodicity are the necessary hypothesis to infer properties of random process X_t from a single realization of it $x(t) = X_t(\omega)$. Since in most cases it is impossible to reproduce exactly the same experiment (especially in fluid dynamics), it is generally assumed that processes are stationary and ergodic, so it is possible to make signal processing on them.

The validity of these hypothesis depend strongly on the choice of the integration time τ. Indeed, a long integration time gives a better approximation of the expected value with time average (it ensures ergodicity). But a long integration time let the time for the signal probability law to change (it unvalidates stationnarity). On the other hand, a the signal probability law is more likely to stay the same within a short integration time (it ensures stationarity). But a short integration time deteriorates the approximation of mathematical expectation with time average (it unvalidates ergodicity).

A.4.2 Fourier transform

The Fourier transform is the basis of many interpretations in signal processing, as it makes the connection between the time-space and the frequency-space. At first, it is defined only for periodic functions, that we decompose in pure harmonics : it yields to Fourier series. The periodic function is described by a collection of coefficients that quantify the contribution of a frequency in the signal. Fourier transform is an extension of that to any integrable function. The integrable function is described by another function, that gives the continuous contribution of frequencies in the signal. The mean value is not important for spectral analysis, therefore the process X_t is supposed centred for this section.

Definition A.19 (Fourier transform)**.**

With words *The Fourier transform a function of time $x(t)$ is the decomposition of the signal onto a basis of purely oscillating functions. The result is a function of frequency $\hat{x}(\xi)$ giving the contribution of the frequency ξ in the original signal.*

Formally

$$\forall x \in L^1(\mathbb{R}, \mathbb{R}),\ \hat{x}(\xi) = \int_{-\infty}^{+\infty} x(t) e^{-i\xi \cdot t} dt$$

But most of the time, we use square-integrable function (function in L^2, not necessarily in L^1). The integral in the given definition of Fourier transform is not defined for L^2 functions. Fourier transform is extended to L^2 space by using the density of $L^1 \cap L^2$ in L^2 and the

continuity of the integral. On L^2, Fourier transform is a linear operator can conserve most of the properties of its L^1 equivalent.

Given a realisation $x(t)$ of the stochastic process X_t, the function $x : t \in [0, +\infty[\mapsto \mathbb{R}$ is assumed $L^1([0, +\infty[, \mathbb{R})$ and $L^2([0, +\infty[, \mathbb{R})$. The signal $x(t)$ is prolonged for $t < 0$ by 0.

A.4.3 Power spectral density

To visualize the frequency information brought by Fourier transform, we use the power spectral density. It represents the energetic contribution of frequencies.

Definition A.20 (Power spectral density)**.**

With words *Power spectral density is the Fourier transform of the autocorrelation.*

Formally

$$\Gamma_x(\xi) = \int_{s\in\mathbb{R}} \mathbb{E}\left[X_{s-t}X_t\right] e^{-i\xi\cdot s} ds$$

This definition is applied to ergodic and stationary processes to reach a computable expression. The auto-covariance is approximated as described in table A.6 :

$$\mathbb{E}\left[X_{s-t}X_t\right] = \lim_{\tau\to\infty} \frac{1}{\tau} \int_0^\tau x(t)x(s-t)dt \tag{A.30}$$

Wiener-Khinchin theorem gives another expression of the PSD, sometimes taken as definition (see Miller and Childers (2012), def. 10.1). Here is a sketch of the proof.

$$\begin{aligned}
\Gamma_x(\xi) &= \int_{s\in\mathbb{R}} \left(\lim_{\tau\to\infty} \frac{1}{\tau} \int_0^\tau x(t)x(s-t)dt \right) e^{-i\xi\cdot s} ds \\
&= \lim_{\tau\to\infty} \frac{1}{\tau} \int_{s\in\mathbb{R}} \int_0^\tau x(t)x(s-t)e^{-i\xi s} dt ds \\
&= \lim_{\tau\to\infty} \frac{1}{\tau} \int_0^\tau x(t) \underbrace{\left(\int_{s\in\mathbb{R}} x(s-t)e^{-i\xi s} ds \right)}_{=\hat{x}(\xi)e^{-i\xi t}} dt \\
&= \hat{x}(\xi) \lim_{\tau\to\infty} \frac{1}{\tau} \int_0^\tau x(t)e^{-i\xi t} dt \\
&\simeq \frac{|\hat{x}(\xi)|^2}{\tau_{max}}
\end{aligned}$$

Thus the power spectral density is estimated by the squared modulus of the Fourier transform divide by the sample length τ_{max}.

A.4.4 Examples

We will now illustrate these notions on the examples of stochastic processes seen before.

Brownian motion (definition A.13)

- $B_t \rightsquigarrow \mathcal{N}(0, |t|)$
- $\mathbb{E}[B_t] = 0$
- $V(B_t) = |t|$
- $\text{cov}(B_s, B_t) = \min(s, t)$

The Brownian motion is stationary at order 1 because its average is always 0. It is also intrinsic. But it is not stationary at order 2 because its variance depends on t (thus it is not stationary in strong sense, neither stationary in weak sense). As a consequence, the relation A.22 ($\gamma = \sigma^2 - C$) does not hold.

- The mean :
$$\mu_B = 0$$
- Autocovariance :
$$C_B(s) = \text{cov}(B_{t+s}, B_t) = \min(t + s, t) = t$$
- Autocorrelation :
$$\rho_B(s) = \frac{\text{cov}(B_{t+s}, B_t)}{\sqrt{V(B_{t+s})\, V(B_t)}} = \sqrt{\frac{t}{t+s}}$$
- Variogram :
$$\gamma_B(s) = |s|$$

The Brownian motion is not ergodic. A famous property of Brownian motion is that it is dilating to infinity (it reaches every point of $\mathbb{R}$ in finite time):

$$\text{Almost surely, } \limsup_{t \to +\infty} B_t = +\infty \text{ and } \liminf_{t \to +\infty} B_t = -\infty \tag{A.31}$$

Hence the function $\tau \mapsto \frac{1}{\tau} \int_{t_0}^{t_0+\tau} b(t)dt$ does not converge for almost every realization $b(t)$ of B_t. As the Brownian motion is not stationary at order 2, its auto-covariance depends on the time gap s and the starting time t. Thus, its spectrum cannot be defined the same way as definition A.20.

To conclude, the Brownian motion is not stationary nor ergodic. Since these properties are essential for a realization to be informative about the whole process, it explains why the single realization shown on figure A.8 is not informative about the Brownian motion. However it is intrinsic with a linear variogram.

Langevin equation (solution of equation A.18)

- $V_t = V_0 e^{-\frac{t}{\tau}} + \sigma \int_0^t e^{-\frac{t-s}{\tau}} dB_s$
- $\mathbb{E}[V_t] = \mathbb{E}[V_0] e^{-\frac{t}{\tau}}$
- $V(V_t) = \sigma^2 \frac{\tau}{2} + \left(V(V_0) - \sigma^2 \frac{\tau}{2} \right) e^{-2\frac{t}{\tau}}$
- $\text{cov}(V_{t+s}, V_t) = \left(V(V_0) - \sigma^2 \frac{\tau}{2} \right) e^{-\frac{2t+s}{\tau}} + \sigma^2 \frac{\tau}{2} e^{-\frac{s}{\tau}}$

The Ornstein-Ulhenbeck process (solution of Langevin equation) is not stationary, because both its mean and its variance depend on time. But, after few τ, both mean and variance reach a limit:

$$\mathbb{E}[V_\infty] = 0$$

$$V(V_\infty) = \sigma^2 \frac{\tau}{2}$$

This limit is called the *stationary solution* of Langevin equation. According to Gardiner (Gardiner (2009), p.74), the stationary solution is Gaussian

$$V_\infty \rightsquigarrow \mathcal{N}\left(0, \sigma^2 \frac{\tau}{2}\right)$$

Hence the stationary solution is order 2 stationary and intrinsic. Asymptotically when $t \to +\infty$, it gives the following expressions:

- The mean :
$$\mu_V = 0$$

- Autocovariance :
$$C_V(s) = \text{cov}(V_{t+s}, V_t) = \sigma^2 \frac{\tau}{2} e^{-\frac{s}{\tau}}$$

- Autocorrelation :
$$\rho_V(s) = \frac{\text{cov}(V_{t+s}, V_t)}{V(V_\infty)} = e^{-\frac{s}{\tau}}$$

- Variogram :
$$\gamma_V(s) = \sigma^2 \frac{\tau}{2} \left(1 - e^{-\frac{s}{\tau}}\right)$$

Exponential decrease of autocovariance is the example take by Gardiner for a sufficient condition for the process to be ergodic (Gardiner (2009), p. 58). Hence, the Ornstein-Ulhenbeck process is asymptotically ergodic.

Concerning its spectrum, the Fourier transform of its auto-covariance (which is an expo-

nential) is a Lorenzian function :

$$\begin{aligned}\Gamma_V(\xi) &= \int_{s\in\mathbb{R}} \sigma^2\frac{\tau}{2}e^{-\frac{s}{\tau}}e^{-i\xi\cdot s}ds \\ &= \sigma^2\frac{\tau}{2}\int_{s\in\mathbb{R}} e^{-\frac{s}{\tau}-i\xi\cdot s}ds \\ &= \frac{\sigma^2\frac{\tau}{2}}{1+4\pi^2\tau^2\xi^2}\end{aligned}$$

To conclude, the Ornstein-Ulhenbeck process is asymptotically stationary of order 2 and ergodic. Its spectrum decrease with a -2 slope in a log-log scale (usual for PSDs). Because of these properties the single realization shown on figure A.10 is informative about the whole process.

A.5 Conclusion

This appendix intended to refresh some basics and to set the notations and the definitions as they will be used in this document. The first section was giving the essential tools to understand what a random variable is. The second section focused the numerical generation of random variables. The third section introduced stochastic processes with two examples: the Brownian motion and the Ornstein-Ulhenbeck process. The fourth and last section made the link between the theoretical stochastic processes and the empirical time series of one realisation.

Random variables are everywhere in turbulence, in filtering and in sensitivity analysis. A random variable is a function from an unknown departure point (usually denoted ω). To represent this "unknown dimension", one makes samples of the same random variable. Such samples are generated on computer using pseudo-random numbers generators. Pseudo-random numbers generators of two programming languages have been tested. The algorithm to generate any random variable from a uniformly distributed sample has been explained. In particular, the algorithm A.1 for the generation of a sample distributed along a given potential is the key element of the filtering process.

Stochastic processes are families of random variables. As measurements are usually composed of time series, stochastic process are relevant to model them. Two examples are given: the Brownian motion and the Ornstein-Ulhenbeck process. The Brownian motion describes the diffusing movement of a particle inside a fluid. In average, the movement is null (there is no trend). But the variance steadily increases: the particle goes farer and farer but always comes back. The Ornstein-Ulhenbeck process is the solution of the Langevin equation. This equation is a simple model for a particle drop into a fluid. Conversely to the Brownian motion, there is a trend and the variance is bounded.

Signal processing can be used in many contexts. The small section with this name in this chapter intents to distinguish theoretical signals (stochastic process) from accessible signals

(a single realisation of the stochastic process). Quantities of importance for applications (average, variance, auto-covariance, variogram, spectrum) are introduced for both. It is shown are they can be compared under the assumptions of stationarity and ergodicity. They are illustrated on the two stochastic processes given in examples.

Appendix B

Proofs

B.1 Formulae on reconstruction outputs

B.1.1 Influence of N on N_{G0}

Theorem B.1 (Influence of N on N_{G0})**.** *If $N > 1$ is the number of particles and at any time there are particles in the probe volume, for any $n \in \mathbb{N}$, there exist $\alpha(n) > 0$ and $\beta(n) > 0$ such that*

$$\mathbb{P}\left(N_{G0} = n\right) \leqslant \alpha(n) e^{-N/\beta(n)} \tag{B.1}$$

Proof. A result of Del Moral (2004) (theorem 7.4.1, page 232) quoted by Baehr (2010) (theorem 2.2) on the degeneracy of the particle filter is useful to reckon the influence of N on N_{G0}. Here is the theorem written with the notation of this manuscript.

> If at any time there are particles in the probe volume , then $\forall N \geqslant 1$ and $\forall t \geqslant 0$, the time τ^N at which potential is null for all particles follows the inequality:
>
> $$\mathbb{P}\left(\tau^N \leqslant t\right) \leqslant a(t) e^{-N/b(t)} \tag{B.2}$$

When the potential vanishes, the system is re-initialized. Thus, the N_{G0} extinctions are independent and the time counter can be reset to zero. If the k-th extinction time is denoted τ_k^N and compared to the time t_k (reset to zero after each extinction), the number of null potential is written

$$\mathbb{P}\left(N_{G0} = n\right) = \prod_{k=1}^{n} \mathbb{P}\left(\tau_k^N \leqslant t_k\right) \leqslant \prod_{k=1}^{n} a(t_k) e^{-N/b(t_k)}$$

with the condition $\sum_{k=1}^{n} t_k \leqslant N_t$.

It tells there exist $\alpha(n) = \prod_{k=1}^{n} a(t_k)$ and $\beta(n) = \left(\sum_{k=1}^{n} 1/b(t_k)\right)^{-1}$ such that

$$\mathbb{P}\left(N_{G0} = n\right) \leqslant \alpha(n) e^{-N/\beta(n)} \tag{B.3}$$

Thus, the number of null potential decreases exponentially with N. □

B.1.2 Influence of σ^{obs} and σ^{add} on N_{G0}

Theorem B.2 (Influence of σ^{obs} and σ^{add} on N_{G0})**.** *If the following assumptions are satisfied*

- *The real wind $V^{\mathbf{r}}_{z,t}$ is stationary at order 2 and ergodic in space and time.*
- *The particles after conditioning $\tilde{V}_t$ are Gaussian with the same mean and variance as $V^{\mathbf{r}}_{z,t}$.*

Then, the average number of null potential N_{G0} is bounded from above by a function of σ^{obs} and σ^{add}.

$$\mathbb{E}\left[N_{G0}\right] \leqslant N_t N_z \frac{(\sigma^{add})^2 + 2k}{-\big((\sigma^{obs})^2 + 2k\big)\log\Big(\iota^2 2\pi\big((\sigma^{obs})^2 + 2k\big)\Big)} \tag{B.4}$$

with $\iota = 10^{-16}$, the zero machine threshold, and $k = \frac{1}{2}V\left(V^{\mathbf{r}}_{z,t}\right)$ (constant thanks to the stationarity assumption).

Proof. The number of null potential is defined as

$$N_{G0} = \left|\left\{(z,t)/\sum_{i=1}^{N} G_{obs}(z,t,i) = 0\right\}\right|$$

with $G_{obs}(z,t,i) = \exp\left(-\frac{\left(\tilde{V}_t^i - V^{\mathbf{o}}(z,t)\right)^2}{2(\sigma^{obs})^2}\right)$. The potential G_{obs} can be seen as a function of two random variables: $\tilde{V}_t^i$ (as a realisation of the random variable $\tilde{V}_t$) and $V^{\mathbf{o}}(z,t)$ (as a realisation of the random variable $V^{\mathbf{o}}_{z,t}$). Let us consider the random variable G:

$$G = \exp\left(-\frac{\left(\tilde{V}_t - V^{\mathbf{o}}_{z,t}\right)^2}{2(\sigma^{obs})^2}\right) \tag{B.5}$$

The sum on the N particles can be seen as the Monte Carlo approximation of the expectation along $\tilde{V}_t$. The observation is considered as a data, thus it is a conditional expectation:

$$\mathbb{E}\left[G|V^{\mathbf{o}}_{z,t}\right] \text{ estimated by } \sum_{i=1}^{N} G_{obs}(z,t,i)$$

If we consider the discrete random variable $T_{z,t}$ such that

$$T_{z,t} = \mathbf{1}_{\mathbb{E}\left[G|V^{\mathbf{o}}_{z,t}\right]=0} = \begin{cases} 1 \text{ if } \mathbb{E}\left[G|V^{\mathbf{o}}_{z,t}\right] = 0 \\ 0 \text{ else} \end{cases}$$

Then, $T_{z,t}$ follows a Bernoulli law with parameter $p_{z,t} = \mathbb{P}\left(\mathbb{E}\left[G|V^{\mathbf{o}}_{z,t}\right] = 0\right)$. And N_{G0} is thus the random variable equal to the sum of all $T_{z,t}$, for $z \in [\![1, N_z]\!]$ and $t \in [\![1, N_t]\!]$.

$$N_{G0} = \sum_{z,t} T_{z,t}$$

If the parameter $p_{z,t}$ were constant, N_{G0} would have follow a binomial law, but it is not

the case. Instead, we will find a bound for $p_{z,t}$ which depends on σ^{obs} and σ^{add} but not on z and t.

First, we express $\mathbb{E}\left[G|V^{\mathbf{o}}_{z,t}\right]$. To do so, we use the assumption that $\tilde{V}_t$ is Gaussian, of mean $\tilde{\mu}$ and variance $\tilde{\sigma}^2$.

$$\mathbb{E}\left[G|V^{\mathbf{o}}_{z,t}\right] = \frac{1}{\sqrt{2\pi}\tilde{\sigma}} \int \exp\left(-\frac{\left(x - V^{\mathbf{o}}_{z,t}\right)^2}{2(\sigma^{obs})^2}\right) \exp\left(-\frac{(x-\tilde{\mu})^2}{2\tilde{\sigma}^2}\right) dx$$

From Bromiley (2003), the product of these two Gaussian functions is expressed, and it yields to

$$\mathbb{E}\left[G|V^{\mathbf{o}}_{z,t}\right] = \frac{S(V^{\mathbf{o}}_{z,t})}{\sqrt{2\pi}\sigma} \int \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) dx = S(V^{\mathbf{o}}_{z,t})$$

with

$$\frac{1}{\sigma^2} = \frac{1}{(\sigma^{obs})^2} + \frac{1}{\tilde{\sigma}^2}\ , \quad \mu = \left(\frac{\tilde{\mu}}{\tilde{\sigma}^2} + \frac{V^{\mathbf{o}}_{z,t}}{(\sigma^{obs})^2}\right)\sigma^2 \quad \text{and} \quad S(V^{\mathbf{o}}_{z,t}) = \frac{1}{\sqrt{2\pi}\frac{\tilde{\sigma}\sigma^{obs}}{\sigma}} \exp\left(-\frac{\sigma^2}{2}\frac{(V^{\mathbf{o}}_{z,t} - \tilde{\mu})^2}{\tilde{\sigma}^2(\sigma^{obs})^2}\right)$$

Solving the equation $\mathbb{E}\left[G|V^{\mathbf{o}}_{z,t}\right] = 0$ yields to $S(V^{\mathbf{o}}_{z,t}) = 0$, which occurs when $|V^{\mathbf{o}}_{z,t} - \tilde{\mu}| = +\infty$. However, because of machine threshold, the equality $\mathbb{E}\left[G|V^{\mathbf{o}}_{z,t}\right] = 0$ is in reality equivalent to $\mathbb{E}\left[G|V^{\mathbf{o}}_{z,t}\right] < \iota$ (where $\iota = 10^{-16}$), which yields to the condition $|V^{\mathbf{o}}_{z,t} - \tilde{\mu}| > \beta(\sigma^{obs})$. Then, we know that theoretically $V^{\mathbf{o}}_{z,t} \rightsquigarrow \mathcal{N}\left(V^{\mathbf{r}}(z,t), (\sigma^{add})^2\right)$ and $\tilde{\mu} = \mathbb{E}\left[V^{\mathbf{r}}_{z,t}\right]$ by assumption. Hence, from Tchebychev inequality, we have

$$\mathbb{P}\left(|V^{\mathbf{o}}_{z,t} - \tilde{\mu}| > \beta(\sigma^{obs})\right) \leqslant \frac{1}{\beta(\sigma^{obs})^2}\left((\sigma^{add})^2 + (V^{\mathbf{r}}(z,t) - \mathbb{E}\left[V^{\mathbf{r}}_{z,t}\right])^2\right)$$

with $\beta(\sigma^{obs}) = \sqrt{-(\tilde{\sigma}^2 + (\sigma^{obs})^2)\log\left(\iota^2 2\pi(\tilde{\sigma}^2 + (\sigma^{obs})^2)\right)}$. In this upper bound, two terms will be rewritten with the wind variance: $\tilde{\sigma}^2 = 2k$ (by assumption) and $(V^{\mathbf{r}}(z,t) - \mathbb{E}\left[V^{\mathbf{r}}_{z,t}\right])^2$. Moreover, since the process $V^{\mathbf{r}}_{z,t}$ is assumed stationary at order 2, the wind variance is a constant and equal to $2k$, as stated in the assumptions. Since $V^{\mathbf{r}}(z,t)$ is a realization of the process $V^{\mathbf{r}}_{z,t}$, the term $(V^{\mathbf{r}}(z,t) - \mathbb{E}\left[V^{\mathbf{r}}_{z,t}\right])^2$ is a realization of the variance estimator when the average is known. The term $(V^{\mathbf{r}}(z,t) - \mathbb{E}\left[V^{\mathbf{r}}_{z,t}\right])^2$ itself is different for each time step t or vertical level z. But since the process $V^{\mathbf{r}}_{z,t}$ is assumed ergodic in time and space, the time and space average $\frac{1}{N_z N_t}\sum_{z,t}(V^{\mathbf{r}}(z,t) - \mathbb{E}\left[V^{\mathbf{r}}_{z,t}\right])^2$ converges toward $V\left(V^{\mathbf{r}}_{z,t}\right)$ when

$N_z N_t \to +\infty$. Hence, the expectation of N_{G0} can be bounded as follows:

$$\begin{aligned}
\mathbb{E}\left[N_{G0}\right] &= \sum_{z,t} \mathbb{E}\left[T_{z,t}\right] \\
&= \sum_{z,t} \mathbb{P}\left(\mathbb{E}\left[G|V^{\mathbf{o}}_{z,t}\right] = 0\right) \\
&\leqslant \sum_{z,t} \frac{(\sigma^{add})^2 + (V^{\mathbf{r}}(z,t) - \mathbb{E}\left[V^{\mathbf{r}}_{z,t}\right])^2}{\beta(\sigma^{obs})^2} \\
&= \frac{1}{\beta(\sigma^{obs})^2} \left(N_t N_z (\sigma^{add})^2 + \underbrace{\sum_{z,t} (V^{\mathbf{r}}(z,t) - \mathbb{E}\left[V^{\mathbf{r}}_{z,t}\right])^2}_{=N_t N_z 2k} \right) \\
&= N_t N_z \frac{(\sigma^{add})^2 + 2k}{-\left((\sigma^{obs})^2 + 2k\right) \log\left(\iota^2 2\pi \left((\sigma^{obs})^2 + 2k\right)\right)}
\end{aligned}$$

which gives the result.

□

B.2 Theorems in sensitivity analysis

B.2.1 ANOVA decomposition

Theorem B.3 (ANOVA decomposition)**.** *Let $f : [0,1]^p \to \mathbb{R}$ be an integrable function. Previous notations hold.*
Then, there exists a unique decomposition of f :

$$\begin{aligned}
f(x) &= \sum_{\mathbf{u} \in I} f_{\mathbf{u}}(x_{\mathbf{u}}) \\
&= f_{\emptyset} + \sum_{i=1}^{p} f_i(x_i) + \sum_{1 \leqslant i < j \leqslant p} f_{ij}(x_i, x_j) + \cdots + f_{1,\ldots,p}(x_1, \ldots, x_p)
\end{aligned} \tag{B.6}$$

such that

$$\forall \mathbf{u} \in I, \forall i \in \mathbf{u}, \int_0^1 f_{\mathbf{u}}(x_{\mathbf{u}}) dx_i = 0 \tag{B.7}$$

The proof of this theorem has been first made by Sobol in Sobol (1976), using Fourier-Haar decompositions. Then a simpler version is in Sobol (1993). This is a proof adapted to our notations.

Proof. (From Sobol (1993, 1976))
First we prove existence, then uniqueness of the family $(f_{\mathbf{u}})_{\mathbf{u} \in I}$ which satisfies properties (B.6) et (B.7).

Existence

Let consider the following functions :

$$g_{\mathbf{u}} : \begin{array}{ccc} [0,1]^{|\mathbf{u}|} & \to & \mathbb{R} \\ x_{\mathbf{u}} & \mapsto & g_{\mathbf{u}}(x_{\mathbf{u}}) = \displaystyle\int_{[0,1]^{|\bar{\mathbf{u}}|}} f(x) dx_{\bar{\mathbf{u}}} \end{array}$$

with the convention $g_{[\![1,p]\!]}(x_1, \ldots, x_p) = f(x)$.

The functions $f_{\mathbf{u}}$ will be built recursively thanks to the functions g. At first, with the hands, in order to satisfy the second part of theorem (B.7).

$$f_\emptyset = \int_{[0,1]^p} f(x) d\mathbf{x} = g_\emptyset$$

- $\forall i \in [\![1,p]\!]$, $f_i(x_i) = \displaystyle\int_{[0,1]^{p-1}} f(x) dx_1 \ldots dx_{i-1} dx_{i+1} \ldots dx_p - f_\emptyset = g_i(x_i) - f_\emptyset$
- $\forall i,j \in [\![1,p]\!]$, $f_{ij}(x_i, x_j) = g_{ij}(x_i, x_j) - f_i(x_i) - f_j(x_j)$
- $\forall i,j,k \in [\![1,p]\!]$, $f_{ijk}(x_i, x_j, x_k) = g_{ijk}(x_i, x_j, x_j) - f_{ij}(x_i, x_j) - f_{ik}(x_i, x_k) - f_{jk}(x_j, x_k)$
- $\ldots$

It will be proven that the functions of the decomposition are

$$f_{\mathbf{u}} : \begin{array}{ccc} [0,1]^{|\mathbf{u}|} & \to & \mathbb{R} \\ x_{\mathbf{u}} & \mapsto & f_{\mathbf{u}}(x_{\mathbf{u}}) = g_{\mathbf{u}}(x_{\mathbf{u}}) - \displaystyle\sum_{i \in \mathbf{u}} f_{\mathbf{u}\setminus i}(x_{\mathbf{u}\setminus i}) \end{array}$$

One can express the functions $(g_{\mathbf{u}})_{\mathbf{u}\in I}$ with the functions of the decomposition:

$$g_{\mathbf{u}}(x_{\mathbf{u}}) = \sum_{\mathbf{v} \subset \mathbf{u}} f_{\mathbf{v}}(x_{\mathbf{v}})$$

And because $g_{[\![1,p]\!]}(x_1, \ldots, x_p) = f(x)$, one gets directly the decomposition 4.1.

$$f(x) = \sum_{\mathbf{u} \in I} f_{\mathbf{u}}(x_{\mathbf{u}})$$

The property B.7 of the decomposition functions is still to be proven. Recursively on the cardinal of $\mathbf{u}$:

$\underline{|\mathbf{u}| = 1}$ Let $\mathbf{u} = i$. Then,

$$\begin{array}{rcl} \int_0^1 f_i(x_i) dx_i & = & \int_0^1 g_i(x_i) dx_i - f_\emptyset \\ & = & \int_0^1 \int_{[0,1]^{|\mathbf{v}|}} f(x) d\mathbf{x}_{\mathbf{v}} dx_i - f_\emptyset \text{ with } v = [\![1,p]\!] \setminus i \\ & = & \int_{[0,1]^p} f(x) d\mathbf{x} - f_\emptyset \\ & = & 0 \end{array}$$

$\star$ $\underline{|\mathbf{u}| = k-1 \Rightarrow |\mathbf{u}| = k}$ Let $\mathbf{u} \in I$ of cardinal k. The following property is assumed true for all $n \leqslant k$:

$$\forall \mathbf{v},\ |\mathbf{v}| = n,\ \forall i \in \mathbf{v},\ \int_0^1 f_{\mathbf{v}}(\mathbf{x}_{\mathbf{v}}) dx_i = 0$$

.

Then this property is also true for $\mathbf{u}$. Let $j \in \mathbf{u}$.

$$\begin{aligned}\int_0^1 f_{\mathbf{u}}(x_{\mathbf{u}})dx_j &= \underbrace{\int_0^1 g_{\mathbf{u}}(x_{\mathbf{u}})dx_j}_{f_{\mathbf{u}\setminus j}(x_{\mathbf{u}\setminus j})} - \sum_{i\in\mathbf{u}} \underbrace{\int_0^1 f_{\mathbf{u}\setminus i}(x_{\mathbf{u}\setminus i})dx_j}_{\begin{cases} 0 & \text{if } i \neq j \\ f_{\mathbf{u}\setminus j}(x_{\mathbf{u}\setminus j}) & \text{if } i = j \end{cases}} \\ &= f_{\mathbf{u}\setminus j}(x_{\mathbf{u}\setminus j}) - f_{\mathbf{u}\setminus j}(x_{\mathbf{u}\setminus j}) = 0\end{aligned}$$

because $|\mathbf{u} \setminus i| = k - 1$ hence the property is true by assumption. The change of $g_{\mathbf{u}}$ come from Fubini's theorem.

A family of functions $f_{\mathbf{u}}$, $\mathbf{u} \in I$ satisfying the theorem has been found. It proves the existence of such a family.

Uniqueness

To prove its uniqueness, let $\left(\tilde{f}_{\mathbf{u}}\right)_{\mathbf{u}\in I}$ and $(f_{\mathbf{u}})_{\mathbf{u}\in I}$ be two families of functions such that properties (B.6) and (B.7) are verified.

It will be proven that $\forall \mathbf{u} \in I$, $f_{\mathbf{u}} = \tilde{f}_{\mathbf{u}}$ recursively on the cardinal of $\mathbf{u}$.

$\underline{|\mathbf{u}| = 0}$ By integration of (B.6) with respect to $(x_1, \ldots, x_p)$:

$$\int_{[0,1]^p} f(x)d\mathbf{x} = f_\emptyset = \tilde{f}_\emptyset$$

because $\forall \mathbf{u} \neq \emptyset$, $\exists i \in [\![1, p]\!]$, $i \in \mathbf{u}$ so $\int_{[0,1]^p} f_{\mathbf{u}}(x_{\mathbf{u}})d\mathbf{x} = \int_{[0,1]^p} \tilde{f}_{\mathbf{u}}(x_{\mathbf{u}})d\mathbf{x} = 0$ thanks to the property (B.7).

⋆ $\underline{|\mathbf{u}| = k - 1 \Rightarrow |\mathbf{u}| = k}$ Let $k \in [\![1, p]\!]$. It is assumed the elements of $\left(\tilde{f}_{\mathbf{u}}\right)_{\mathbf{u}\in I}$ and $(f_{\mathbf{u}})_{\mathbf{u}\in I}$ are equal for $|\mathbf{u}| < k$.

Let $\mathbf{v} \in I$, $|\mathbf{v}| = k$. One wants to prove that $\tilde{f}_{\mathbf{v}} = f_{\mathbf{v}}$. To achieve this, one makes the integration of (B.6) with respect to $x_{\bar{\mathbf{v}}}$:

$$\int_{[0,1]^{|\bar{\mathbf{v}}|}} f(x)d\mathbf{x} = f_\emptyset + \sum_{\mathbf{u}\in I} \int_{[0,1]^{|\bar{\mathbf{v}}|}} f_{\mathbf{u}}(x_{\mathbf{u}})d\mathbf{x}_{\bar{\mathbf{v}}} = \tilde{f}_\emptyset + \sum_{\mathbf{u}\in I} \int_{[0,1]^{|\bar{\mathbf{v}}|}} \tilde{f}_{\mathbf{u}}(x_{\mathbf{u}})d\mathbf{x}_{\bar{\mathbf{v}}}$$

In the sum, the terms $\mathbf{u} \in I$ of cardinal lower than k are equal on each side (hypothesis of recurrence). One takes them out of the sum. For f and $\tilde{f}$, the remaining integral is written as follows

$$\int_{[0,1]^{|\bar{\mathbf{v}}|}} f_{\mathbf{u}}(x_{\mathbf{u}})d\mathbf{x}_{\bar{\mathbf{v}}} = \begin{cases} f_{\mathbf{v}}(\mathbf{x}_{\mathbf{v}}) & \text{if } \mathbf{u} = \mathbf{v} \\ 0 & \text{if } \mathbf{u} \neq \mathbf{v} \end{cases}$$

The case $\mathbf{u} \neq \mathbf{v}$ is split in two sub-cases : $\exists i \in \mathbf{u}, i \notin \mathbf{v}$ or $\exists i \in \mathbf{u}, i \notin \mathbf{v}$.

If $\exists i \in \mathbf{u}, i \notin \mathbf{v}$, then $i \in \bar{\mathbf{v}}$ and the property (B.7) ensures nullity.

If $\exists i \in \mathbf{v}, i \notin \mathbf{u}$, then $|\mathbf{u}| \leqslant k$. Terms of cardinal lower than k have been removed thanks to the recurrence hypothesis. Only terms of equal cardinal are left. The two sub-cases lead thus to the same conclusion. We are back to the previous sub-case and the property (B.7) ensures nullity.

It has been shown that $\forall k \in [\![0,p]\!]$, $\forall \mathbf{u} \in I$, $|\mathbf{u}| = k$, $f_{\mathbf{u}} = \tilde{f}_{\mathbf{u}}$. One can conclude that

$$\forall \mathbf{u} \in I,\ f_{\mathbf{u}} = \tilde{f}_{\mathbf{u}}$$

Which ensures the uniqueness of the family $(f_{\mathbf{u}})_{\mathbf{u}\in I}$ and ends the proof. □

B.2.2 Orthogonality in ANOVA decomposition

Proposition B.1. *Let $f : [0,1]^p \to \mathbb{R}$ be integrable and $(f_{\mathbf{u}})_{\mathbf{u}\in I}$ the functions of its ANOVA decomposition. Then,*

$$\forall \mathbf{u}, \mathbf{v} \in I,\ \mathbf{u} \neq \mathbf{v},\ \int_{[0,1]^p} f_{\mathbf{u}}(x_{\mathbf{u}}) f_{\mathbf{v}}(x_{\mathbf{v}}) dx = 0 \tag{B.8}$$

Proof. (From Sobol (1993, 1976))
Soient $\mathbf{u}$ et $\mathbf{v}$ dans I tels que $\mathbf{u} \neq \mathbf{v}$. Précision le sens de $\mathbf{u} \neq \mathbf{v}$, négation de $\mathbf{u} = \mathbf{v}$.

$$\mathbf{u} = \mathbf{v} \quad \Leftrightarrow \quad \mathbf{u} \subset \mathbf{v} \ \wedge\ \mathbf{v} \subset \mathbf{u} \quad \Leftrightarrow \quad \forall i \in \mathbf{u}, i \in \mathbf{v} \ \wedge\ \forall j \in \mathbf{v}, j \in \mathbf{u}$$

Par négation, on a

$$\mathbf{u} \neq \mathbf{v} \quad \Leftrightarrow \quad \exists i \in \mathbf{u}, i \notin \mathbf{v} \ \vee\ \exists j \in \mathbf{v}, j \notin \mathbf{u}$$

Dans notre problème, $\mathbf{u}$ et $\mathbf{v}$ ont des rôles symétriques. On peut donc considérer $\exists i \in \mathbf{u}, i \notin \mathbf{v}$ ou bien $\exists j \in \mathbf{v}, j \notin \mathbf{u}$ indifféremment. Disons que l'on considère $\exists i \in \mathbf{u}, i \notin \mathbf{v}$. Alors, d'après le théorème de Fubini,

$$\int_{[0,1]^p} f_{\mathbf{u}}(x_{\mathbf{u}}) f_{\mathbf{v}}(x_{\mathbf{v}}) dx = \int_{[0,1]^{p-1}} \left(\int_0^1 f_{\mathbf{u}}(x_{\mathbf{u}}) f_{\mathbf{v}}(x_{\mathbf{v}}) dx_i \right) dx_{\bar{i}} = \int_{[0,1]^{p-1}} \left(f_{\mathbf{v}}(x_{\mathbf{v}}) \underbrace{\int_0^1 f_{\mathbf{u}}(x_{\mathbf{u}}) dx_i}_{=0} \right) dx_{\bar{i}} = 0$$

Le fait que $i \in \mathbf{u}, i \notin \mathbf{v}$ permet de sortir $f_{\mathbf{v}}(x_{\mathbf{v}})$ de l'intégrale et d'annuler $\int_0^1 f_{\mathbf{u}}(x_{\mathbf{u}}) dx_i$ grâce à la propriété (B.7). □

B.2.3 Orthogonality of Hoeffding spaces

Proposition B.2 (Orthogonality of Hoeffding spaces)**.**

$$\forall\, \mathbf{u}, \mathbf{v} \in I, \mathbf{u} \neq \mathbf{v}, \quad H^0_{\mathbf{u}} \perp H^0_{\mathbf{v}} \tag{B.9}$$

Proof. (From Chastaing (2013))
Comme pour la décomposition ANOVA, ce résultat se prouve en deux parties : d'abord un résultat du type (B.7), ensuite le résultat d'orthogonalité proprement dit (B.9).
Montrons d'abord

$$\forall \mathbf{u} \in I,\ \forall h_{\mathbf{u}} \in H^0_{\mathbf{u}},\ \forall i \in \mathbf{u},\ \int h_{\mathbf{u}}(x_{\mathbf{u}}) \eta_{X_i}(dx_i) = 0 \tag{B.10}$$

Soit $\mathbf{u} \in I$, $h_{\mathbf{u}} \in H^0_{\mathbf{u}}$ et $i \in \mathbf{u}$. Remarquons que $\{i, \bar{\mathbf{u}}\} = \overline{\{u \setminus i\}}$ que $\int \eta_{X_{\bar{\mathbf{u}}}}(dx_{\bar{\mathbf{u}}}) = 1$.
L'indépendance des composantes permet d'écrire

$$\begin{aligned}
\int h_{\mathbf{u}}(x_{\mathbf{u}})\eta_{X_i}(dx_i) &= \int \left(\int h_{\mathbf{u}}(x_{\mathbf{u}})\eta_{X_i}(dx_i) \right) \eta_{X_{\bar{\mathbf{u}}}}(dx_{\bar{\mathbf{u}}}) && \text{indépendance des composantes} \\
&= \int h_{\mathbf{u}}(x_{\mathbf{u}})\eta_{X_{\bar{\mathbf{v}}}}(dx_{\bar{\mathbf{v}}}) && \text{en prenant } \mathbf{v} = \mathbf{u} \setminus i \\
&= \mathbb{E}\left[h_{\mathbf{u}}(x_{\mathbf{u}})|X_{\mathbf{v}}\right] = 0 && \text{car } |\mathbf{v}| < |\mathbf{u}|
\end{aligned}$$

ce qui prouve le résultat intermédiaire.

Soit maintenant $\mathbf{u} \neq \mathbf{v} \in I$, $h_{\mathbf{u}} \in H^0_{\mathbf{u}}$ et $h_{\mathbf{v}} \in H^0_{\mathbf{v}}$. Comme $\mathbf{u} \neq \mathbf{v}$ on prend $i \in \mathbf{u}, i \notin \mathbf{v}$.

$$\int h_{\mathbf{u}}(x_{\mathbf{u}})h_{\mathbf{v}}(x_{\mathbf{v}})\eta_X(x) = \int h_{\mathbf{u}}(x_{\mathbf{v}}) \underbrace{\left(\int h_{\mathbf{u}}(x_{\mathbf{u}})\eta_{X_i}(dx_i) \right)}_{=0} \eta_{X_{\bar{i}}}(dx_{\bar{i}}) = 0$$

Le cas $i \in \mathbf{v}, i \notin \mathbf{u}$ est immédiat puisque $\mathbf{u}$ et $\mathbf{v}$ ont un rôle symétrique. □

B.2.4 Hoeffding's lemma

Lemme B.1 (Hoeffding projection)**.** *Considering*

- $X_1, ..., X_p$ *independent random variables.*
- $T \in L^2(\Omega, \mathbb{R})$ *(real random variable of finite variance* $\mathbb{E}\left[T^2\right] < +\infty$*).*

Then, for all $\mathbf{u} \in I$*, the orthogonal projection of* T *in* $H^0_{\mathbf{u}}$ *in written*

$$\pi_{H^0_{\mathbf{u}}}(T) = \sum_{\mathbf{v} \subset \mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} \mathbb{E}\left[T|X_{\mathbf{v}}\right] \tag{B.11}$$

Proof. (From Chastaing (2013))
Soit $\mathbf{u} \in I$. Montrons d'abord que l'opérateur définit par l'équation (B.11) est le projecteur orthogonal de L^2 sur $H^0_{\mathbf{u}}$.
On désigne par $\pi_{H^0_{\mathbf{u}}}$ le projecteur orthogonal de L^2 sur $H^0_{\mathbf{u}}$. Son existence est garantie par le fait que $H^0_{\mathbf{u}}$ est une sous-espace vectoriel de L^2 ($H^0_{\mathbf{u}}$ est stable par combinaison linéaire puisque l'espérance est linéaire).
On désigne par $\phi_{\mathbf{u}}$ l'opérateur définit par l'équation (B.11) : $\phi_{\mathbf{u}}(T) = \sum_{\mathbf{v} \subset \mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} \mathbb{E}\left[T|X_{\mathbf{v}}\right]$
On veut prouver que $\forall T \in L^2$, $\phi_{\mathbf{u}}(T) = \pi_{H^0_{\mathbf{u}}}(T)$.

Soit $T \in L^2$. Par définition de $\pi_{H^0_{\mathbf{u}}}$, $L^2 = \ker(\pi_{H^0_{\mathbf{u}}}) \overset{\perp}{\oplus} \mathrm{Im}(\pi_{H^0_{\mathbf{u}}}) = (H^0_{\mathbf{u}})^{\perp} \overset{\perp}{\oplus} H^0_{\mathbf{u}}$. Donc

$$\exists! \, (F', G') \in (H^0_{\mathbf{u}})^{\perp} \times H^0_{\mathbf{u}}, \; T = F' + G'$$

de plus $G' = \pi_{H^0_{\mathbf{u}}}(T)$ et $F' = T - \pi_{H^0_{\mathbf{u}}}(T)$.
Or,

$$T = \underbrace{T - \phi_{\mathbf{u}}(T)}_{F} + \underbrace{\phi_{\mathbf{u}}(T)}_{G}$$

donc il suffit de montrer que $G \in H^0_{\mathbf{u}}$ et $F \in (H^0_{\mathbf{u}})^\perp$ pour conclure par unicité de la décomposition orthogonale que $G = G'$ et $F = F'$. On aura alors prouvé que $\forall T \in L^2$, $\phi_{\mathbf{u}}(T) = G = G' = \pi_{H^0_{\mathbf{u}}}(T)$.

$\boxed{\phi_{\mathbf{u}}(T) \in H^0_{\mathbf{u}}}$

Soit $\mathbf{w} \in I$, $\mathbf{w} \subset \mathbf{u}$. Montrons que $\mathbb{E}\left[G|X_{\mathbf{w}}\right] = 0$.

$$\begin{aligned}
\mathbb{E}\left[\phi_{\mathbf{u}}(T)|X_{\mathbf{w}}\right] &= \sum_{\mathbf{v}\subset\mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|}\mathbb{E}\left[\mathbb{E}\left[T|X_{\mathbf{v}}\right]|X_{\mathbf{w}}\right] \\
&= \sum_{\mathbf{v}\subset\mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|}\mathbb{E}\left[\mathbb{E}\left[T|X_{\mathbf{v}}\right]|X_{\mathbf{v}\cap\mathbf{w}}\right] && \text{par indépendance des } X_i \\
&= \sum_{\mathbf{v}\subset\mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|}\mathbb{E}\left[T|X_{\mathbf{v}\cap\mathbf{w}}\right] && \text{car } \sigma(X_{\mathbf{v}\cap\mathbf{w}}) \subset \sigma(X_{\mathbf{v}})
\end{aligned}$$

Arrêtons-nous sur ce résultat en examinant quelques cas particuliers.
Pour $\mathbf{u} = \{i\}$,

$$\phi_i(T) = \mathbb{E}\left[T|X_i\right] - \mathbb{E}\left[T\right]$$

Pour $\mathbf{u} = \{i, j\}$,

$$\phi_{i,j}(T) = \mathbb{E}\left[T|X_i, X_j\right] - \mathbb{E}\left[T|X_i\right] - \mathbb{E}\left[T|X_j\right] + \mathbb{E}\left[T\right]$$

Pour $\mathbf{u} = \{i, j, k\}$,

$$\begin{aligned}
\phi_{i,j,k}(T) = \quad & \mathbb{E}\left[T|X_i, X_j, X_k\right] \\
& -\mathbb{E}\left[T|X_i, X_j\right] - \mathbb{E}\left[T|X_i, X_k\right] - \mathbb{E}\left[T|X_j, X_k\right] \\
& +\mathbb{E}\left[T|X_i\right] + \mathbb{E}\left[T|X_j\right] + \mathbb{E}\left[T|X_k\right] \\
& -\mathbb{E}\left[T\right]
\end{aligned}$$

Prenons par exemple $\mathbf{w} = \{i\}$ pour conditionner $\phi_{i,j,k}(T)$. On rappelle la convention $\mathbb{E}\left[T|\emptyset\right] = \mathbb{E}\left[T\right]$.

$$\begin{aligned}
\mathbb{E}\left[\phi_{i,j,k}(T)|X_i\right] &= \mathbb{E}\left[T|X_i\right] \\
&\quad -\mathbb{E}\left[T|X_i\right] - \mathbb{E}\left[T|X_i\right] - \mathbb{E}\left[T\right] \\
&\quad +\mathbb{E}\left[T|X_i\right] + \mathbb{E}\left[T\right] + \mathbb{E}\left[T\right] \\
&\quad -\mathbb{E}\left[T\right] \\
&= 2\mathbb{E}\left[T|X_i\right] - 2\mathbb{E}\left[T|X_i\right] + 2\mathbb{E}\left[T\right] - 2\mathbb{E}\left[T\right] \\
&= 0
\end{aligned}$$

De même si l'on prend $\mathbf{w} = \{i, j\}$ pour conditionner $\phi_{i,j,k}(T)$.

$$\begin{aligned}
\mathbb{E}\left[\phi_{i,j,k}(T)|X_i, X_j\right] = \quad & \mathbb{E}\left[T|X_i, X_j\right] \\
& -\mathbb{E}\left[T|X_i, X_j\right] - \mathbb{E}\left[T|X_i\right] - \mathbb{E}\left[T|X_j\right] \\
& +\mathbb{E}\left[T|X_i\right] + \mathbb{E}\left[T|X_j\right] + \mathbb{E}\left[T\right] \\
& -\mathbb{E}\left[T\right] \\
\mathbb{E}\left[\phi_{i,j,k}(T)|X_i, X_j\right] = \quad & \mathbb{E}\left[T|X_i, X_j\right] - \mathbb{E}\left[T|X_i, X_j\right] + \mathbb{E}\left[T|X_i\right] - \mathbb{E}\left[T|X_i, X_j\right] \\
& +\mathbb{E}\left[T|X_j\right] - \mathbb{E}\left[T|X_j\right] + \mathbb{E}\left[T\right] - \mathbb{E}\left[T\right] \\
\mathbb{E}\left[\phi_{i,j,k}(T)|X_i, X_j\right] = \quad & 0
\end{aligned}$$

On voit sur ces cas particuliers que la somme sur les $\mathbf{v} \subset \mathbf{u}$ se transforme en somme sur

$\mathbf{t} \subset \mathbf{w}$. On compte $\binom{|\mathbf{u}|-|\mathbf{w}|}{|\mathbf{t}|}$ termes identiques. On généralise ce résultat :

$$\begin{aligned}
\mathbb{E}\left[\phi_{\mathbf{u}}(T)|X_{\mathbf{w}}\right] &= \sum_{\mathbf{v}\subset\mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|}\mathbb{E}\left[T|X_{\mathbf{v}\cap\mathbf{w}}\right] \\
&= \sum_{\mathbf{t}\subset\mathbf{w}} \sum_{j=0}^{|\mathbf{u}|-|\mathbf{w}|} (-1)^{|\mathbf{u}|-|\mathbf{t}|-j}\binom{|\mathbf{u}|-|\mathbf{w}|}{j}\mathbb{E}\left[T|X_{\mathbf{t}}\right] \\
&= \sum_{\mathbf{t}\subset\mathbf{w}} (-1)^{|\mathbf{u}|-|\mathbf{t}|}\mathbb{E}\left[T|X_{\mathbf{t}}\right] \underbrace{\left(\sum_{j=0}^{|\mathbf{u}|-|\mathbf{w}|} (-1)^{j}\binom{|\mathbf{u}|-|\mathbf{w}|}{j}\right)}_{=0} \\
&= 0
\end{aligned}$$

On a montré que $\forall \mathbf{w} \in I$, $\mathbf{w} \subset \mathbf{u}$, $\mathbb{E}\left[\phi_{\mathbf{u}}(T)|X_{\mathbf{w}}\right] = 0$. Donc $\phi_{\mathbf{u}}(T) \in H_u^0$.

$\boxed{T - \phi_{\mathbf{u}}(T) \in (H_{\mathbf{u}}^0)^{\perp}}$

Soit $h_{\mathbf{u}} \in H_{\mathbf{u}}^0$. Montrons que $\mathbb{E}\left[(T - \phi_{\mathbf{u}}(T))h_{\mathbf{u}}(X_{\mathbf{u}})\right] = 0$.

$$\begin{aligned}
\mathbb{E}\left[(T - \phi_{\mathbf{u}}(T))h_{\mathbf{u}}(X_{\mathbf{u}})\right] &= \mathbb{E}\left[Th_{\mathbf{u}}(X_{\mathbf{u}})\right] - \mathbb{E}\left[\phi_{\mathbf{u}}(T))h_{\mathbf{u}}(X_{\mathbf{u}})\right] \\
&= \mathbb{E}\left[Th_{\mathbf{u}}(X_{\mathbf{u}})\right] - \underbrace{\mathbb{E}\left[\mathbb{E}\left[T|X_{\mathbf{u}}\right]h_{\mathbf{u}}(X_{\mathbf{u}})\right]}_{\alpha} - \sum_{\substack{\mathbf{v}\subset\mathbf{u}\\ \mathbf{v}\neq\mathbf{u}}} (-1)^{|\mathbf{u}|-|\mathbf{v}|}\underbrace{\mathbb{E}\left[\mathbb{E}\left[T|X_{\mathbf{v}}\right]h_{\mathbf{u}}(X_{\mathbf{u}})\right]}_{\beta}
\end{aligned}$$

Précisons les termes α et β :

$$\alpha = \mathbb{E}\left[\mathbb{E}\left[T|X_{\mathbf{u}}\right]h_{\mathbf{u}}(X_{\mathbf{u}})\right] = \mathbb{E}\left[\mathbb{E}\left[Th_{\mathbf{u}}(X_{\mathbf{u}})|X_{\mathbf{u}}\right]\right] = \mathbb{E}\left[Th_{\mathbf{u}}(X_{\mathbf{u}})\right]$$

car $X_{\mathbf{u}}$ est $X_{\mathbf{u}}$-mesurable, $h_{\mathbf{u}}$ est mesurable, donc $h_{\mathbf{u}}(X_{\mathbf{u}})$ est $X_{\mathbf{u}}$-mesurable.

$$\beta = \mathbb{E}\left[\mathbb{E}\left[T|X_{\mathbf{v}}\right]h_{\mathbf{u}}(X_{\mathbf{u}})\right] = \mathbb{E}\left[\mathbb{E}\left(\mathbb{E}\left[T|X_{\mathbf{v}}\right]h_{\mathbf{u}}(X_{\mathbf{u}})\middle|X_{\mathbf{v}}\right)\right] = \mathbb{E}\left[\mathbb{E}\left[T|X_{\mathbf{v}}\right]\underbrace{\mathbb{E}\left[h_{\mathbf{u}}(X_{\mathbf{u}})|X_{\mathbf{v}}\right]}_{=0}\right] = 0$$

Donc

$$\mathbb{E}\left[(T - \phi_{\mathbf{u}}(T))h_{\mathbf{u}}(X_{\mathbf{u}})\right] = \mathbb{E}\left[Th_{\mathbf{u}}(X_{\mathbf{u}})\right] - \mathbb{E}\left[Th_{\mathbf{u}}(X_{\mathbf{u}})\right] - 0 = 0$$

On vient de montrer que $(T - \phi_{\mathbf{u}}(T)) \perp h_{\mathbf{u}}(X_{\mathbf{u}})$, $\forall h_{\mathbf{u}}(X_{\mathbf{u}}) \in H_{\mathbf{u}}^0$. Donc $(T - \phi_{\mathbf{u}}(T)) \in (H_{\mathbf{u}}^0)^{\perp}$. Par unicité de la décomposition (propriété du projecteur orthogonal $\pi_{H_{\mathbf{u}}^0}$), on a bien

$$\forall T \in L^2,\ \pi_{H_{\mathbf{u}}^0}(T) = \sum_{\mathbf{v}\subset\mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|}\mathbb{E}\left[T|X_{\mathbf{v}}\right]$$

□

B.2.5 Hoeffding decomposition

Theorem B.4 (Hoeffding decomposition)**.** *Let $Y : (\Omega, \mathcal{F}, \mathbb{P}) \to (\mathbb{R}, \mathcal{B}(\mathbb{R}))$ et $X : (\Omega, \mathcal{F}, \mathbb{P}) \to (\mathbb{R}^p, \mathcal{B}(\mathbb{R}^p))$ such that $Y = f(X)$ with $f : (\mathbb{R}^p, \mathcal{B}(\mathbb{R}^p)) \to (\mathbb{R}, \mathcal{B}(\mathbb{R}))$, a measurable function. Previous notations hold.*
Under the following assumptions :

1. *Y has a finite variance (i.e. $\mathbb{E}\left[Y^2\right] < +\infty$).*
2. *The inputs X_i, $i \in [\![1, p]\!]$ are independent : $p_X(x) = \prod_{i=1}^p p_{X_i}(x_i)$.*

Then it exists an unique decomposition of Y with respect to $(X_i)_i$:

$$\begin{aligned} Y &= \sum_{\mathbf{u} \in I} f_{\mathbf{u}}(X_{\mathbf{u}}) \\ &= f_\emptyset + \sum_{i=1}^p f_i(X_i) + \sum_{1 \leqslant i < j \leqslant p} f_{ij}(X_i, X_j) + \cdots + f_{1,\ldots,p}(X_1, \ldots, X_p) \end{aligned} \tag{B.12}$$

such that

$$\forall \mathbf{u} \in I,\ f_{\mathbf{u}}(X_{\mathbf{u}}) = \sum_{\mathbf{v} \subset \mathbf{u}} (-1)^{|\mathbf{u}| - |\mathbf{v}|} \mathbb{E}\left[Y | X_{\mathbf{v}}\right] \tag{B.13}$$

Proof. (From Chastaing (2013))
Plaçons-nous dans le cadre de la projection de Hoeffding. Soit Y est de variance finie (i.e. $\mathbb{E}\left[Y^2\right] < +\infty$) telle que $Y = f(X)$. Par définition de L^2, $Y \in L^2(\Omega, \mathbb{R})$, et par définition des espaces de Hoeffding, $f \in H_{[\![1,p]\!]}$.

Montrons d'abord

$$\forall \mathbf{u} \in I,\ H_{\mathbf{u}} \subset \bigoplus_{\mathbf{v} \subset \mathbf{u}} H^0_{\mathbf{v}} \tag{B.14}$$

Soit $\mathbf{u} \in I$ et $h_{\mathbf{u}}(X_{\mathbf{u}}) \in H_{\mathbf{u}}$. D'après la proposition (B.2) (orthogonalité des espaces de Hoeffding), le projecteur orthogonal sur l'ensemble $\bigoplus_{\mathbf{v} \subset \mathbf{u}} H^0_{\mathbf{v}}$ s'écrit comme la somme des projecteurs orthogonaux sur chacun des ensembles $H^0_{\mathbf{v}}$.

$$\pi_{\bigoplus_{\mathbf{v} \subset \mathbf{u}} H^0_{\mathbf{v}}} = \sum_{\mathbf{v} \subset \mathbf{u}} \pi_{H^0_{\mathbf{v}}}$$

Ainsi,

$$\begin{aligned} h_{\mathbf{u}}(X_{\mathbf{u}}) - \pi_{\bigoplus_{\mathbf{v} \subset \mathbf{u}} H^0_{\mathbf{v}}}(h_{\mathbf{u}}(X_{\mathbf{u}})) &= h_{\mathbf{u}}(X_{\mathbf{u}}) - \sum_{\mathbf{v} \subset \mathbf{u}} \pi_{H^0_{\mathbf{v}}}(h_{\mathbf{u}}(X_{\mathbf{u}})) \\ &= h_{\mathbf{u}}(X_{\mathbf{u}}) - h_{\mathbf{u}}(X_{\mathbf{u}}) \qquad \text{car } \forall \mathbf{v} \subset \mathbf{u},\ \mathbf{v} \neq \mathbf{u},\ \pi_{H^0_{\mathbf{v}}}(h_{\mathbf{u}}(X_{\mathbf{u}})) = 0 \\ &= 0 \end{aligned}$$

On vient de montrer que $h_{\mathbf{u}}(X_{\mathbf{u}}) = \pi_{\bigoplus_{\mathbf{v} \subset \mathbf{u}} H^0_{\mathbf{v}}}(h_{\mathbf{u}}(X_{\mathbf{u}})) \in \bigoplus_{\mathbf{v} \subset \mathbf{u}} H^0_{\mathbf{v}}$ pour tout élément $h_{\mathbf{u}}(X_{\mathbf{u}})$ de $H_{\mathbf{u}}$. Donc $H_{\mathbf{u}} \subset \bigoplus_{\mathbf{v} \subset \mathbf{u}} H^0_{\mathbf{v}}$.

On applique ce résultat à $f \in H_{[\![1,p]\!]} \subset \bigoplus_{\mathbf{u}\in I} H^0_{\mathbf{u}}$.

$$f(X) = \sum_{\mathbf{u}\in I} \underbrace{\pi_{H^0_{\mathbf{u}}}(f(X))}_{f_{\mathbf{u}}(X_{\mathbf{u}})}$$

Et l'expression des $f_{\mathbf{u}}(X_{\mathbf{u}})$ découle du lemme de Hoeffding (B.1).

□

B.2.6 Variance decomposition

Corollary B.1. *Under the same assumptions than for the theorem* **??**,

$$V(Y) = \sum_{\mathbf{u}\in I} V(f_{\mathbf{u}}(X_{\mathbf{u}})) = \sum_{\mathbf{u}\in I} \left(V(\mathbb{E}\left[Y|X_{\mathbf{u}}\right]) + \sum_{\mathbf{v}\subset\mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} V(\mathbb{E}\left[Y|X_{\mathbf{v}}\right]) \right) \tag{B.15}$$

Proof. (From Chastaing (2013))
Comme Y est de variance finie (hypothèse du théorème **??**), on peut prendre la variance dans l'équation 4.6.

$$\begin{aligned}
V(Y) &= V\left(\sum_{\mathbf{u}\in I} f_{\mathbf{u}}(X_{\mathbf{u}})\right) \\
&= \sum_{\mathbf{u}\in I} V(f_{\mathbf{u}}(X_{\mathbf{u}})) + \sum_{\substack{\mathbf{u},\mathbf{v}\in I \\ \mathbf{u}\cap\mathbf{v}\neq\mathbf{u},\mathbf{v}}} \underbrace{\operatorname{cov}(f_{\mathbf{u}}(X_{\mathbf{u}}), f_{\mathbf{v}}(X_{\mathbf{v}}))}_{=0 \text{ car } H^0_{\mathbf{u}} \perp H^0_{\mathbf{v}}} \\
&= \sum_{\mathbf{u}\in I} V\left(\sum_{\mathbf{v}\subset\mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} \mathbb{E}\left[Y|X_{\mathbf{v}}\right]\right) \\
&= \sum_{\mathbf{u}\in I} \left(V(\mathbb{E}\left[Y|X_{\mathbf{u}}\right]) + \sum_{\substack{\mathbf{v}\subset\mathbf{u} \\ \mathbf{v}\neq\mathbf{u}}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} V(\mathbb{E}\left[Y|X_{\mathbf{v}}\right]) + \sum_{\substack{\mathbf{v},\mathbf{w}\subset\mathbf{u} \\ \mathbf{v}\cap\mathbf{w}\neq \\ \mathbf{w},\mathbf{v}}} \underbrace{\operatorname{cov}(\mathbb{E}\left[Y|X_{\mathbf{w}}\right], \mathbb{E}\left[Y|X_{\mathbf{v}}\right])}_{=0 \text{ par indépendance}} \right) \\
&= \sum_{\mathbf{u}\in I} \left(V(\mathbb{E}\left[Y|X_{\mathbf{u}}\right]) + \sum_{\substack{\mathbf{v}\subset\mathbf{u} \\ \mathbf{v}\neq\mathbf{u}}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} V(\mathbb{E}\left[Y|X_{\mathbf{v}}\right]) \right)
\end{aligned}$$

□

B.2.7 Simple, complete and total Sobol indices

Proposition B.3. *Let $\mathbf{u} \in I$. With the notation introduced above, simple, total and complete Sobol indices have the following expression :*

$$D_{\mathbf{u}} = V(\mathbb{E}[Y|X_{\mathbf{u}}]) + \sum_{\substack{\mathbf{v}\subset\mathbf{u}\\ \mathbf{v}\neq\mathbf{u}}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} V(\mathbb{E}[Y|X_{\mathbf{v}}]) \tag{B.16}$$

$$D^C_{\mathbf{u}} = V(\mathbb{E}[Y|X_{\mathbf{u}}]) \tag{B.17}$$

$$D^T_{\mathbf{u}} = V(Y) - V(\mathbb{E}[Y|X_{\bar{\mathbf{u}}}]) \tag{B.18}$$

Proof. (Evoked in Owen (2013) and Homma and Saltelli (1996))

Expression 4.9 is the direct application of theorem 4.2.

- Expression 4.10 is obtained by writing the double sum and grouping the terms of same sign:

$$\begin{aligned}
D^C_{\mathbf{u}} &= \sum_{\substack{\mathbf{v}\in I\\ \mathbf{v}\subseteq\mathbf{u}}} D_{\mathbf{v}} = \sum_{\substack{\mathbf{v}\in I\\ \mathbf{v}\subseteq\mathbf{u}}} \sum_{\substack{\mathbf{w}\in I\\ \mathbf{w}\subseteq\mathbf{v}}} (-1)^{|\mathbf{v}|-|\mathbf{w}|} V(\mathbb{E}[Y|X_{\mathbf{w}}]) \\
&= V(\mathbb{E}[Y|X_{\mathbf{u}}]) + \sum_{\mathbf{t}\subset\mathbf{u}} \sum_{j=0}^{|\mathbf{u}|-|\mathbf{t}|} (-1)^{|\mathbf{u}|-|\mathbf{t}|-j} \binom{|\mathbf{u}|-|\mathbf{t}|}{j} V(\mathbb{E}[Y|X_{\mathbf{t}}]) \\
&= V(\mathbb{E}[Y|X_{\mathbf{u}}]) + \sum_{\mathbf{t}\subset\mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{t}|} V(\mathbb{E}[Y|X_{\mathbf{t}}]) \underbrace{\sum_{j=0}^{|\mathbf{u}|-|\mathbf{t}|} (-1)^j \binom{|\mathbf{u}|-|\mathbf{t}|}{j}}_{=0} \\
&= V(\mathbb{E}[Y|X_{\mathbf{u}}])
\end{aligned}$$

- Expression 4.11 is a consequence of 4.10:

$$D^T_{\mathbf{u}} + D^C_{\bar{\mathbf{u}}} = \sum_{\substack{\mathbf{v}\in I\\ \mathbf{v}\cap\mathbf{u}\neq\emptyset}} D_{\mathbf{v}} + \sum_{\substack{\mathbf{v}\in I\\ \mathbf{v}\subseteq\bar{\mathbf{u}}}} D_{\mathbf{v}} = \sum_{\mathbf{v}\in I} D_{\mathbf{v}} = V(Y)$$

because $\{\mathbf{v}\in I, \mathbf{v}\cap\mathbf{u}\neq\emptyset\}\cup\{\mathbf{v}\in I, \mathbf{v}\subseteq\bar{\mathbf{u}}\} = I$ (see **?** and **?** equation 11). It follows that

$$D^T_{\mathbf{u}} = V(Y) - D^C_{\bar{\mathbf{u}}} = V(Y) - V(\mathbb{E}[Y|X_{\bar{\mathbf{u}}}])$$

□

B.3 Results related to penalized regression

B.3.1 Sobol indices estimation with regression

Lemme B.2. *For any set* $\mathbf{u}$ *of indices* ($\mathbf{u} \in I'$)*, when*

$$(a_{\mathbf{u}}, b_{\mathbf{u}}) = \underset{(a,b)}{arg\min} \left\{ \mathbb{E}\left[(Y - aY_{\mathbf{u}} - b)^2 \right] + \mathbb{E}\left[(Y_{\mathbf{u}} - aY - b)^2 \right] \right\} \tag{B.19}$$

then

$$a_{\mathbf{u}} = S_{\mathbf{u}}$$

Proof. In the function to minimize, Y and $Y_{\mathbf{u}}$ have a symmetric role and follows the same law.

$$\begin{aligned} J(a,b) &= \mathbb{E}\left[(Y - aY_{\mathbf{u}} - b)^2 \right] + \mathbb{E}\left[(Y_{\mathbf{u}} - aY - b)^2 \right] \\ &= \mathbb{E}\left[Y^2\right] + a^2\mathbb{E}\left[Y_{\mathbf{u}}^2\right] + b^2 - 2a\mathbb{E}\left[Y_{\mathbf{u}}Y\right] - 2ab\mathbb{E}\left[Y_{\mathbf{u}}\right] - 2b\mathbb{E}\left[Y\right] \\ &\quad + \mathbb{E}\left[Y_{\mathbf{u}}^2\right] + a^2\mathbb{E}\left[Y^2\right] + b^2 - 2a\mathbb{E}\left[Y_{\mathbf{u}}Y\right] - 2ab\mathbb{E}\left[Y\right] - 2b\mathbb{E}\left[Y_{\mathbf{u}}\right] \\ &= 2\left(\mathbb{E}\left[Y^2\right] + a^2\mathbb{E}\left[Y^2\right] + b^2 - 2a\mathbb{E}\left[Y_{\mathbf{u}}Y\right] - 2ab\mathbb{E}\left[Y\right] - 2b\mathbb{E}\left[Y\right]\right) \end{aligned}$$

which is parabolic, thus has only one minimum, reached at the point denoted $(a_{\mathbf{u}}, b_{\mathbf{u}})$. We solve $\nabla J(a_{\mathbf{u}}, b_{\mathbf{u}}) = 0$.

$$\nabla J(a,b) = \begin{pmatrix} 4a\mathbb{E}\left[Y^2\right] + 4b\mathbb{E}\left[Y\right] - 4\mathbb{E}\left[Y_{\mathbf{u}}Y\right] \\ 4b + 4a\mathbb{E}\left[Y\right] - 4\mathbb{E}\left[Y\right] \end{pmatrix}$$

Setting the second component to zero of the gradient yields to equation :

$$b_{\mathbf{u}} = \mathbb{E}\left[Y\right](1 - a_{\mathbf{u}}) \tag{B.20}$$

Setting the first component to zero and using the last equation yield to

$$a_{\mathbf{u}} = \frac{\mathbb{E}\left[Y_{\mathbf{u}}Y\right] - \mathbb{E}\left[Y\right]^2}{\mathbb{E}\left[Y^2\right] - \mathbb{E}\left[Y\right]^2} = \frac{\mathrm{cov}(Y, Y_{\mathbf{u}})}{V(Y)} = S_{\mathbf{u}}$$

□

B.3.2 Solution of Lasso regression

Proposition B.4. *For any* $\mathbf{u} \in I'$*, the Lasso and least squares estimators are related according to the following formula:*

$$\widehat{S}_{\mathbf{u}}^{l1} = \max\left(\widehat{S}_{\mathbf{u}}^{ls} - \varepsilon_1, 0\right)$$

with $\varepsilon_1 = \frac{\lambda_1}{2\sigma^2}$.

Proof. The cost function to minimize is denoted

$$J: \begin{array}{ccc} \mathbb{R} & \to & \mathbb{R} \\ a & \mapsto & J(a) = \|Y - aY_{\mathbf{u}}\|_2^2 + \lambda_1 |a| \end{array}$$

For the sake of readability, notations including $\mathbf{u}$ are shortened as follow: $\sigma^2 = Y^T Y = Y_{\mathbf{u}}^T Y_{\mathbf{u}}$, $\widehat{a}^{ls} = \widehat{S}_{\mathbf{u}}^{ls} = (Y_{\mathbf{u}}^T Y_{\mathbf{u}})^{-1} Y_{\mathbf{u}}^T Y$. This cost function is developed.

$$J(a) = \sigma^2 \left(1 + a^2 - 2a\widehat{a}^{ls} + \frac{\lambda_1}{\sigma^2}|a|\right)$$

The function J is differentiable everywhere but in $a = 0$. For any $a \neq 0$,

$$\frac{\partial J}{\partial a}(a) = 2\sigma^2 \left(a - \widehat{a}^{ls} + \frac{\lambda_1}{2\sigma^2}\text{sign}(a)\right)$$

The term $\varepsilon_1 = \frac{\lambda_1}{2\sigma^2}$ appears, this notation holds for the rest of the proof. The derivative is discontinuous when $a = 0$. It jumps from $-\widehat{a}^{ls} - \varepsilon_1$ to $-\widehat{a}^{ls} + \varepsilon_1$. When we solve $\frac{\partial J}{\partial a}(\widehat{a}) = 0$ we must distinguish the case of $0 \in [-\widehat{a}^{ls} - \varepsilon_1, -\widehat{a}^{ls} + \varepsilon_1]$ (that is to say $|\widehat{a}^{ls}| \leqslant \varepsilon_1$). Three cases are to consider:

- $|\widehat{a}^{ls}| \leqslant \varepsilon_1$
- $\widehat{a}^{ls} > \varepsilon_1$
- $\widehat{a}^{ls} < -\varepsilon_1$

When $|\widehat{a}^{ls}| \leqslant \varepsilon_1$, there is no solution for the equation $\frac{\partial J}{\partial a}(\widehat{a}) = 0$. Nevertheless, when $|\widehat{a}^{ls}| \leqslant \varepsilon_1$, the value 0 is within the discontinuity at $a = 0$.

$$\lim_{a \to 0^+} \frac{\partial J}{\partial a}(a) = 2\sigma^2 \left(\varepsilon_1 - \widehat{a}^{ls}\right) \geqslant 0 \text{ and } \lim_{a \to 0^-} \frac{\partial J}{\partial a}(a) = -2\sigma^2 \left(\varepsilon_1 + \widehat{a}^{ls}\right) \leqslant 0$$

Hence, when $|\widehat{a}^{ls}| \leqslant \varepsilon_1$, $J(a)$ reaches its minimum at $\widehat{a} = 0$.

When $\widehat{a}^{ls} > \varepsilon_1$, setting the derivative to zero gives

$$\widehat{a} = \widehat{a}^{ls} - \varepsilon_1$$

When $\widehat{a}^{ls} < -\varepsilon$, setting the derivative to zero gives

$$\widehat{a} = \widehat{a}^{ls} + \varepsilon_1$$

Finally, all cases are gathered into the single following expression:

$$\widehat{a} = \text{sign}(\widehat{a}^{ls}) \max\left(|\widehat{a}^{ls}| - \varepsilon_1, 0\right)$$

In the case of Sobol indices, it is reasonable to dismiss the case $\widehat{a}^{ls} < -\varepsilon_1$ (it can happen either if the least square estimation is really bad or the penalty is very small). Going back to the problem 8.13, the Lasso estimator is given by $\widehat{S}_{\mathbf{u}}^{l1} = \widehat{a}$.

□

B.3.3 Solution of best subset regression

Proposition B.5. *For any $\mathbf{u} \in I'$, the best subset and least squares estimators are related according to the following formula:*

$$\widehat{S}^{l0}_{\mathbf{u}} = \widehat{S}^{ls}_{\mathbf{u}} \mathbf{1}_{\widehat{S}^{ls}_{\mathbf{u}} > \varepsilon_0}$$

with $\varepsilon_0 = \sqrt{\frac{\lambda_0}{\sigma^2}}$.

Proof. We assume exactly k upon d coefficients of the vector $\mathbf{a}$ are non-zero and we index $\mathbf{a}$ such that they are first.

$$\mathbf{a} = (a_{\mathbf{u}})_{\mathbf{u} \in I'} = (a_1, ..., a_k, 0, ..., 0)$$

The cost function to minimize depends on the number of non-zero:

$$J(k) = \sum_{i=1}^{k} \|Y - a_i Y_i\|_2^2 + (2^p - k)\|Y\|_2^2 + k\lambda_0$$

It is worth to set the next coefficient a_{k+1} to a non-zero value only if it shrinks the cost function:

$$\begin{aligned} a_{k+1} \neq 0 &\Leftrightarrow J(k+1) < J(k) \\ &\Leftrightarrow \|Y\|_2^2 - \|Y - a_{k+1} Y_{k+1}\|_2^2 - \lambda_0 > 0 \end{aligned}$$

and the next coefficient will take the value which maximizes the improvement. For any $\mathbf{u} \in I'$, the best subset estimation is the solution to

$$\max_a \left\{ \|Y\|_2^2 - \|Y - aY_{\mathbf{u}}\|_2^2 - \lambda_0 \right\}$$

This gain function to maximize is denoted $G(a)$.

$$\begin{aligned} G(a) &= \|Y\|_2^2 - \|Y - aY_{\mathbf{u}}\|_2^2 - \lambda_0 \\ &= \sigma^2 - (Y^T Y + a^2 Y_{\mathbf{u}}^T Y_{\mathbf{u}} - 2aY^T Y_{\mathbf{u}}) - \lambda_0 \\ &= 2\sigma^2 (a\widehat{a}^{ls} - \tfrac{a^2}{2} - \tfrac{\lambda_0}{2\sigma^2}) \end{aligned}$$

Solving $G'(\hat{a}) = 0$ gives $\hat{a} = \widehat{a}^{ls}$. Hence, when it is worth to add a non-zero coefficient, this coefficient is equal to the least square estimator. It is worth to add a non-zero coefficient if $G(\hat{a}) > 0$, that is to say when

$$(\widehat{a}^{ls})^2 - \frac{(\widehat{a}^{ls})^2}{2} - \frac{\lambda_0}{2\sigma^2} > 0 \quad \text{or when} \quad |\widehat{a}^{ls}| > \sqrt{\frac{\lambda_0}{\sigma^2}}$$

Combining these condition gives the result:

$$\hat{a} = \widehat{a}^{ls} \mathbf{1}_{|\widehat{a}^{ls}| > \varepsilon_0}$$

with $\varepsilon_0 = \sqrt{\lambda_0}/\sigma$.

□

B.3.4 Bias, variance and error of prediction and estimation

Formulae to link error of prediction $(Y - \hat{Y})$ and the error of estimation $(\beta - \hat{\beta})$ when the the errors are estimated by cross-validation with a testing sample independent from the training sample. Mentioned at section 8.23, page 235.

$$\mathsf{Bi}_Y = \mathbb{E}\left[X\right]\mathsf{Bi}_\beta \tag{B.21}$$

$$\mathsf{Var}_Y = \mathbb{E}\left[X\mathsf{Var}_\beta X^T\right] + V\Big(X\mathbb{E}\left[\widehat{\beta}\right]\Big) \tag{B.22}$$

$$\mathsf{MSE}_Y = \mathsf{Bi}_\beta^T\mathbb{E}\left[X^TX\right]\mathsf{Bi}_\beta + \mathbb{E}\left[\mathrm{tr}(X\mathsf{Var}_\beta X^T)\right] + \sigma_0^2 \tag{B.23}$$

Proof. of equations (B.21), (B.22) and (B.23).

Equation of the bias (B.21)**.**

$$\mathsf{Bi}_Y = \mathbb{E}\left[\widehat{Y} - Y\right] = \mathbb{E}\left[X(\widehat{\beta} - \beta) - \epsilon\right] = \mathbb{E}\left[X\right]\mathsf{Bi}_\beta$$

because $\widehat{\beta}$ is independent from X.

Equation of the variance (B.22)**.**

From the total variance formula, we have

$$\mathsf{Var}_Y = \mathbb{E}\left[V\Big(X\widehat{\beta}|X\Big)\right] + V\Big(\mathbb{E}\left[X\widehat{\beta}|X\right]\Big) = \mathbb{E}\left[X\underbrace{V\Big(\widehat{\beta}|X\Big)}_{=\mathsf{Var}_\beta}\right] + V\left(X\underbrace{\mathbb{E}\left[\widehat{\beta}|X\right]}_{=\mathbb{E}\left[\widehat{\beta}\right]}\right)$$

because $\widehat{\beta}$ is independent from X, the conditioning to X does not matter.

Equation of the MSE (B.23)**.**

From the law of total expectation, we have $\mathsf{MSE}_Y = \mathbb{E}\left[\mathbb{E}\left[(\widehat{Y} - Y)^T(\widehat{Y} - Y)|X\right]\right]$.

$$\begin{aligned}
\mathbb{E}\left[(\widehat{Y} - Y)^T(\widehat{Y} - Y)|X\right] &= \mathbb{E}\left[(\widehat{Y} - X\mathbb{E}\left[\widehat{\beta}\right] + X\mathbb{E}\left[\widehat{\beta}\right] - Y)^T(\widehat{Y} - X\mathbb{E}\left[\widehat{\beta}\right] + X\mathbb{E}\left[\widehat{\beta}\right] - Y)|X\right] \\
&= \underbrace{\mathbb{E}\left[(Y - X\mathbb{E}\left[\widehat{\beta}\right])^T(Y - X\mathbb{E}\left[\widehat{\beta}\right])|X\right]}_{\mathbf{I}} \\
&\quad \underbrace{+\mathbb{E}\left[(\widehat{Y} - X\mathbb{E}\left[\widehat{\beta}\right])^T(\widehat{Y} - X\mathbb{E}\left[\widehat{\beta}\right])|X\right]}_{\mathbf{II}} \\
&\quad \underbrace{+2\mathbb{E}\left[(Y - X\mathbb{E}\left[\widehat{\beta}\right])^T(\widehat{Y} - X\mathbb{E}\left[\widehat{\beta}\right])|X\right]}_{\mathbf{III}}
\end{aligned}$$

Because $\widehat{\beta}$ is the estimator built on the sub-sample (Y_A, A), we can assume that $\widehat{\beta}$ is independent of X. In particular, $\mathbb{E}\left[\widehat{\beta}\right] = \mathbb{E}\left[\widehat{\beta}|X\right]$, which is helpful to explicit the terms **I**, **II** and **III**.

The term **I** is linked to the bias on β:

$$\mathbf{I} = \mathbb{E}\left[(X\beta + \epsilon - X\mathbb{E}\left[\widehat{\beta}\right])^T(X\beta + \epsilon - X\mathbb{E}\left[\widehat{\beta}\right])|X\right] = \mathbb{E}\left[\mathsf{Bi}_\beta^T X^T X \mathsf{Bi}_\beta|X\right] + \mathbb{E}\left[\epsilon^2\right] = \mathsf{Bi}_\beta^T X^T X \mathsf{Bi}_\beta + \mathbb{E}\left[\epsilon^2\right]$$

with $\mathbb{E}\left[\epsilon^2\right] = \sigma_0^2$.
The term **II** is linked to the variance on β:

$$\mathbf{II} = \mathbb{E}\left[(X\widehat{\beta} - X\mathbb{E}\left[\widehat{\beta}\right])^T(X\widehat{\beta} - X\mathbb{E}\left[\widehat{\beta}\right])|X\right] = \mathrm{tr}(V\left(X\widehat{\beta}|X\right)) = \mathrm{tr}(X\mathsf{Var}_\beta X^T)$$

The term **III** is null:

$$\mathbf{III} = \mathbb{E}\left[(X\beta + \epsilon - X\mathbb{E}\left[\widehat{\beta}\right])^T(X\widehat{\beta} - X\mathbb{E}\left[\widehat{\beta}\right])|X\right] = (\beta - \mathbb{E}\left[\widehat{\beta}\right])^T X^T X \underbrace{\mathbb{E}\left[(\widehat{\beta} - \mathbb{E}\left[\widehat{\beta}\right])|X\right]}_{=0} = 0$$

Finally,

$$\mathbb{E}\left[(\widehat{Y} - Y)^T(\widehat{Y} - Y)|X\right] = \mathsf{Bi}_\beta^T X^T X \mathsf{Bi}_\beta + \mathrm{tr}(X\mathsf{Var}_\beta X^T) + \sigma_0^2$$

Taking the expectation gives the result.

□

Appendix C

Complete results of 2-by-2 experiments

C.1 Recap of framework

The system described by figure C.1 has been designed to assess the reconstruction system. Its inputs are parameters of the reconstruction or surroundings for which we want to know the influence (summarized table C.1). Its output are indicators of some qualities of the reconstruction method (summarized table C.2). A "2-by-2 experiment" is the run of the system when only 2 inputs are varying and all outputs are computed. The 2 inputs vary on a regular grid with 30 points for each (thus, 900 runs for each experiment).

To browse among the figures, use hyperlinks in

- the table of contents.
- the table C.3.
- the list of figures page 384.

Notation	Description	Place in the system
C_0	Kolmogorov "constant"	Reconstruction (Lagrangian model)
C_1	Fluctuation coefficient	Reconstruction (Lagrangian model)
ℓ	Spatial interaction length	Reconstruction (Lagrangian model)
N	Number of particles	Reconstruction (filtering)
σ^{add}	True observation noise	Simulation of observation
σ^{obs}	Guess of observation noise	Reconstruction (filtering)
σ^{X}	Discretization error in the Lagrangian model	Reconstruction (Lagrangian model)
σ^{V}	Default standard deviation of wind speed	Reconstruction (Lagrangian model)
τ	Integration time	Output computation

Table C.1 – Summary of input parameters for the sensitivity analysis

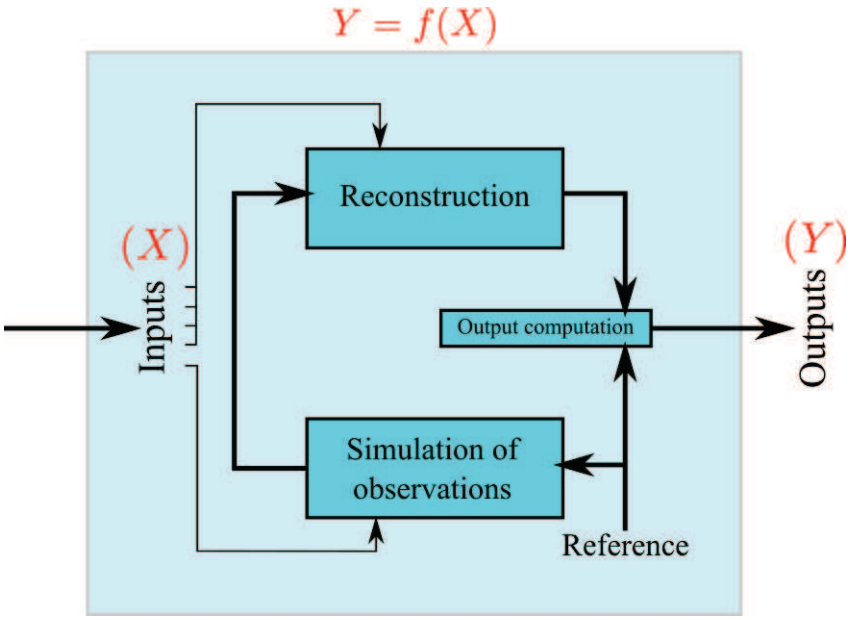


Figure C.1 – Diagram of the system on which is done the sensitivity analysis.

Notation	Description	Definition equation	Results
N_{G0}	Number of null potential	$N_{G0} = \lvert\{G = 0\}\rvert$	p.345
b	Slope of the wind PSD	$\Gamma(\kappa) = A\kappa^b$	p.349
r_k	Root-mean-squared error on the TKE	$\lVert\overline{k^{LS}(z,t)}^{\tau} - k^T(z,t')\rVert_2$	p.353
r_V	Root-mean-squared error on the wind	$r_V = \lVert\hat{V}_{z,t} - V_{z,t}^{ref}\rVert_2$	p.357
T_{exe}	Time of execution	$T_{exe} = t_{end} - t_{start}$	p.361

Table C.2 – Summary of output parameters for the sensitivity analysis

	C_0	C_1	ℓ	N	σ^{add}	σ^{obs}	σ^V	σ^X	τ
C_0		p.303	p.306					p.309	
C_1	p.303								
ℓ	p.306			p.312	p.315	p.318			
N			p.312		p.321	p.324			p.327
σ^{add}			p.315	p.321		p.330			p.333
σ^{obs}			p.318	p.324	p.330			p.336	p.339
σ^V								p.342	
σ^X	p.309					p.336	p.342		
τ				p.327	p.333	p.339			

Table C.3 – Couples of inputs experimented: results are on the indicated page (hyperlink). This table is a copy of 7.1.

C.2 Results by experiment (couple of inputs)

An experiment consists in the computation of all 5 outputs when only two inputs vary on a regular grid. Table C.3 gives all the experiments carried out. There are 13 experiments in total.

C.2.1 C_0 and C_1

In this experiment, only C_0 and C_1 vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

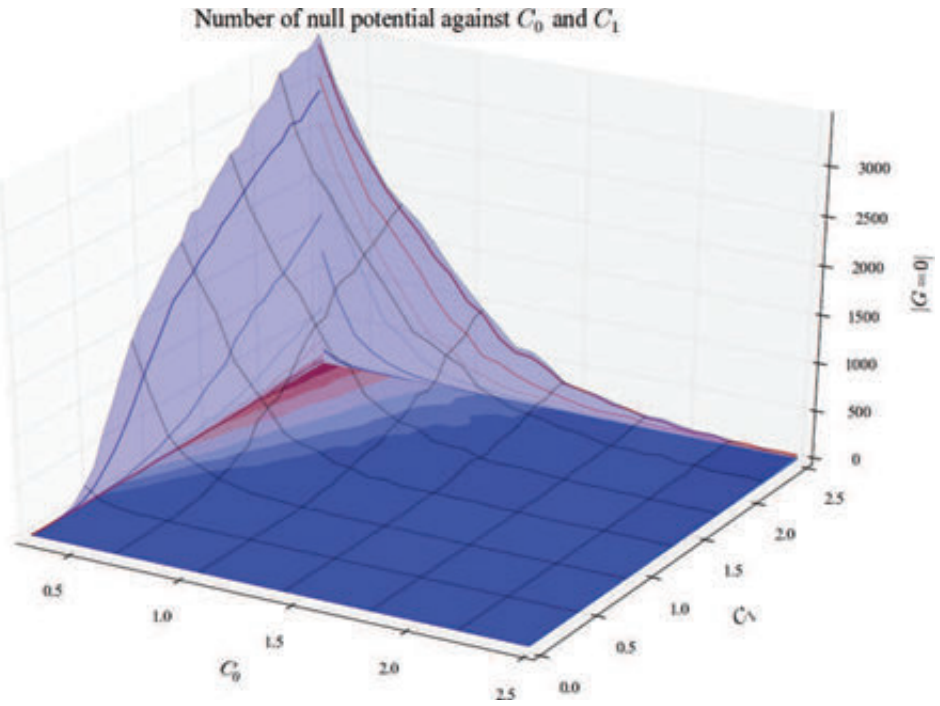


Figure C.2 – Number of null potential when C_0 and C_1 vary.

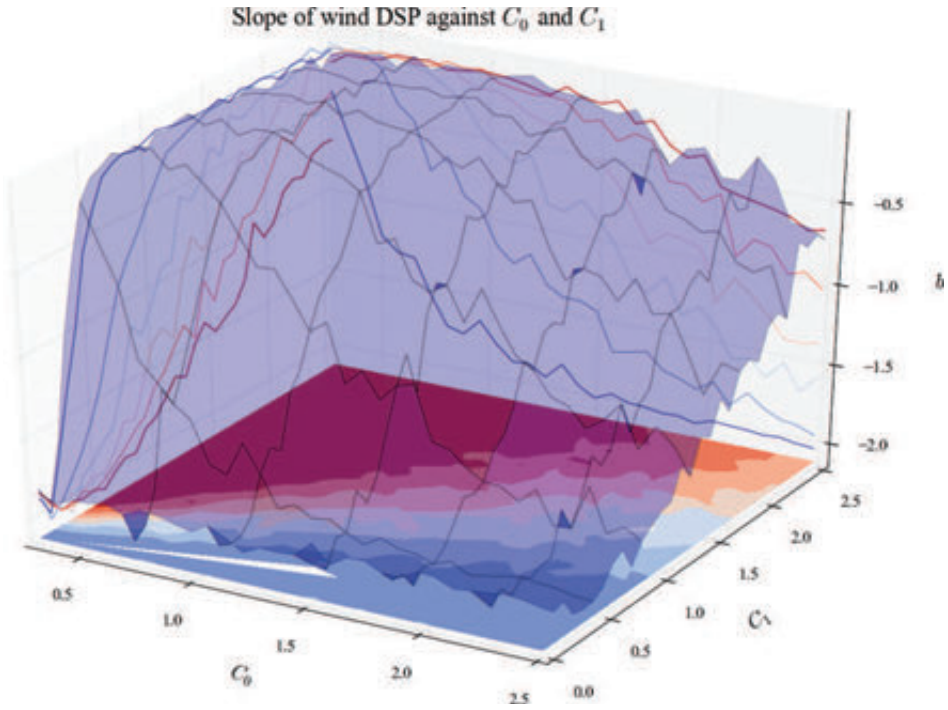


Figure C.3 – PSD slope when C_0 and C_1 vary.

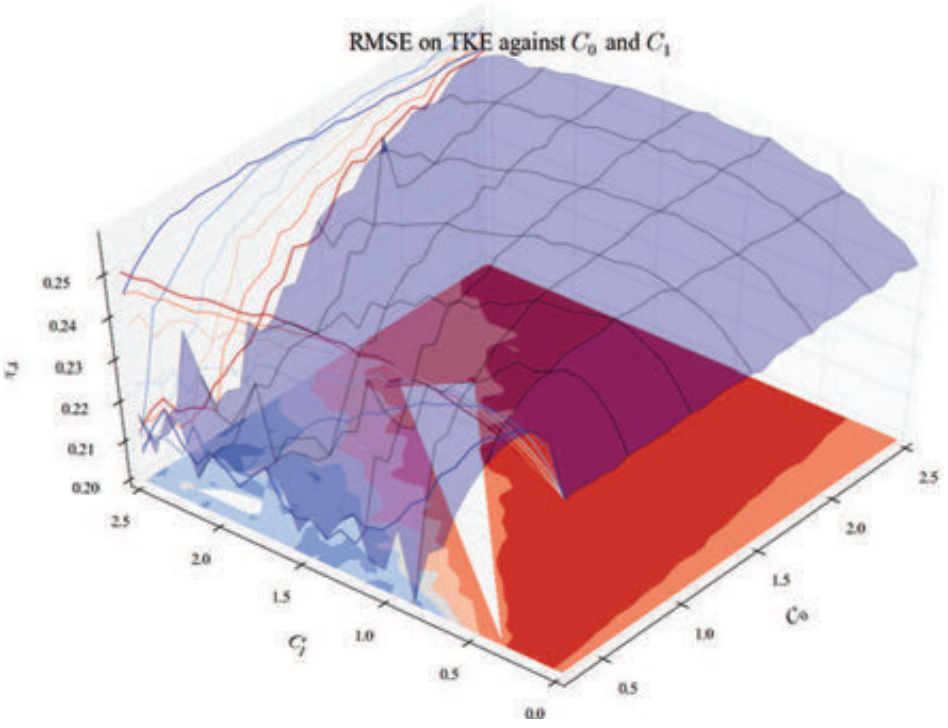


Figure C.4 – RMSE on TKE when C_0 and C_1 vary.

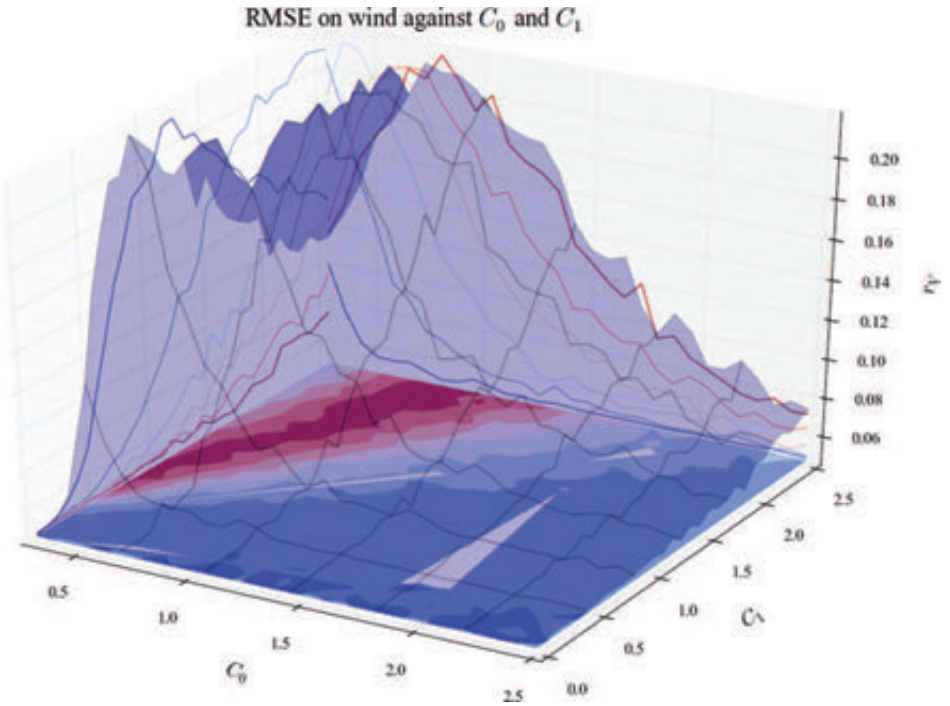


Figure C.5 – RMSE on wind when C_0 and C_1 vary.

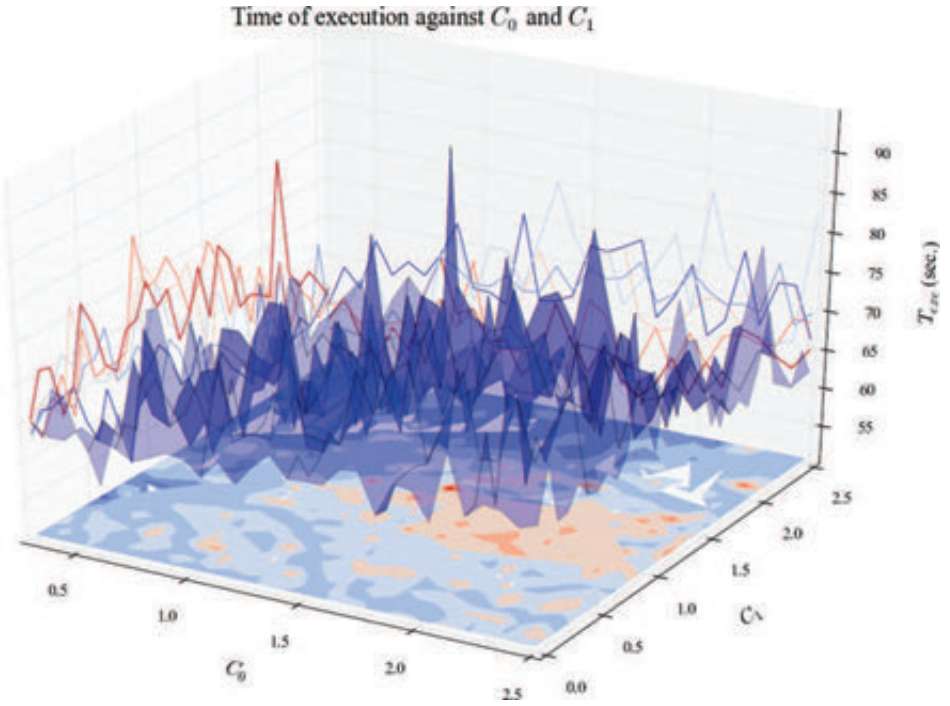


Figure C.6 – Execution time when C_0 and C_1 vary.

C.2.2 C_0 and ℓ

In this experiment, only C_0 and ℓ vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

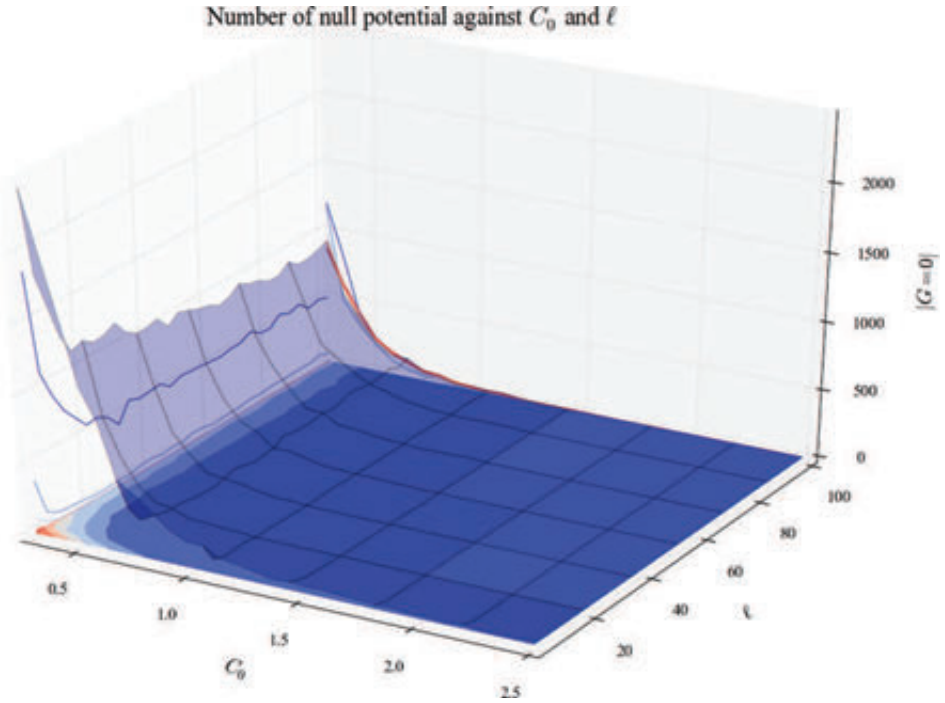


Figure C.7 – Number of null potential when C_0 and ℓ vary.

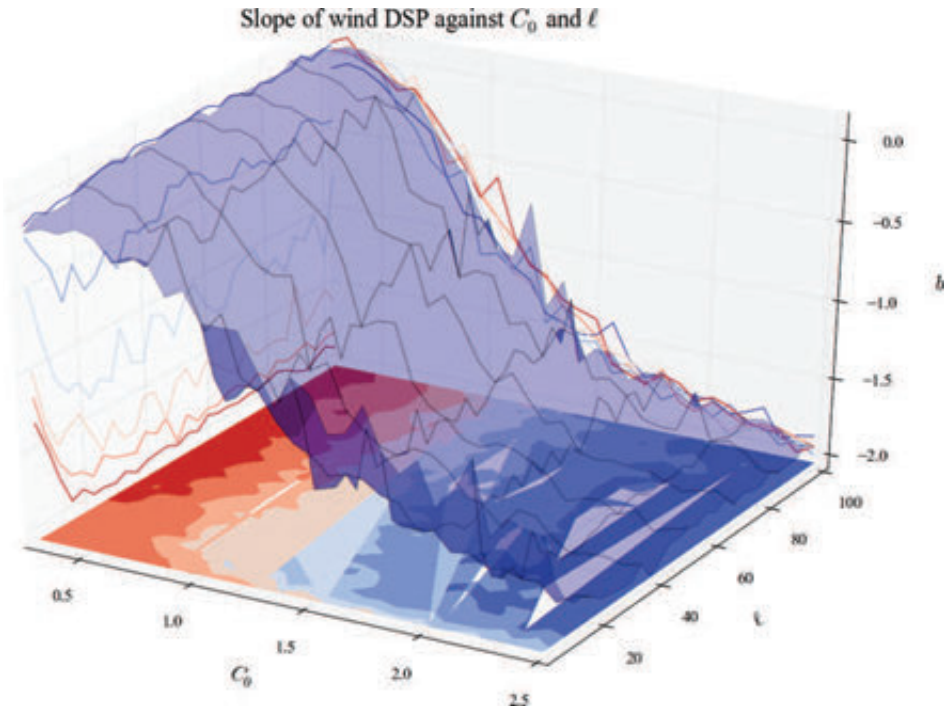


Figure C.8 – PSD slope when C_0 and ℓ vary.

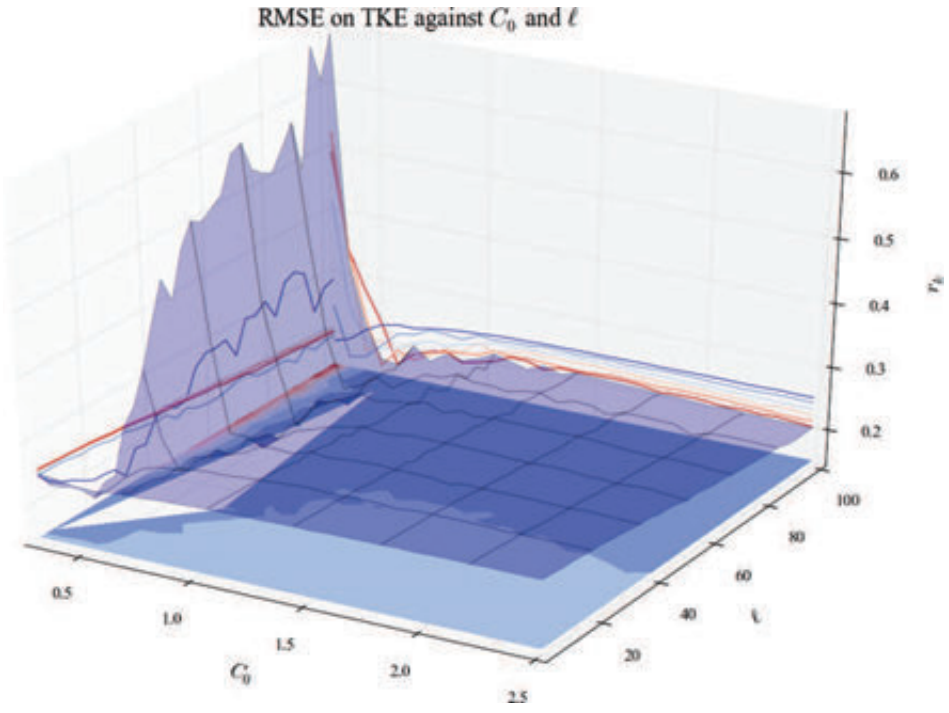


Figure C.9 – RMSE on TKE when C_0 and ℓ vary.

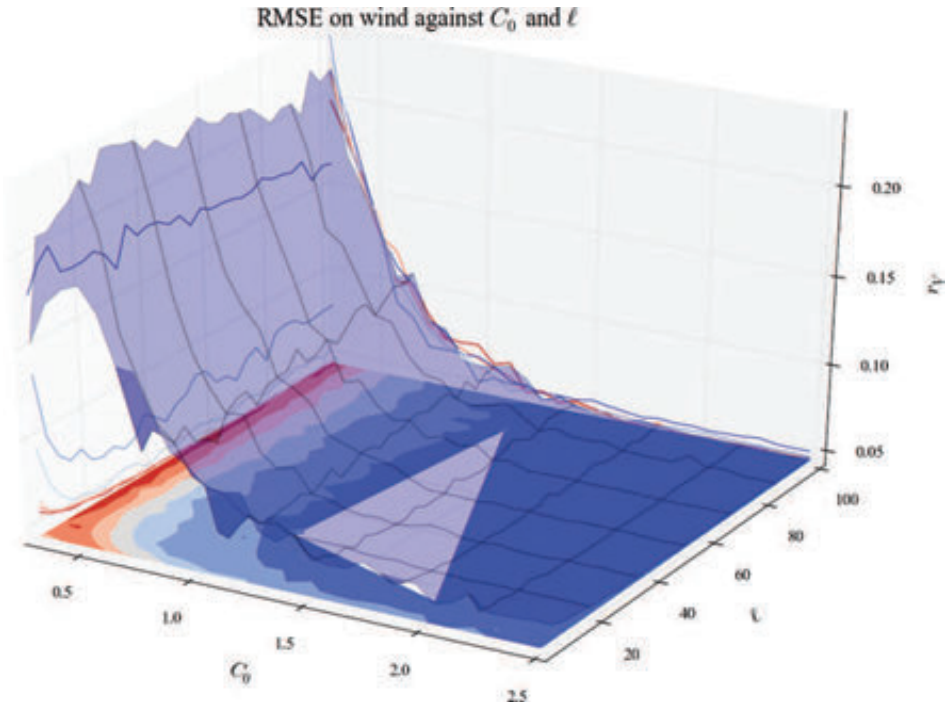


Figure C.10 – RMSE on wind when C_0 and ℓ vary.

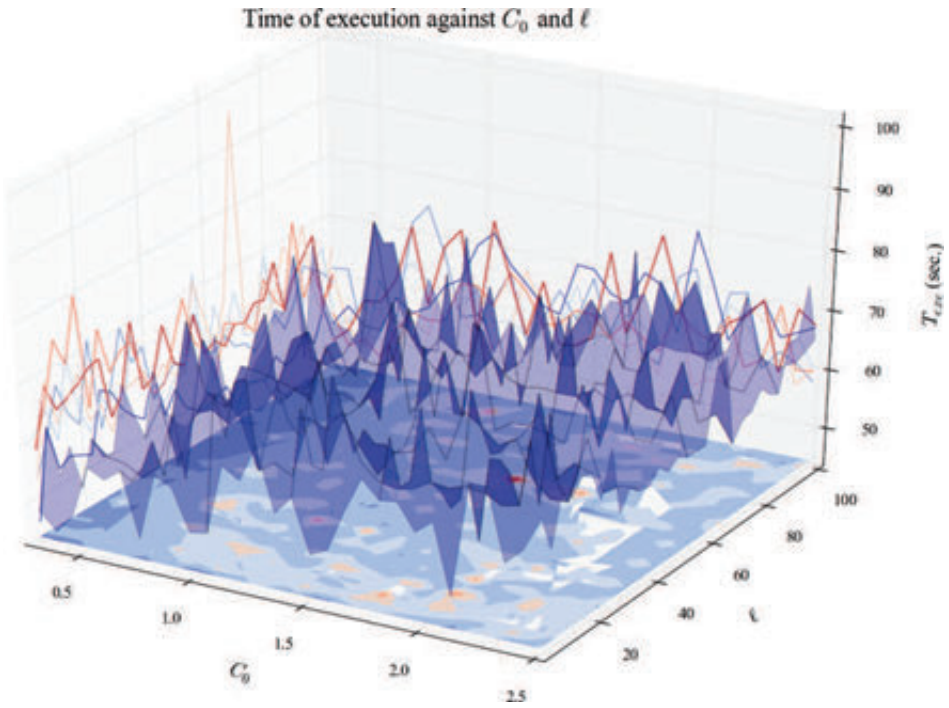


Figure C.11 – Execution time when C_0 and ℓ vary.

C.2.3 C_0 and σ^X

In this experiment, only C_0 and σ^X vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

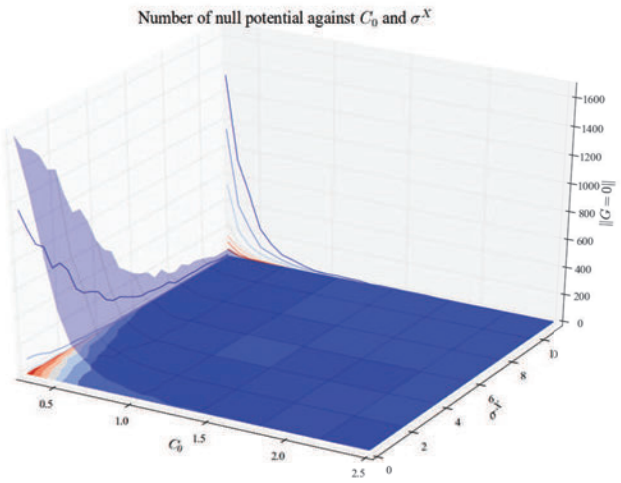


Figure C.12 – Number of null potential when C_0 and σ^X vary.

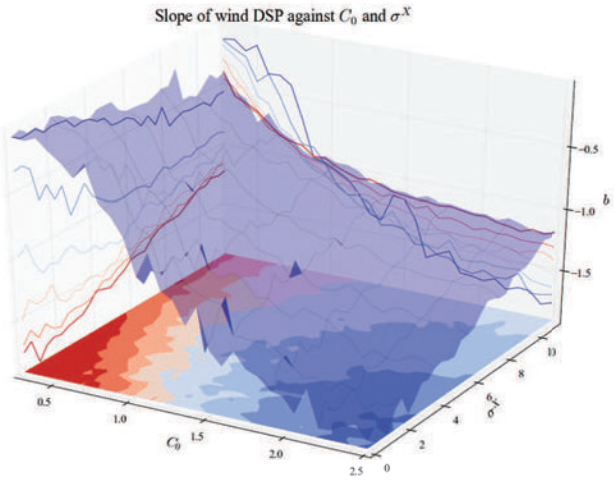


Figure C.13 – PSD slope when C_0 and σ^X vary.

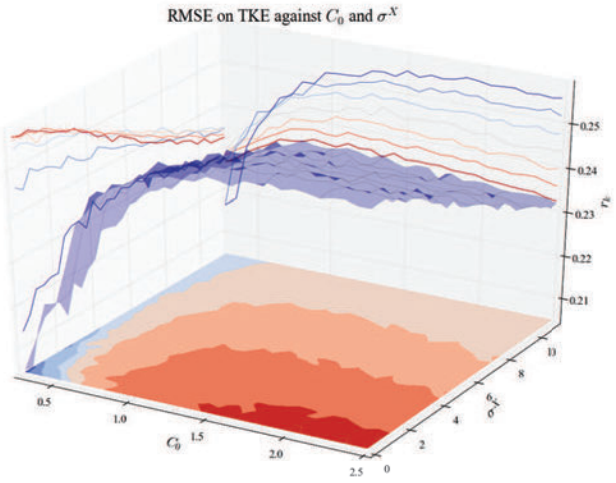


Figure C.14 – RMSE on TKE when C_0 and σ^X vary.

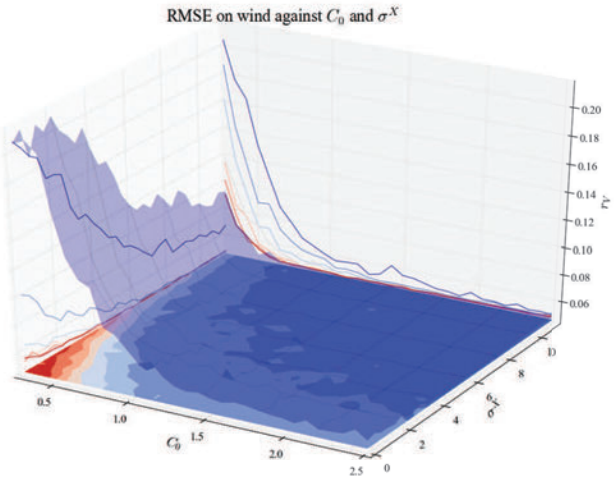


Figure C.15 – RMSE on wind when C_0 and σ^X vary.

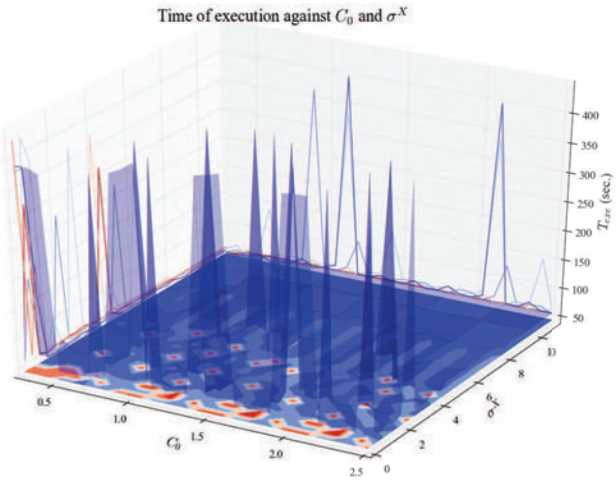


Figure C.16 – Execution time when C_0 and σ^X vary.

C.2.4 ℓ and N

In this experiment, only ℓ and N vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

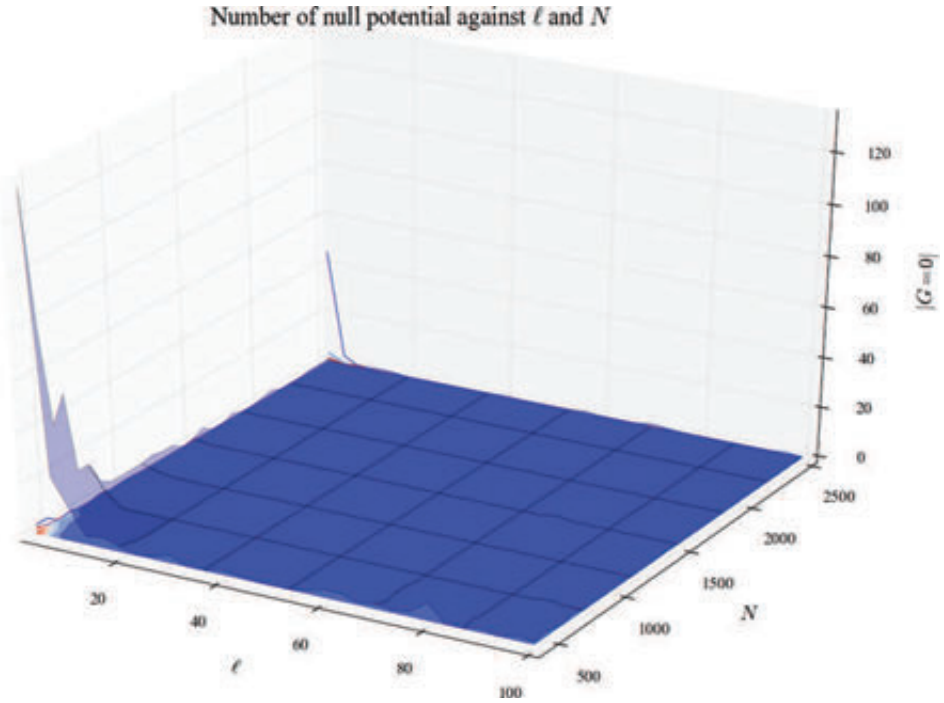


Figure C.17 – Number of null potential when ℓ and N vary.

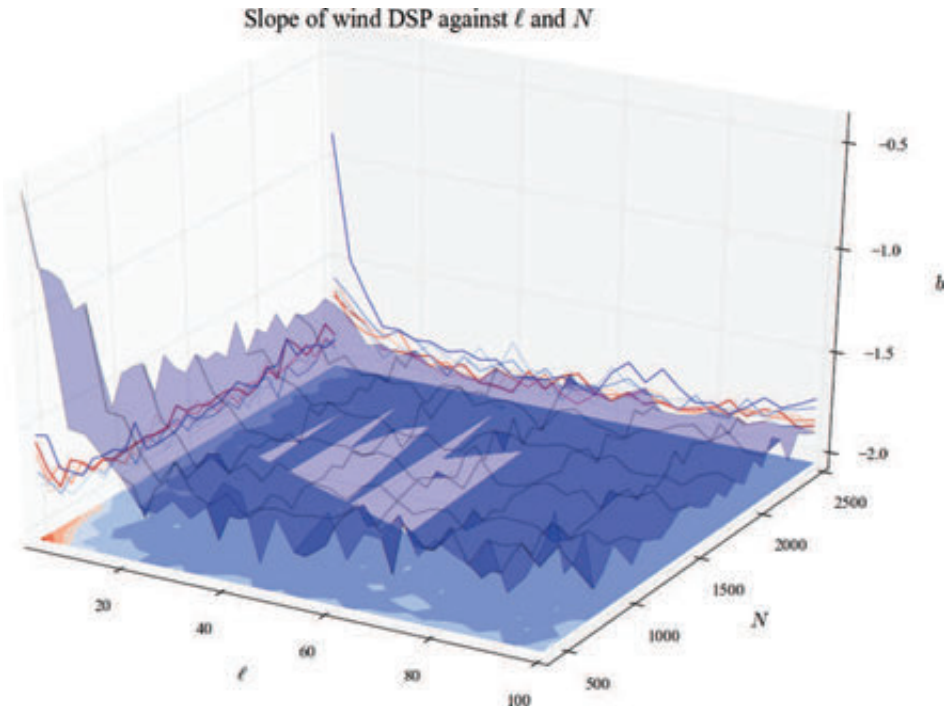


Figure C.18 – PSD slope when ℓ and N vary.

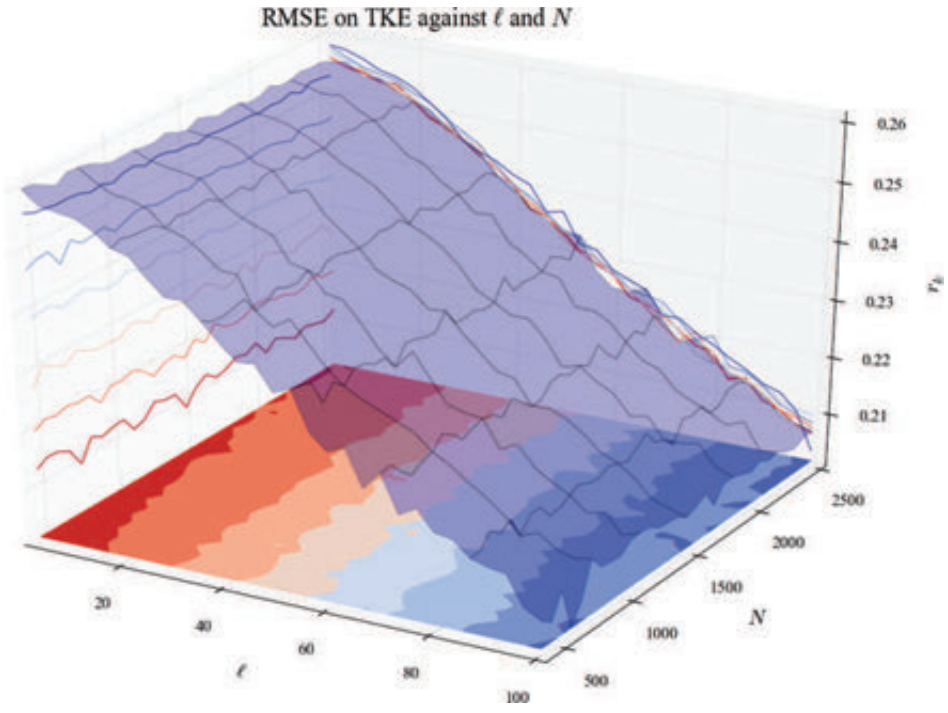


Figure C.19 – RMSE on TKE when ℓ and N vary.

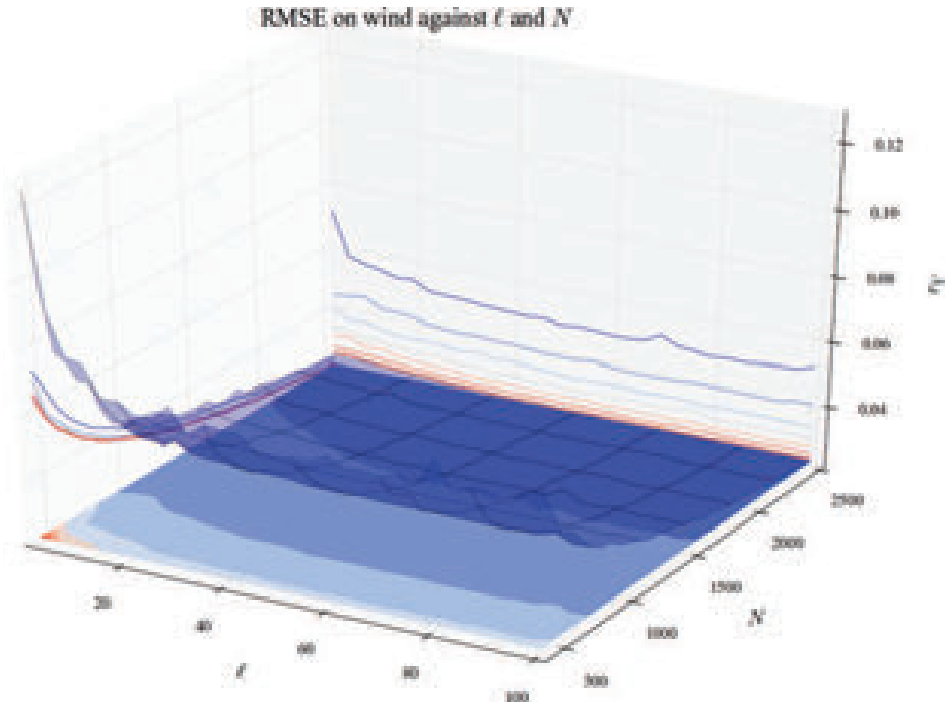


Figure C.20 – RMSE on wind when ℓ and N vary.

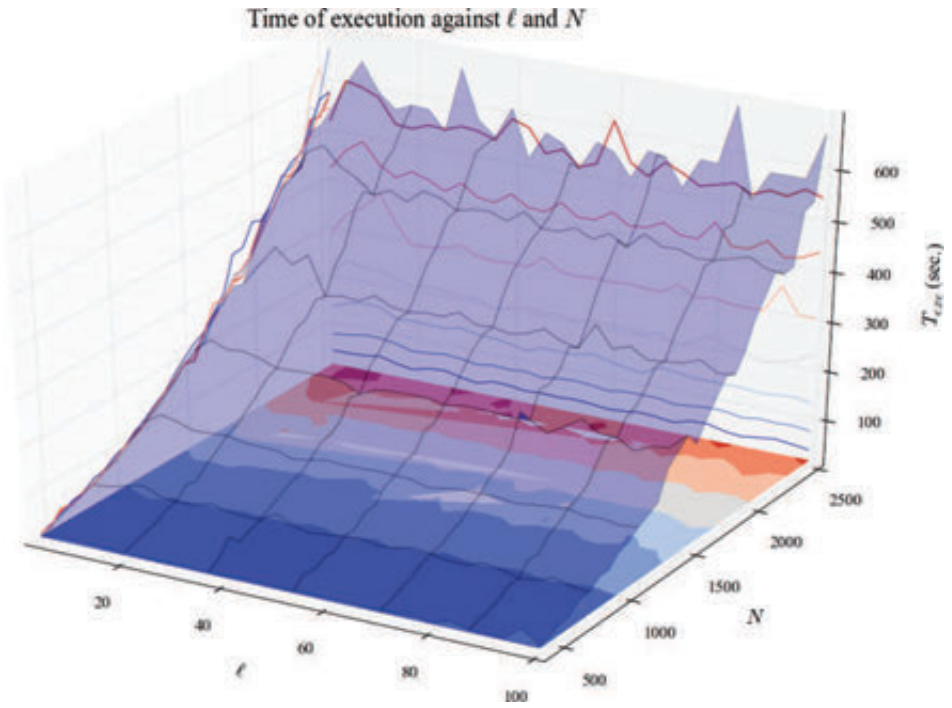


Figure C.21 – Execution time when ℓ and N vary.

C.2.5 ℓ and σ^{add}

In this experiment, only ℓ and σ^{add} vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

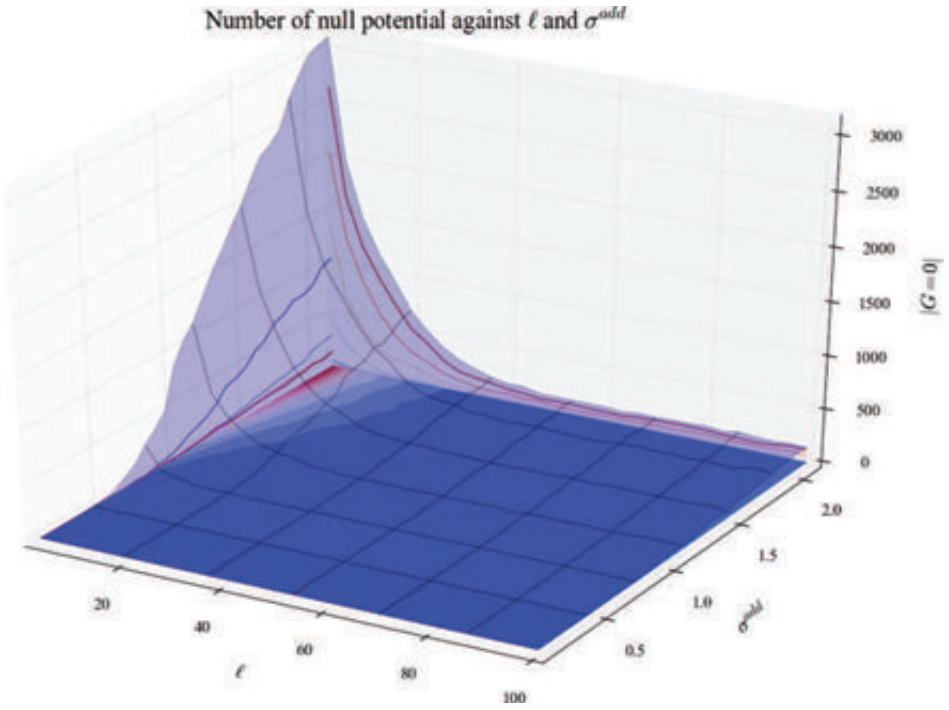


Figure C.22 – Number of null potential when ℓ and σ^{add} vary.

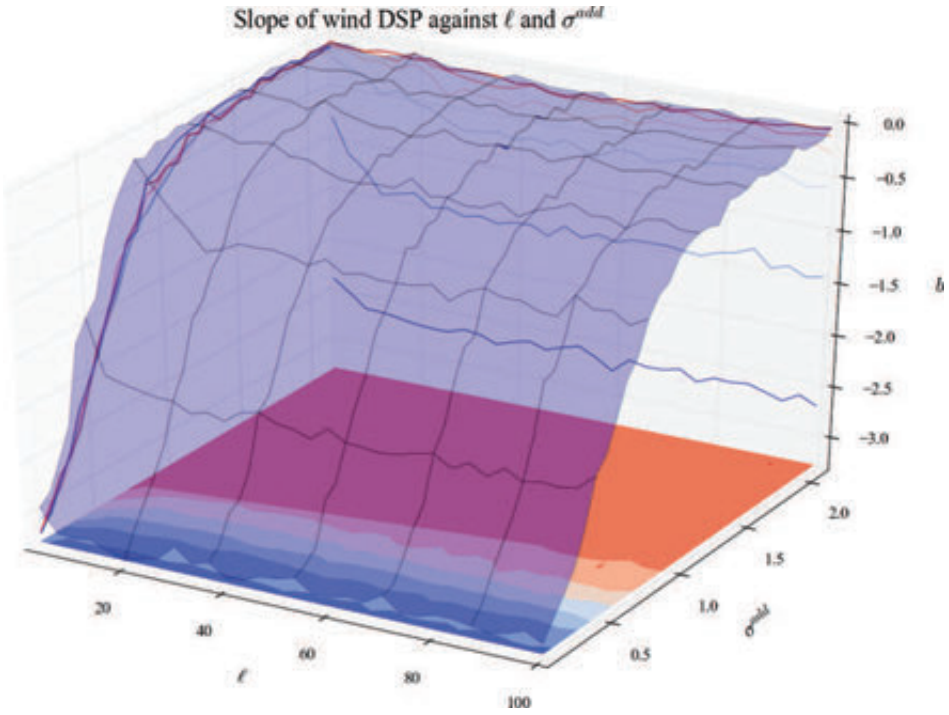


Figure C.23 – PSD slope when ℓ and σ^{add} vary.

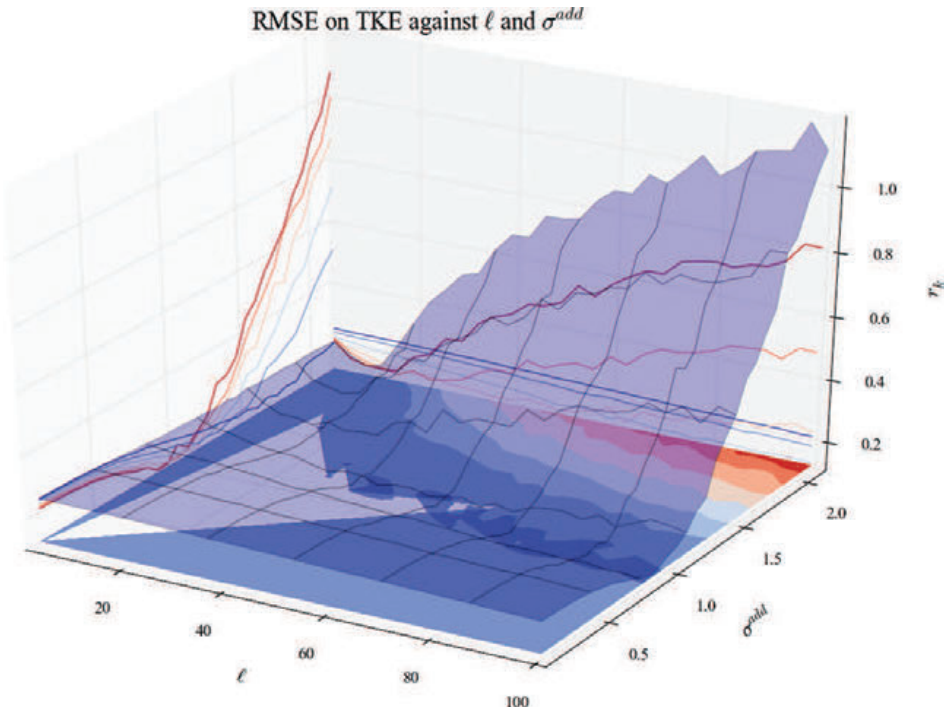


Figure C.24 – RMSE on TKE when ℓ and σ^{add} vary.

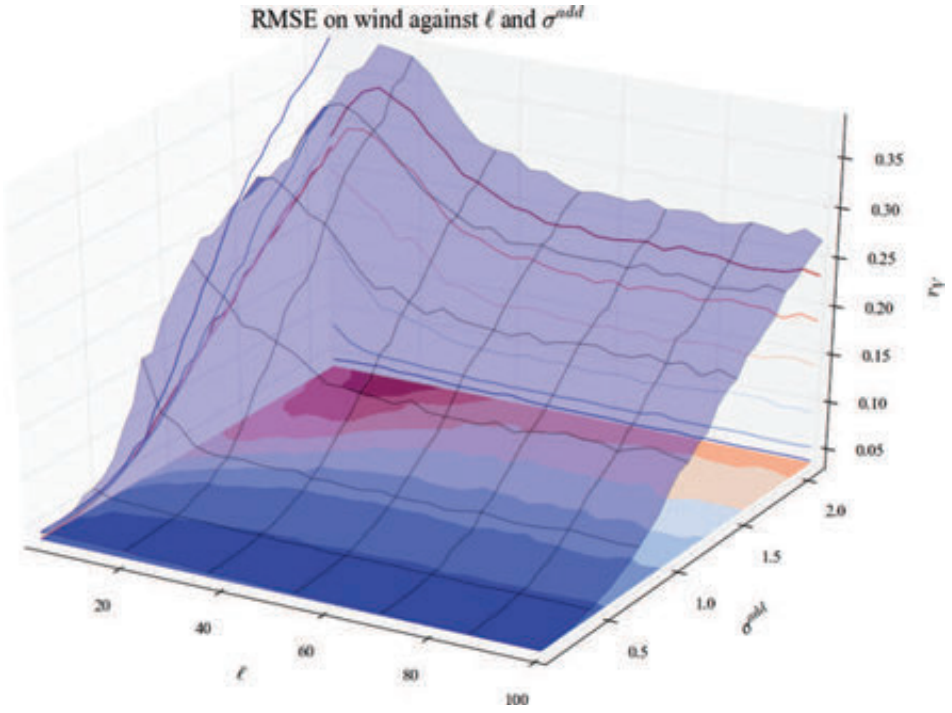


Figure C.25 – RMSE on wind when ℓ and σ^{add} vary.

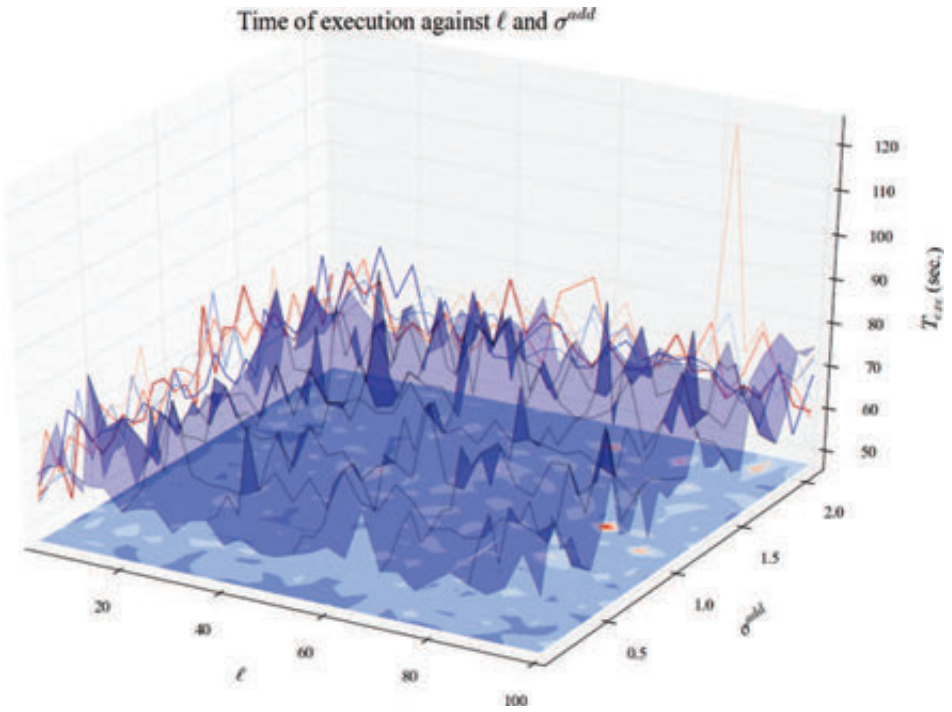


Figure C.26 – Execution time when ℓ and σ^{add} vary.

C.2.6 ℓ and σ^{obs}

In this experiment, only ℓ and σ^{obs} vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

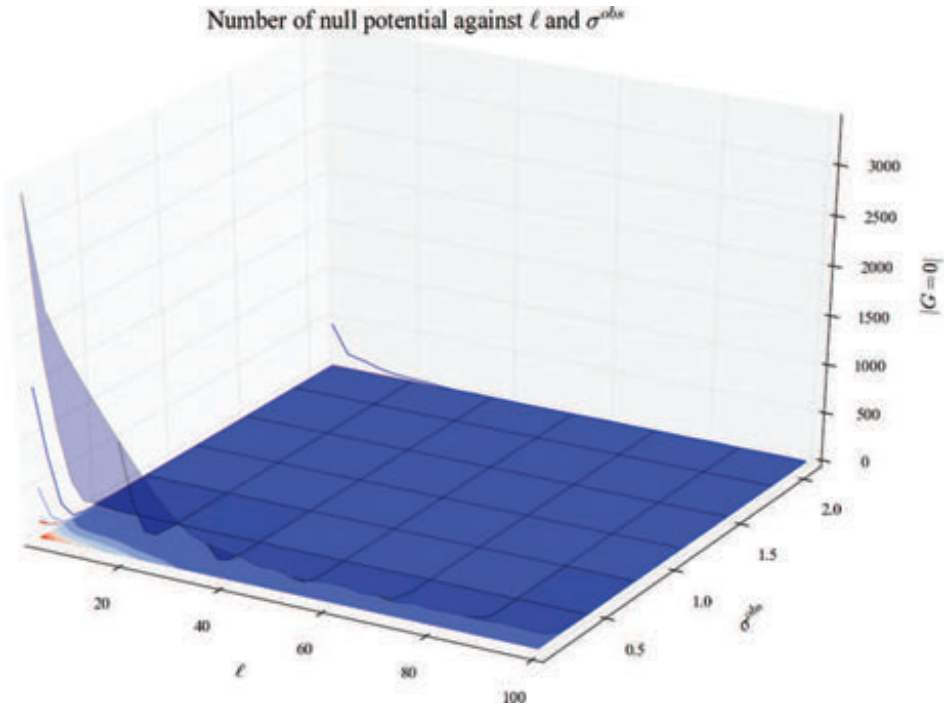


Figure C.27 – Number of null potential when ℓ and σ^{obs} vary.

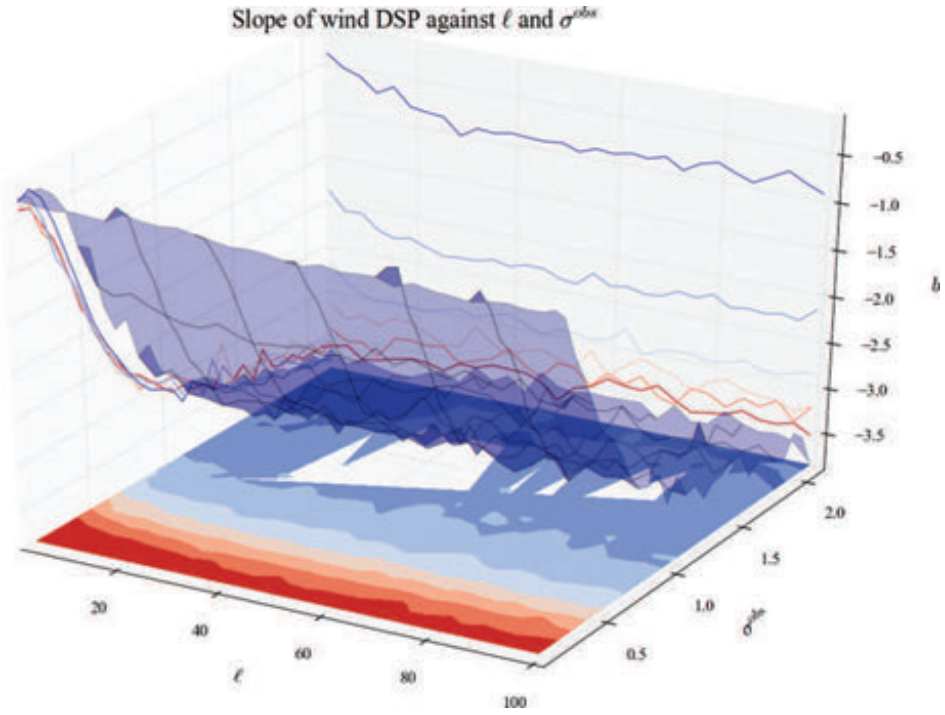


Figure C.28 – PSD slope when ℓ and σ^{obs} vary.

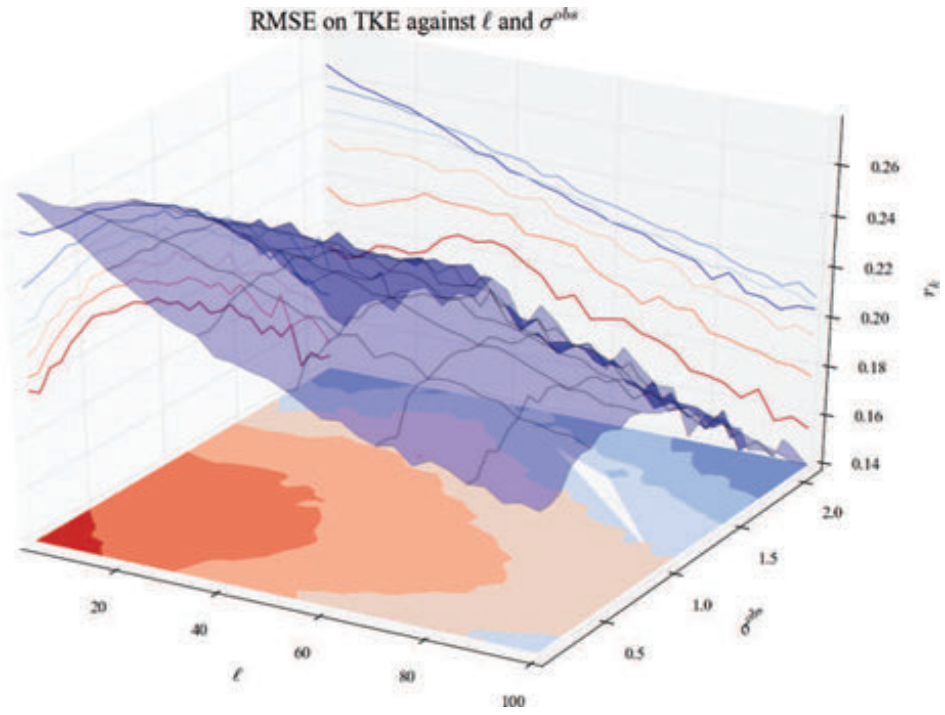


Figure C.29 – RMSE on TKE when ℓ and σ^{obs} vary.

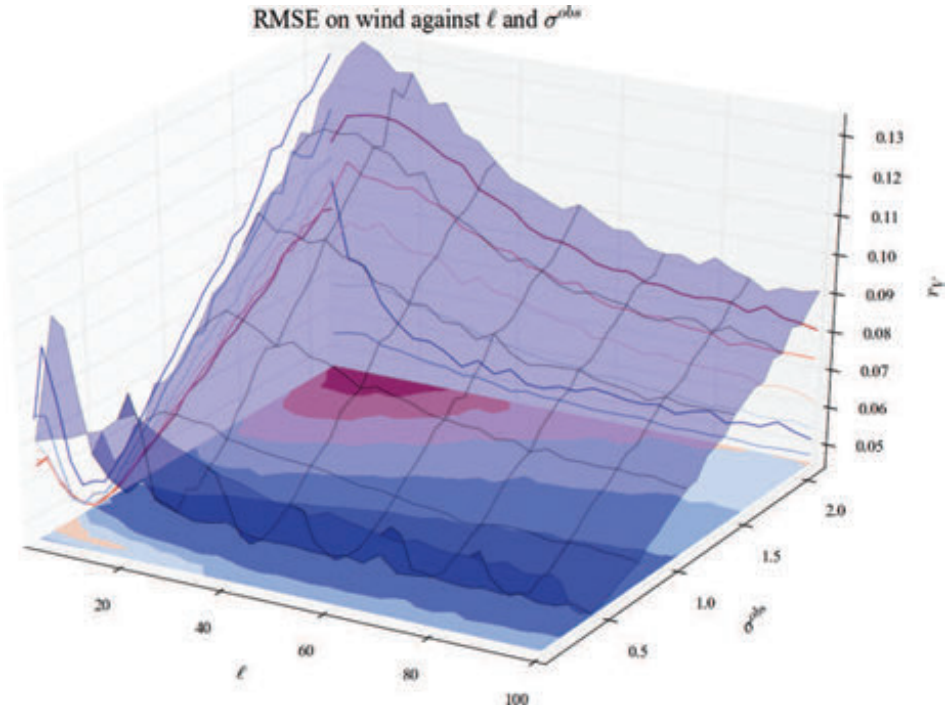


Figure C.30 – RMSE on wind when ℓ and σ^{obs} vary.

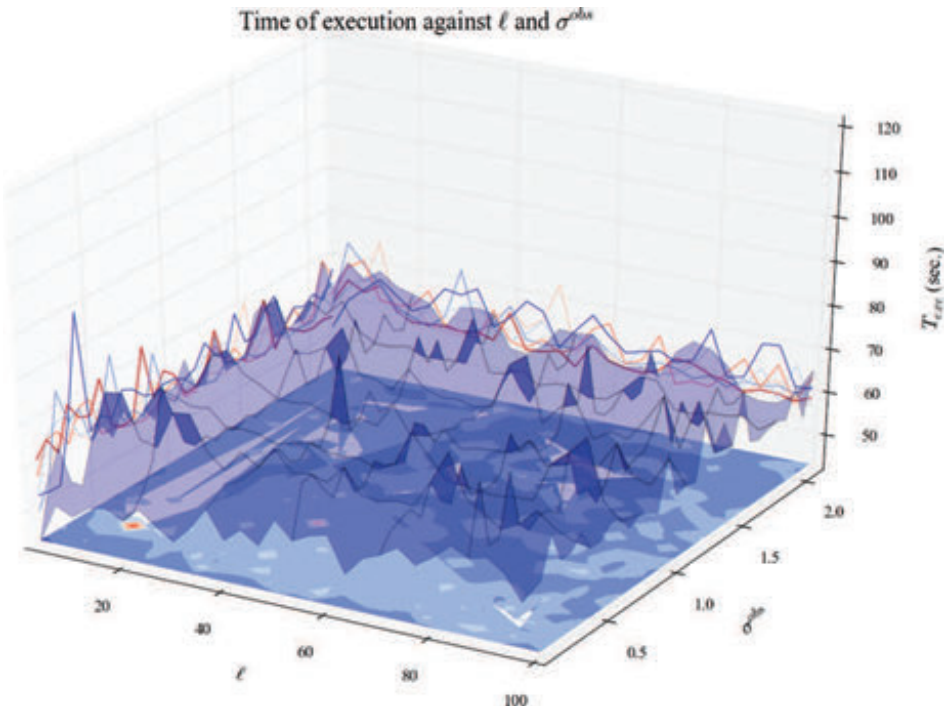


Figure C.31 – Execution time when ℓ and σ^{obs} vary.

C.2.7 N and σ^{add}

In this experiment, only N and σ^{add} vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

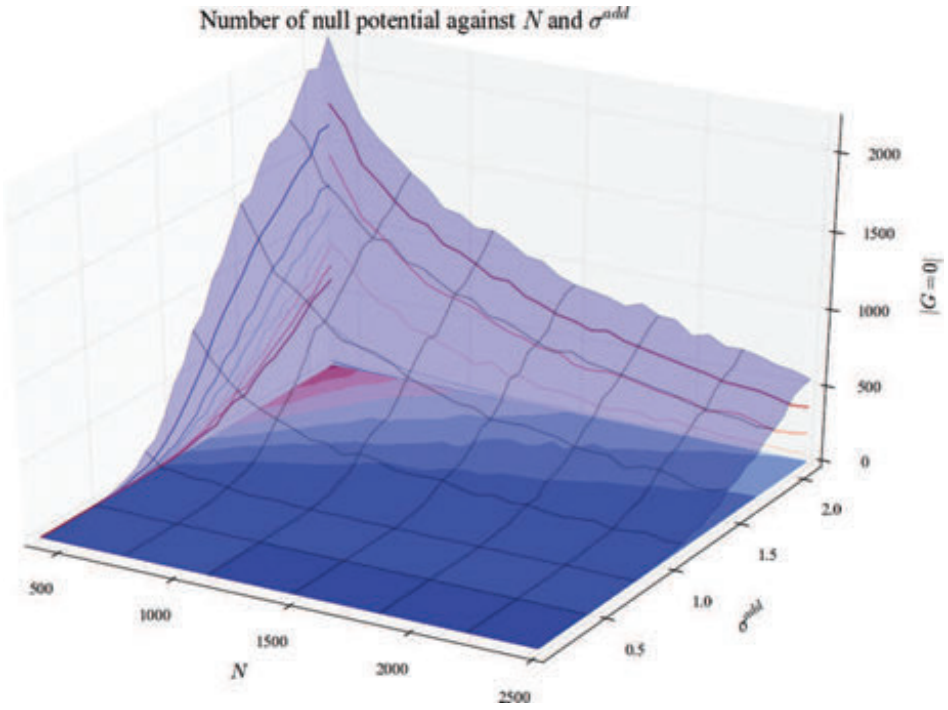


Figure C.32 – Number of null potential when N and σ^{add} vary.

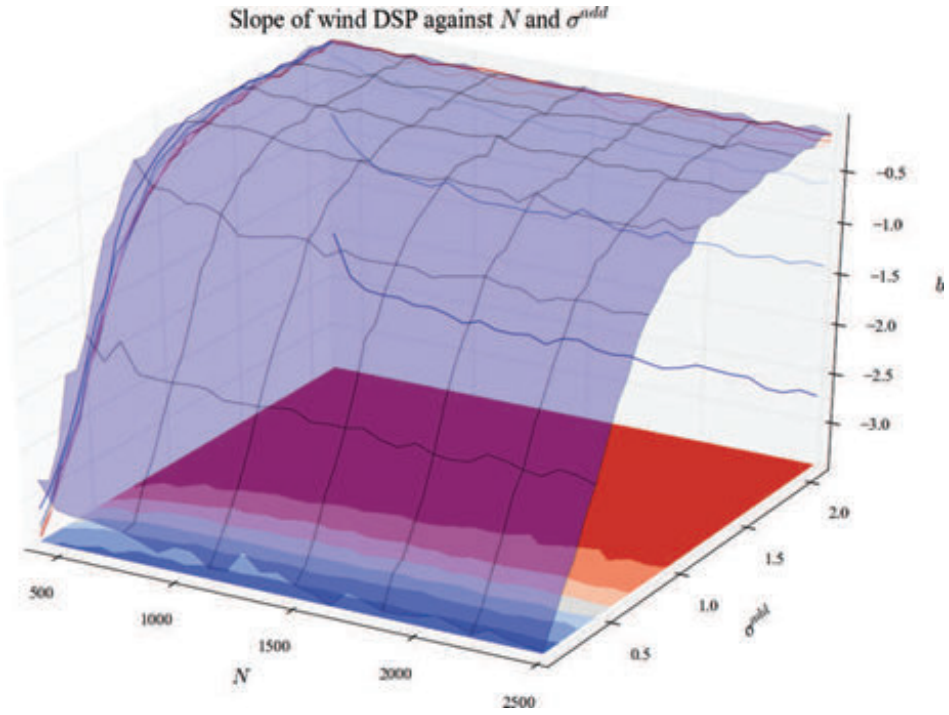


Figure C.33 – PSD slope when N and σ^{add} vary.

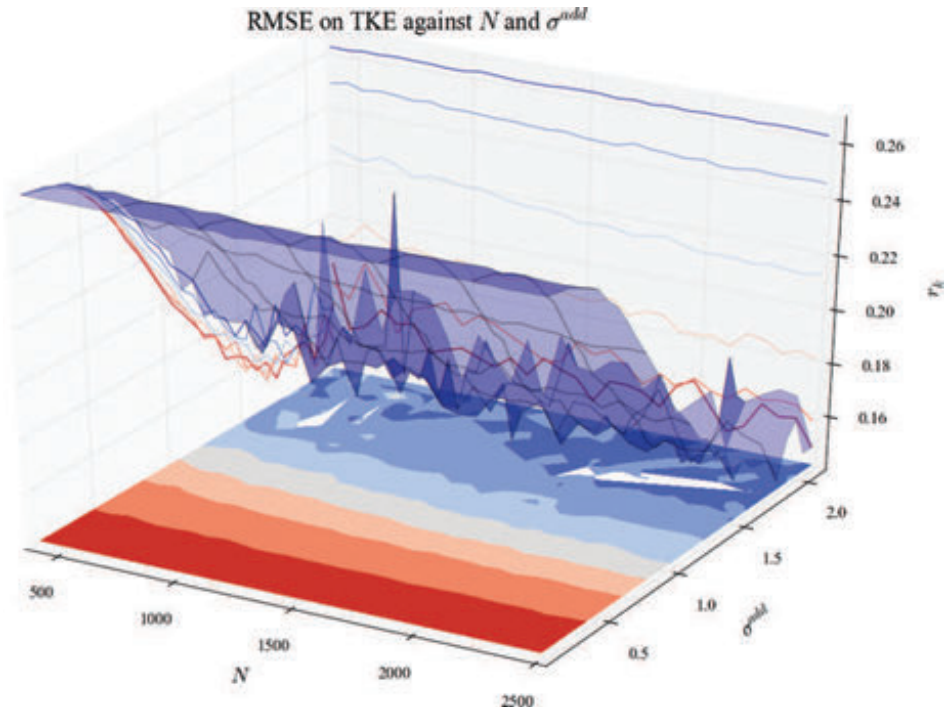


Figure C.34 – RMSE on TKE when N and σ^{add} vary.

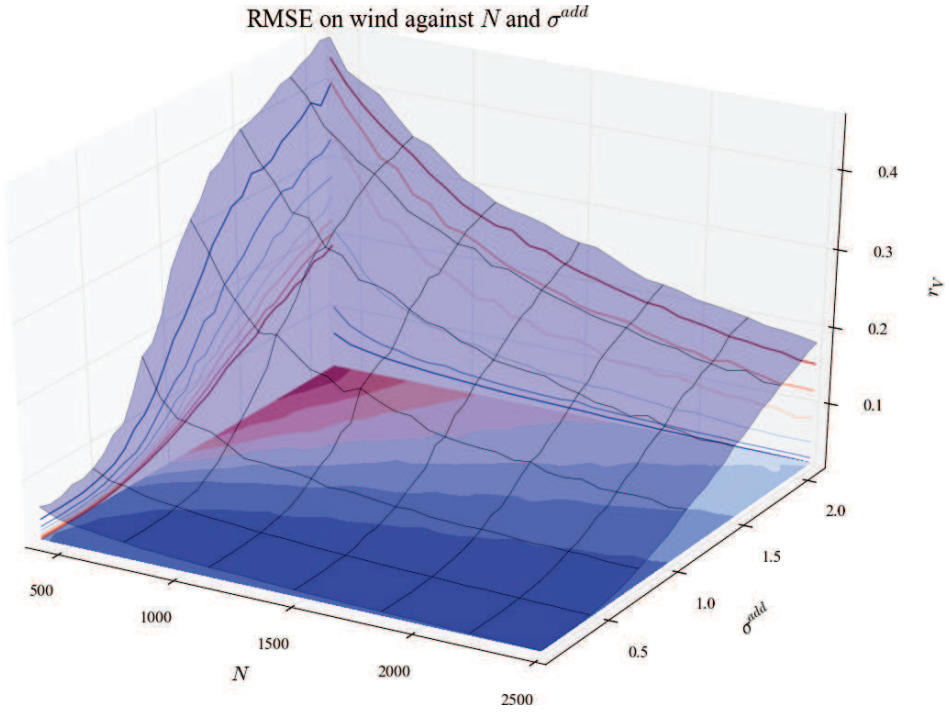


Figure C.35 – RMSE on wind when N and σ^{add} vary.

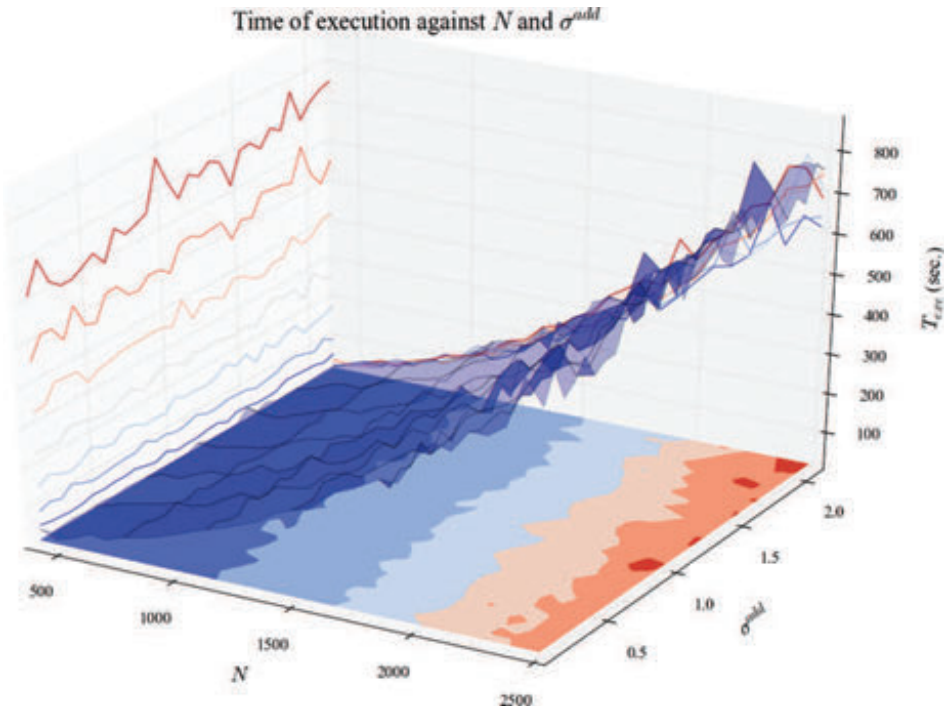


Figure C.36 – Execution time when N and σ^{add} vary.

C.2.8 N and σ^{obs}

In this experiment, only N and σ^{obs} vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

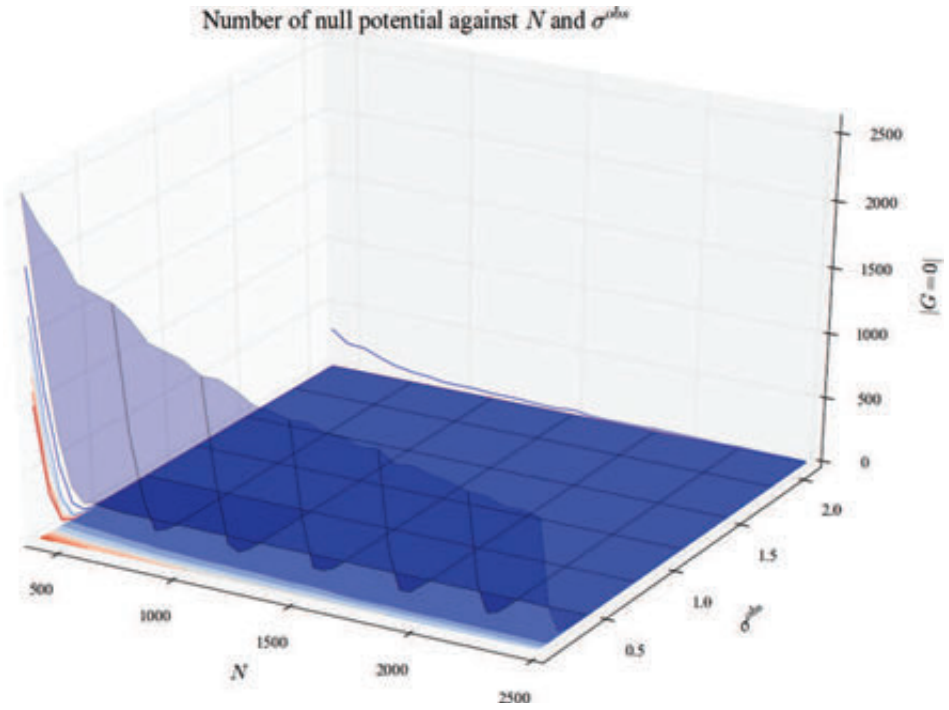


Figure C.37 – Number of null potential when N and σ^{obs} vary.

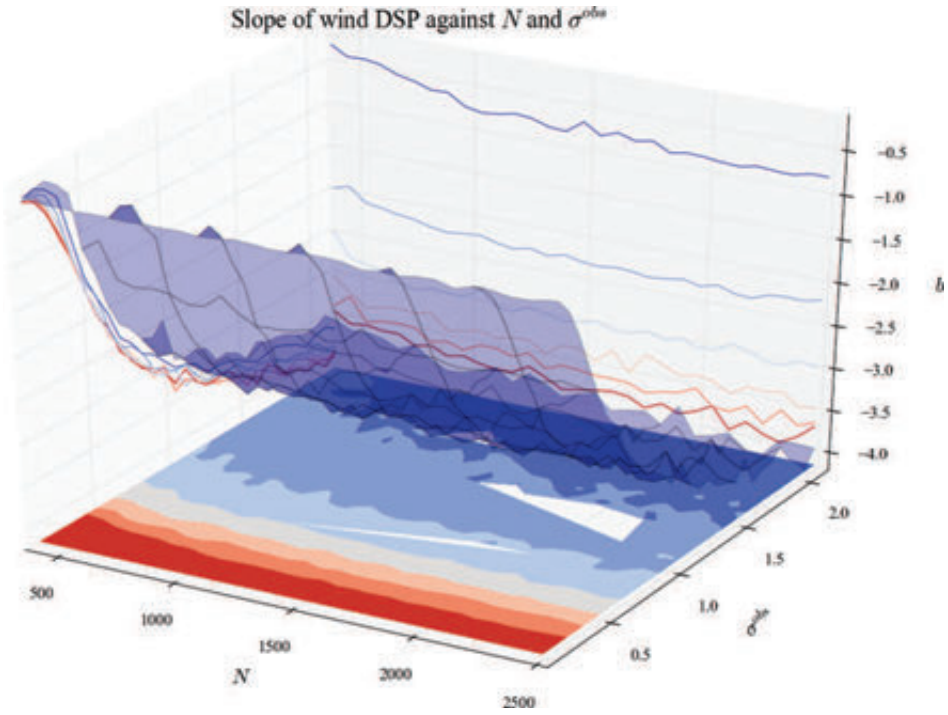


Figure C.38 – PSD slope when N and σ^{obs} vary.

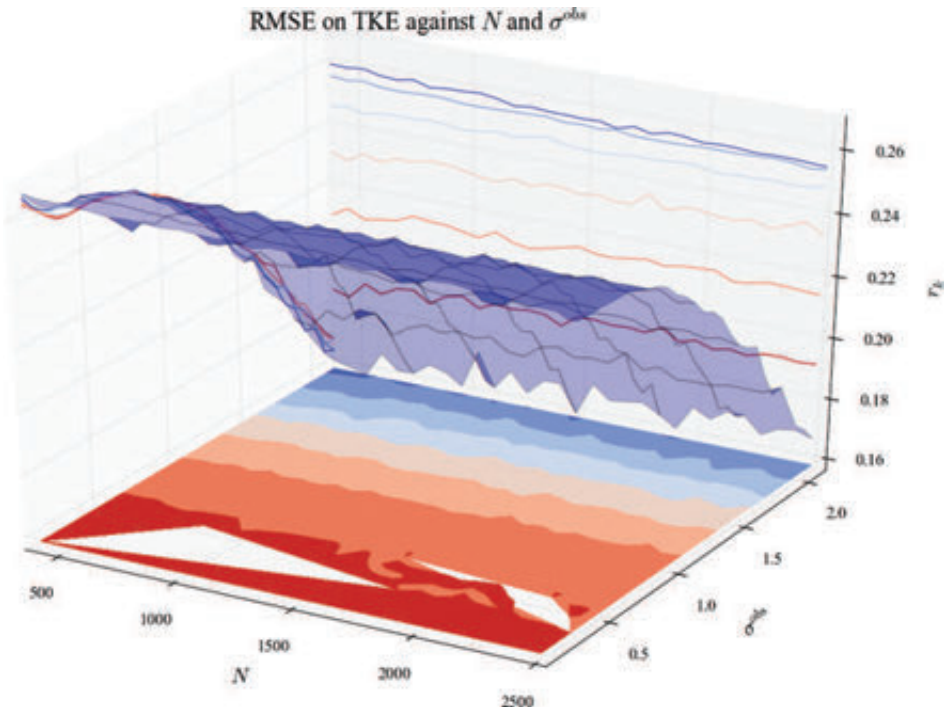


Figure C.39 – RMSE on TKE when N and σ^{obs} vary.

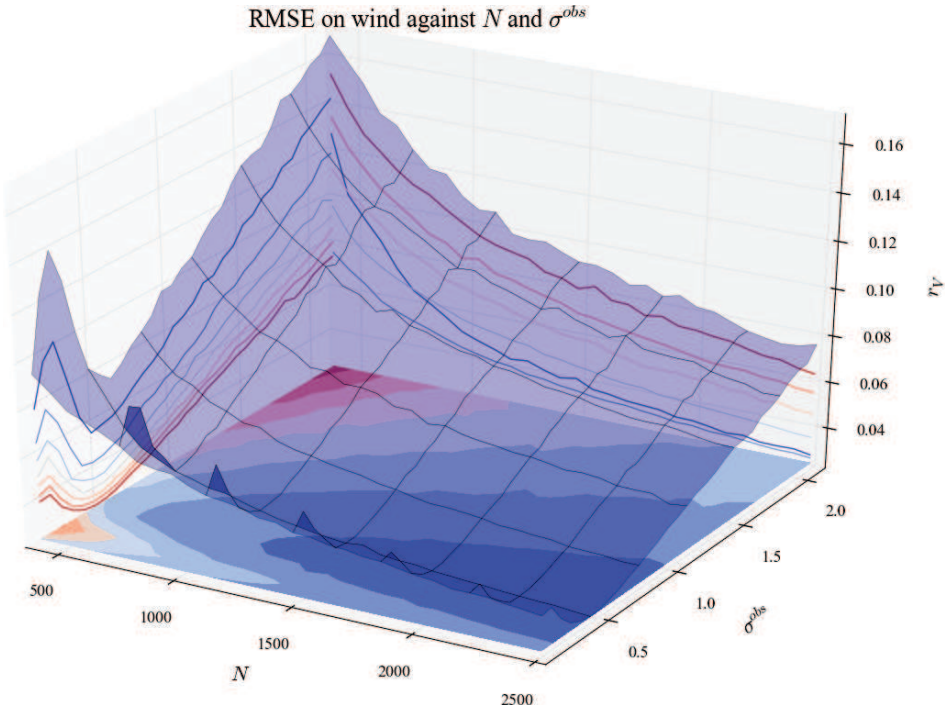


Figure C.40 – RMSE on wind when N and σ^{obs} vary.

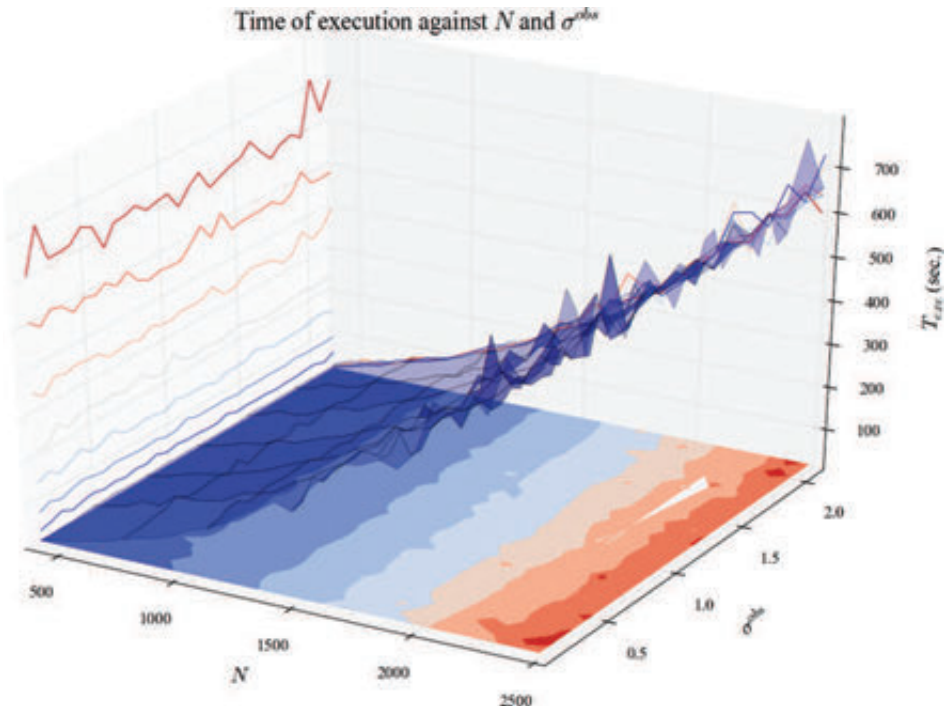


Figure C.41 – Execution time when N and σ^{obs} vary.

C.2.9 N and τ

In this experiment, only N and τ vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

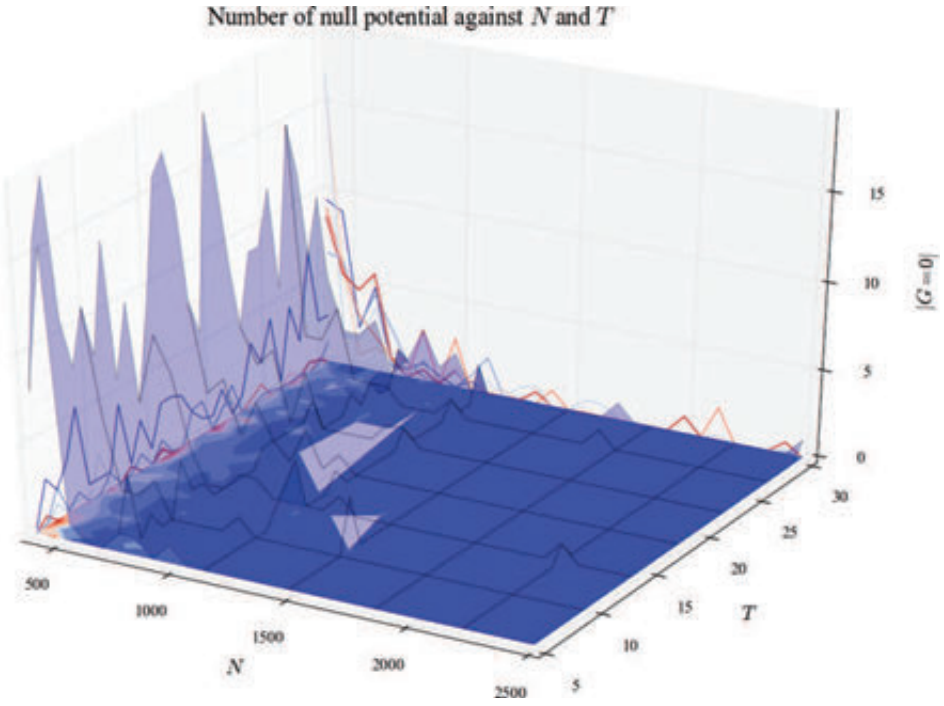


Figure C.42 – Number of null potential when N and τ vary.

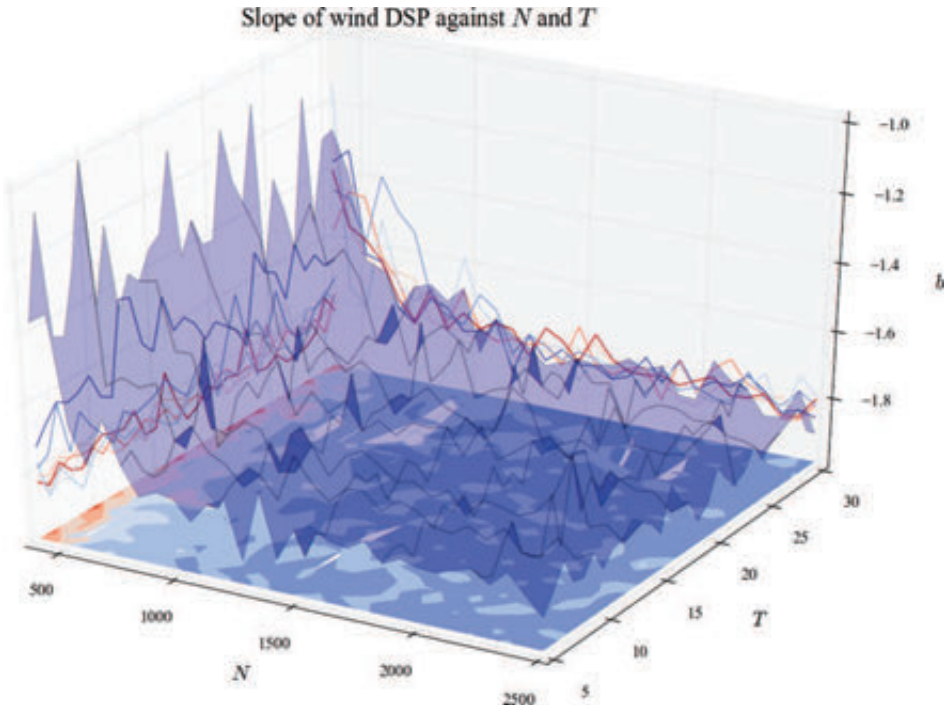


Figure C.43 – PSD slope when N and τ vary.

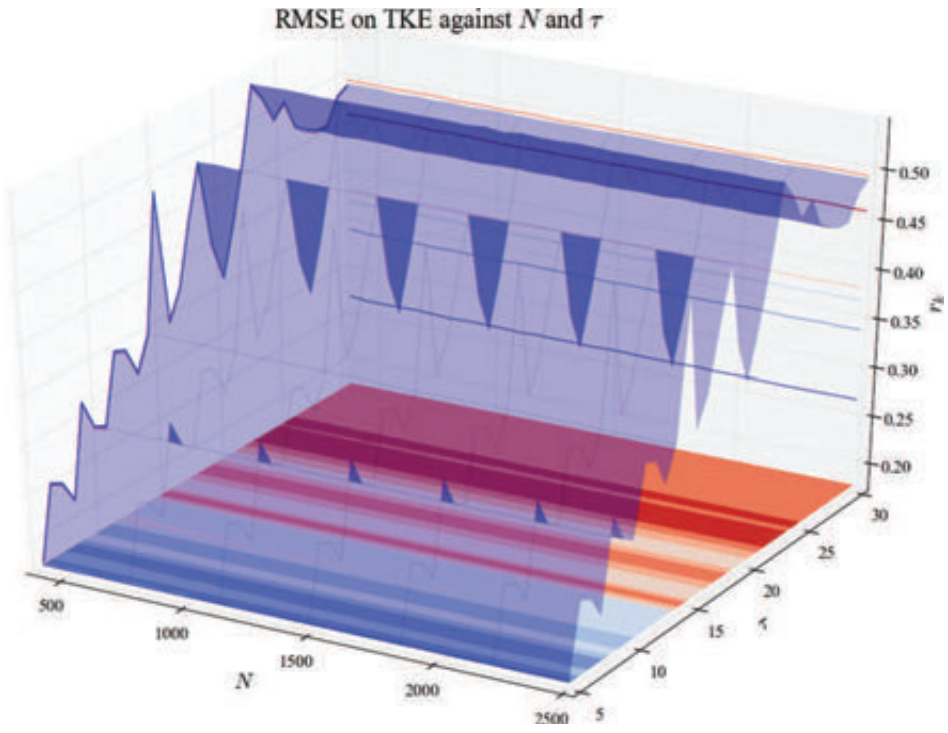


Figure C.44 – RMSE on TKE when N and τ vary.

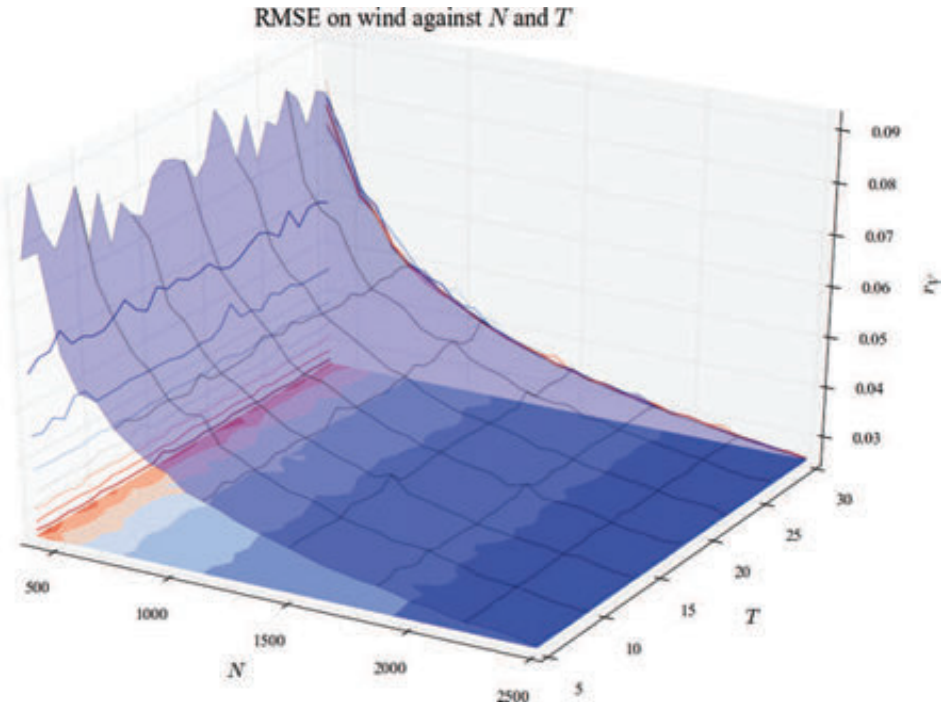


Figure C.45 – RMSE on wind when N and τ vary.

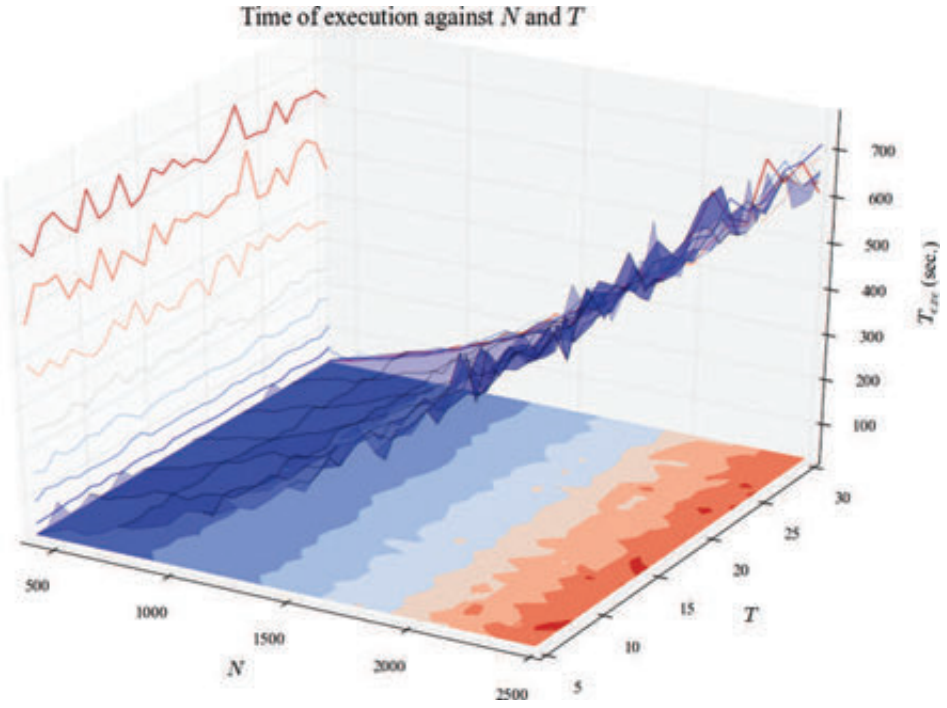


Figure C.46 – Execution time when N and τ vary.

C.2.10 σ^{add} and σ^{obs}

In this experiment, only σ^{add} and σ^{obs} vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

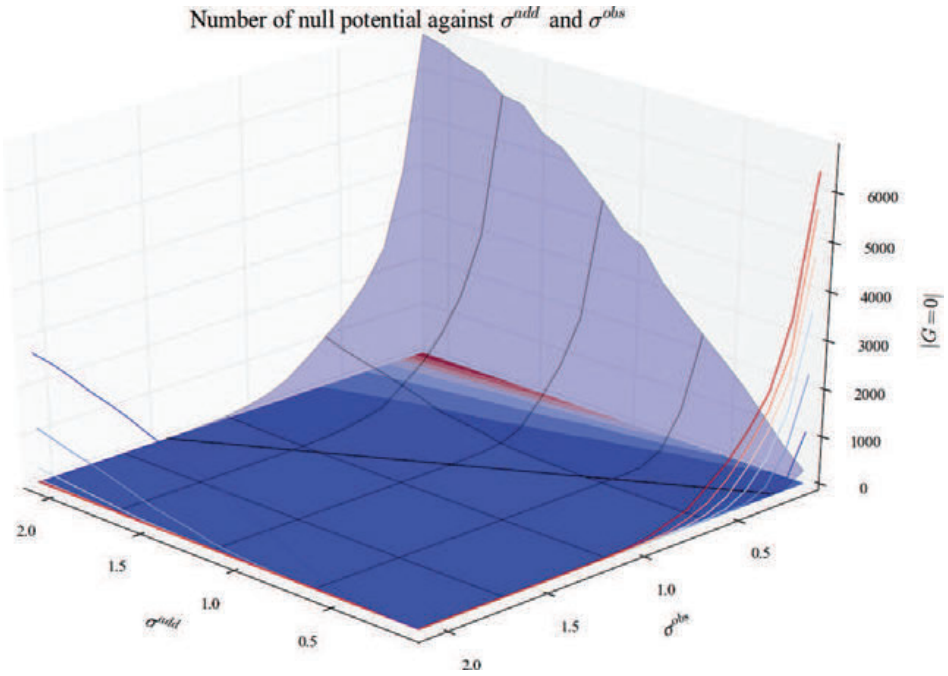


Figure C.47 – Number of null potential when σ^{add} and σ^{obs} vary.

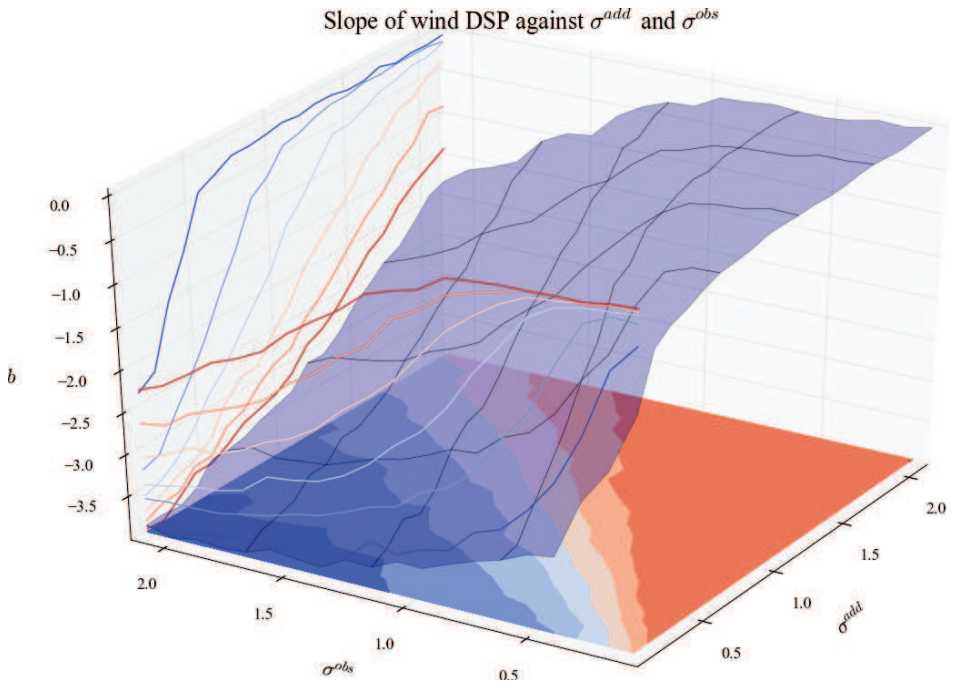


Figure C.48 – PSD slope when σ^{add} and σ^{obs} vary.

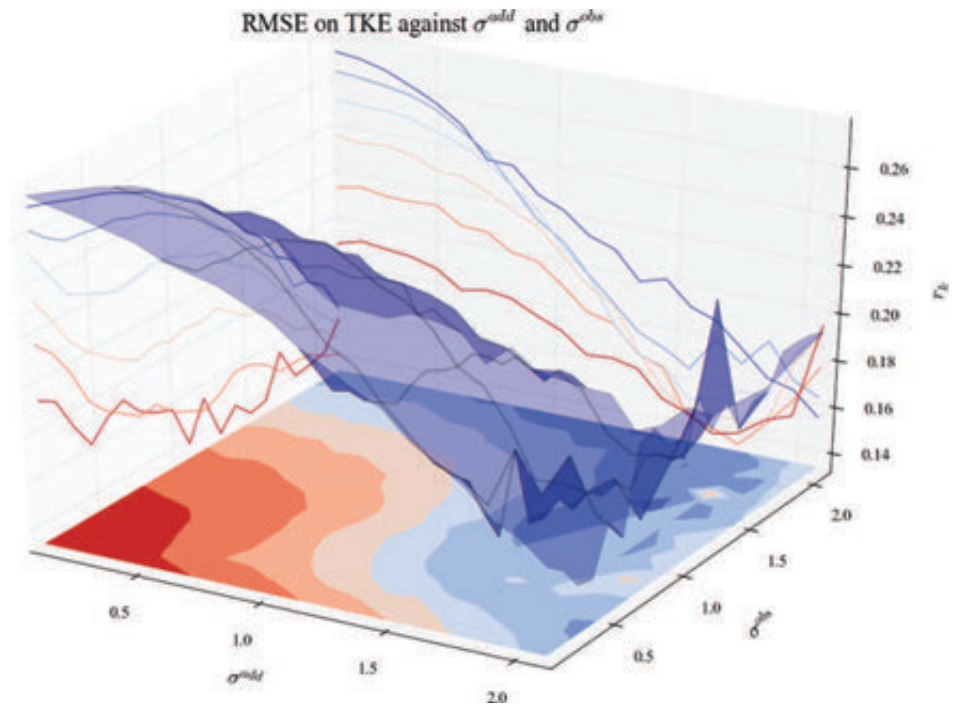


Figure C.49 – RMSE on TKE when σ^{add} and σ^{obs} vary.

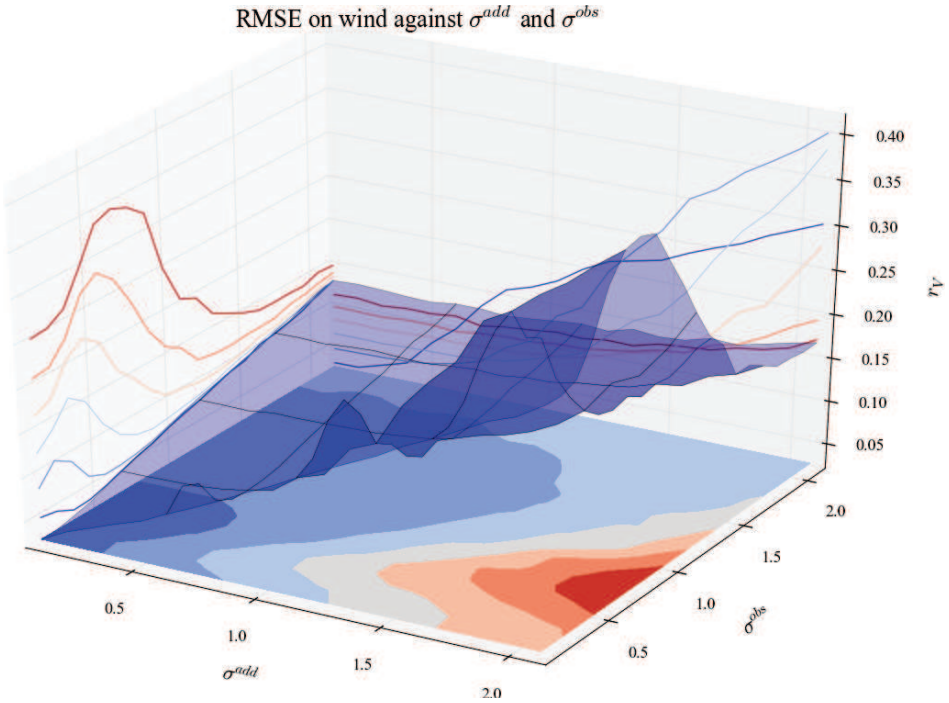


Figure C.50 – RMSE on wind when σ^{add} and σ^{obs} vary.

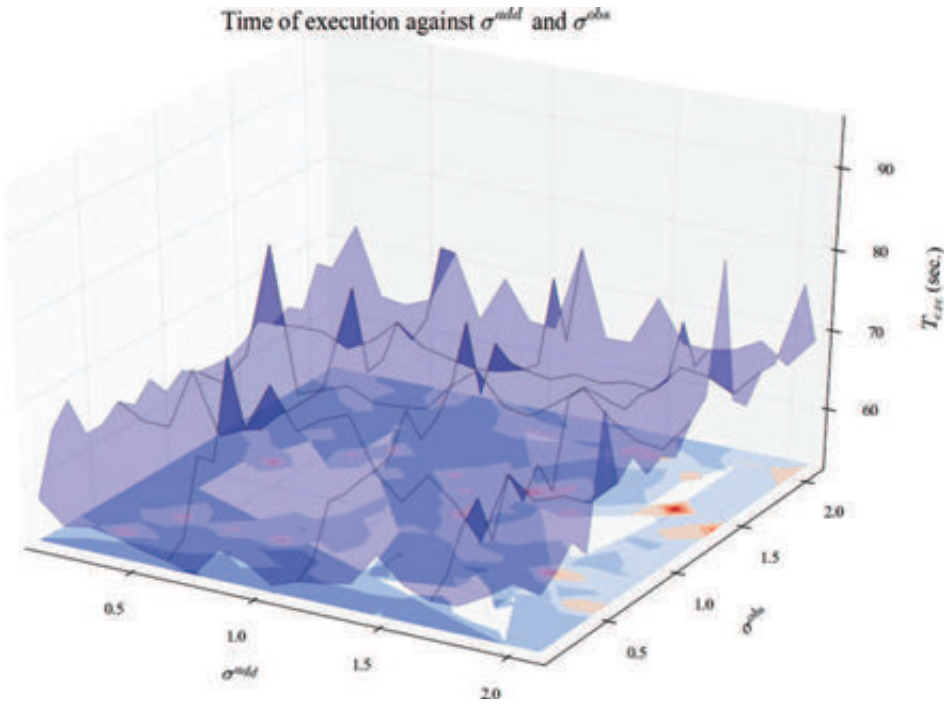


Figure C.51 – Execution time when σ^{add} and σ^{obs} vary.

C.2.11 σ^{add} and τ

In this experiment, only σ^{add} and τ vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

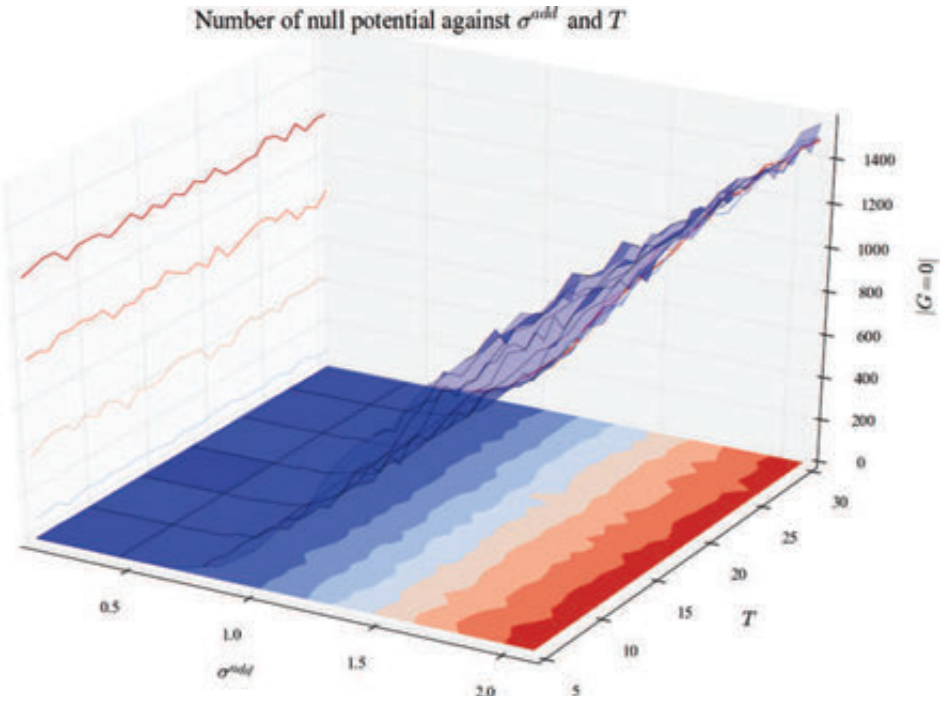


Figure C.52 – Number of null potential when σ^{add} and τ vary.

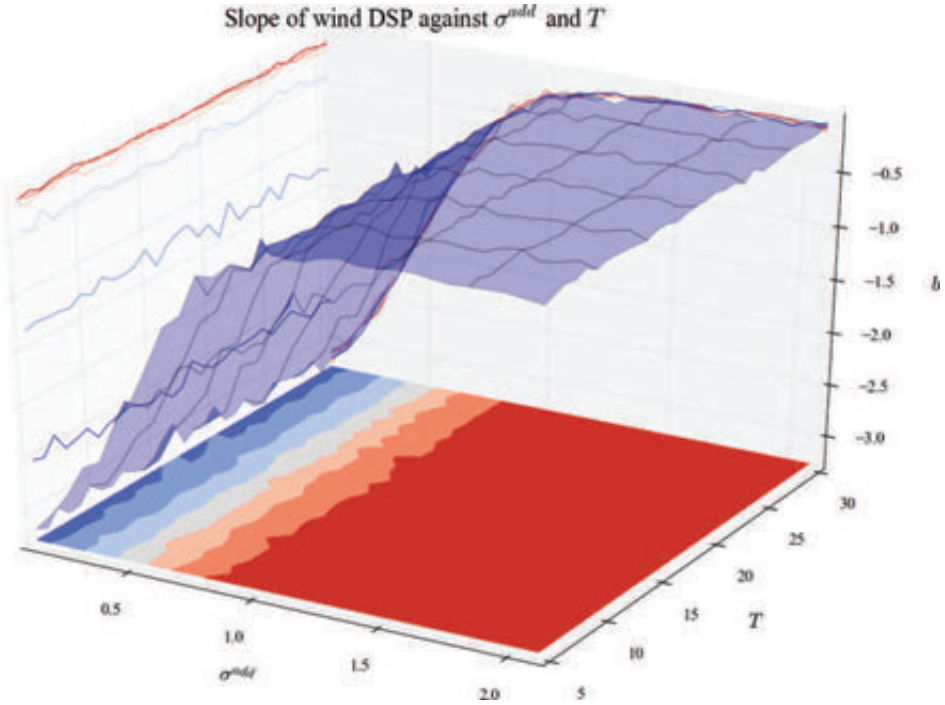


Figure C.53 – PSD slope when σ^{add} and τ vary.

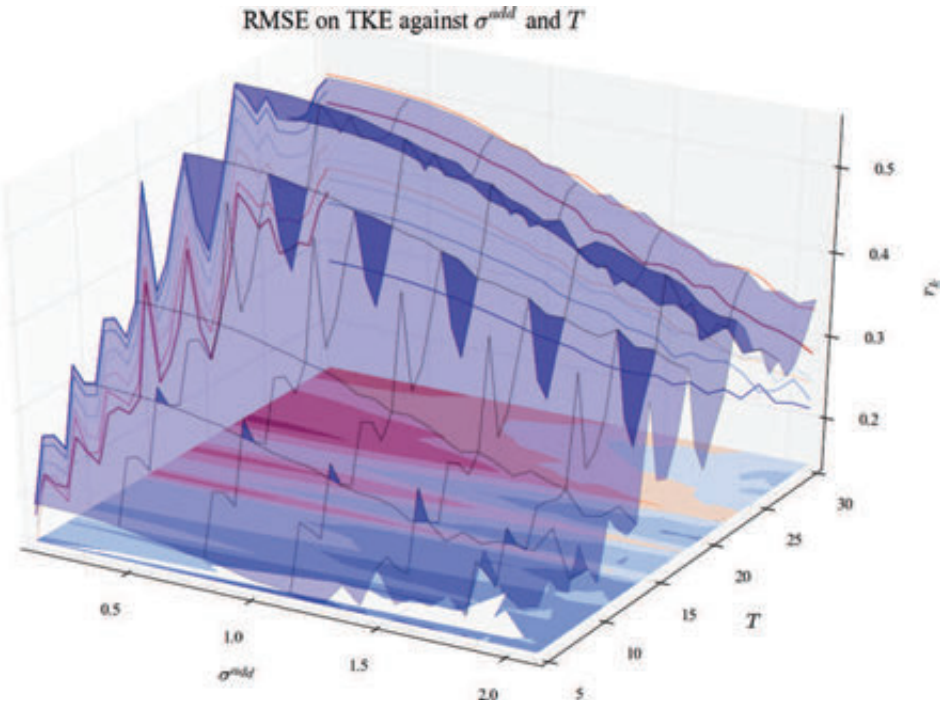


Figure C.54 – RMSE on TKE when σ^{add} and τ vary.

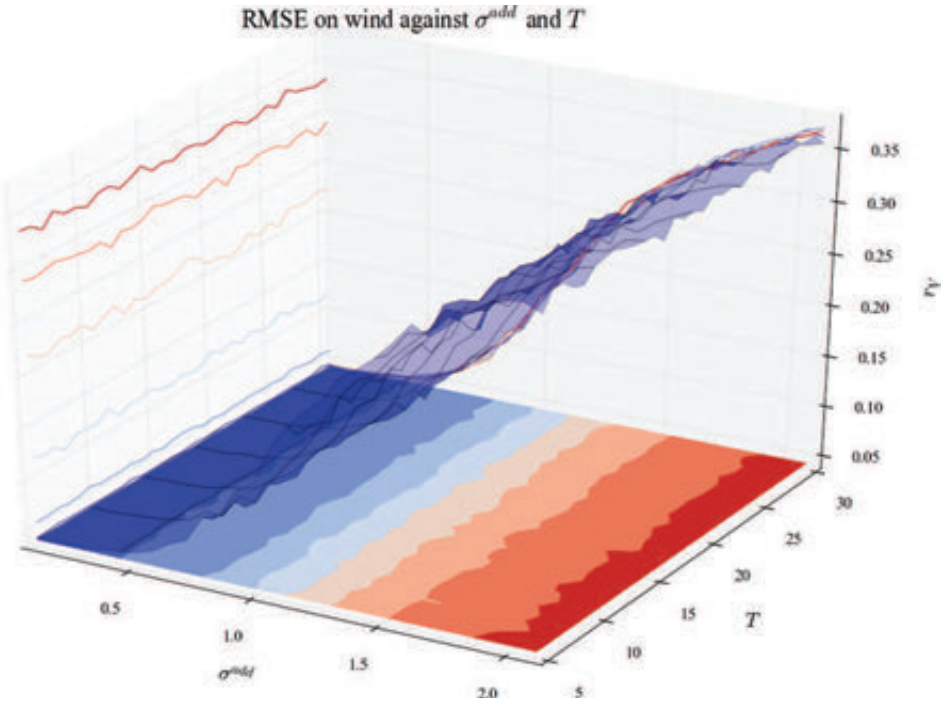


Figure C.55 – RMSE on wind when σ^{add} and τ vary.

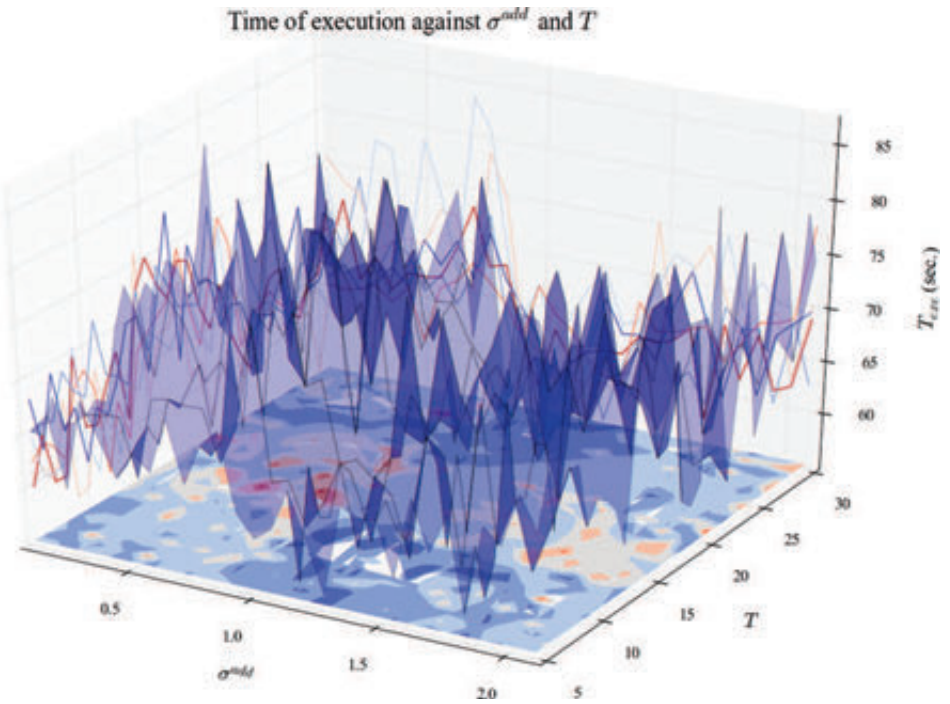


Figure C.56 – Execution time when σ^{add} and τ vary.

C.2.12 σ^{obs} and σ^X

In this experiment, only σ^{obs} and σ^X vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

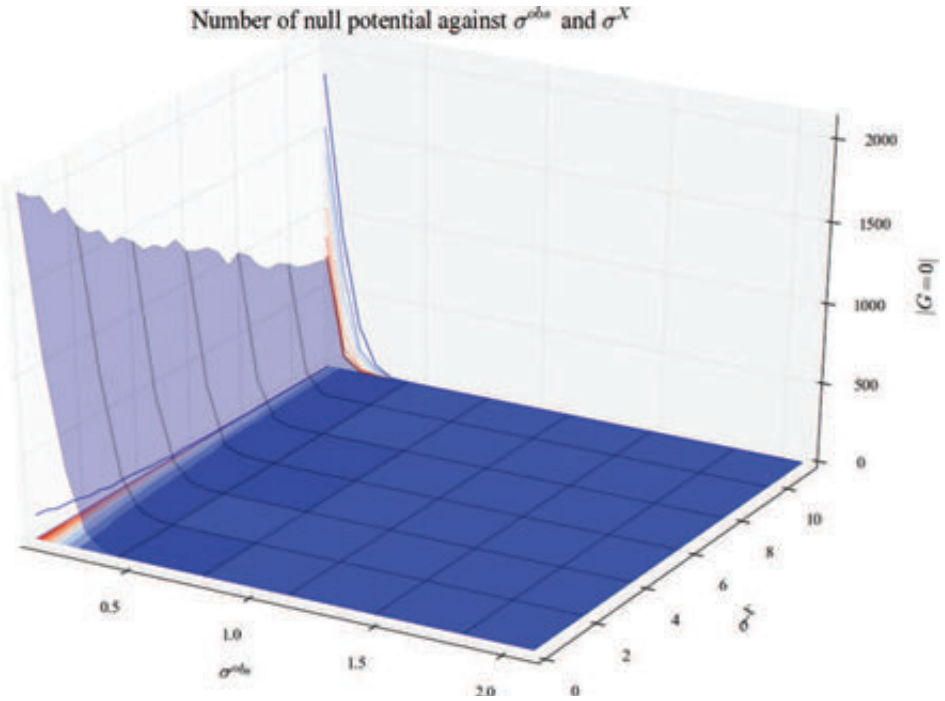


Figure C.57 – Number of null potential when σ^{obs} and σ^X vary.

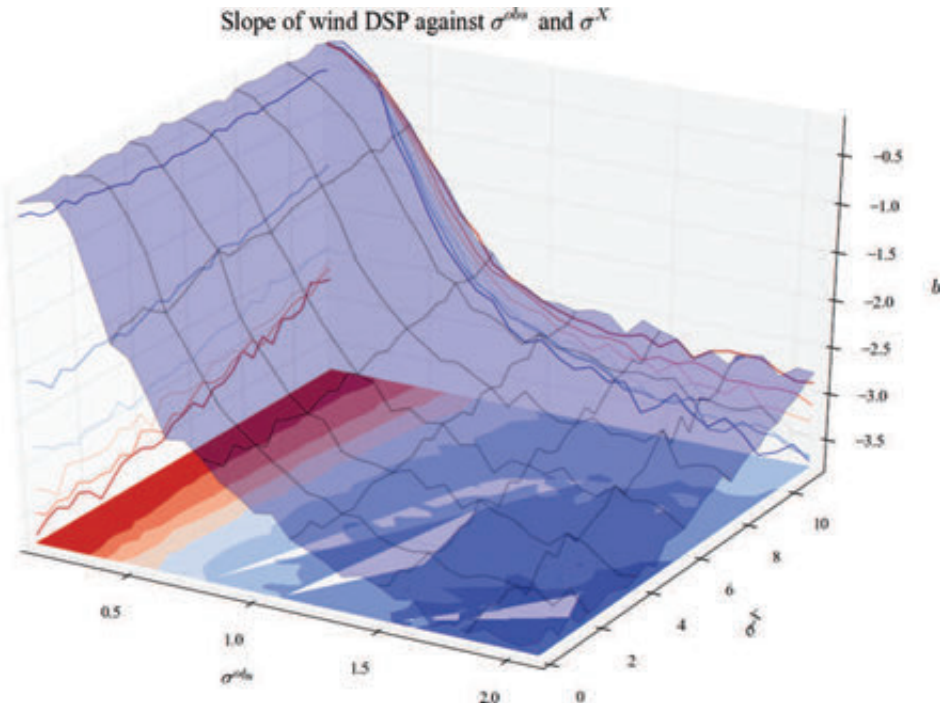


Figure C.58 – PSD slope when σ^{obs} and σ^X vary.

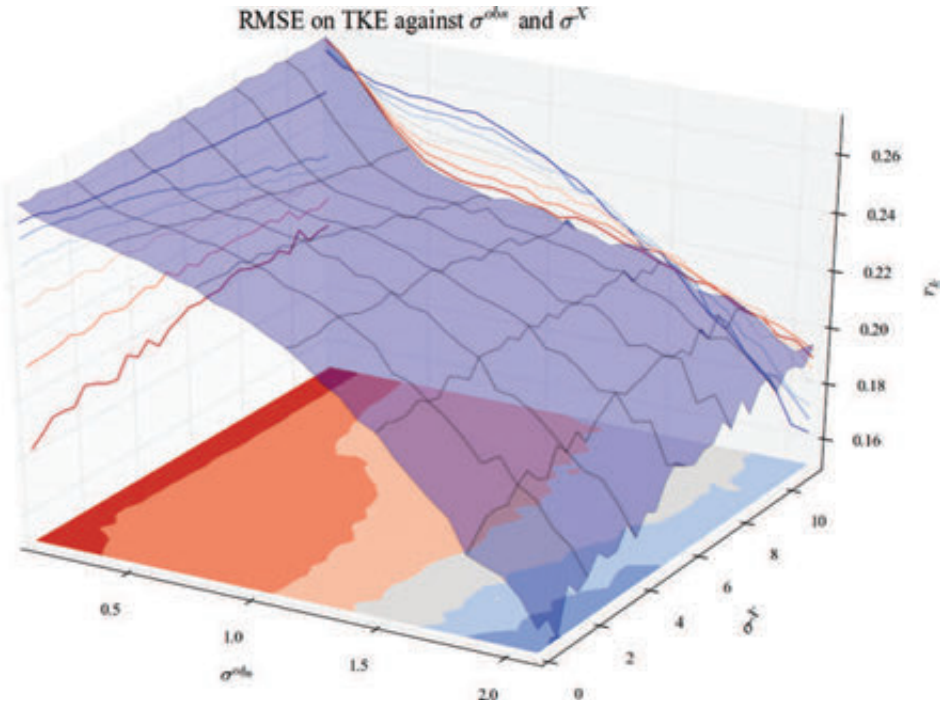


Figure C.59 – RMSE on TKE when σ^{obs} and σ^X vary.

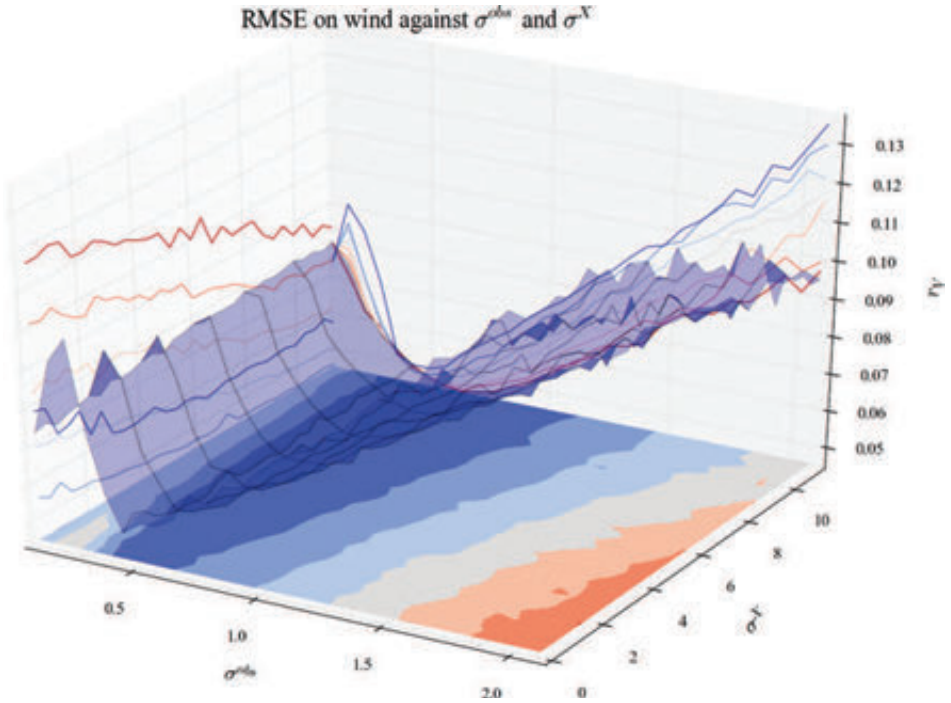


Figure C.60 – RMSE on wind when σ^{obs} and σ^X vary.

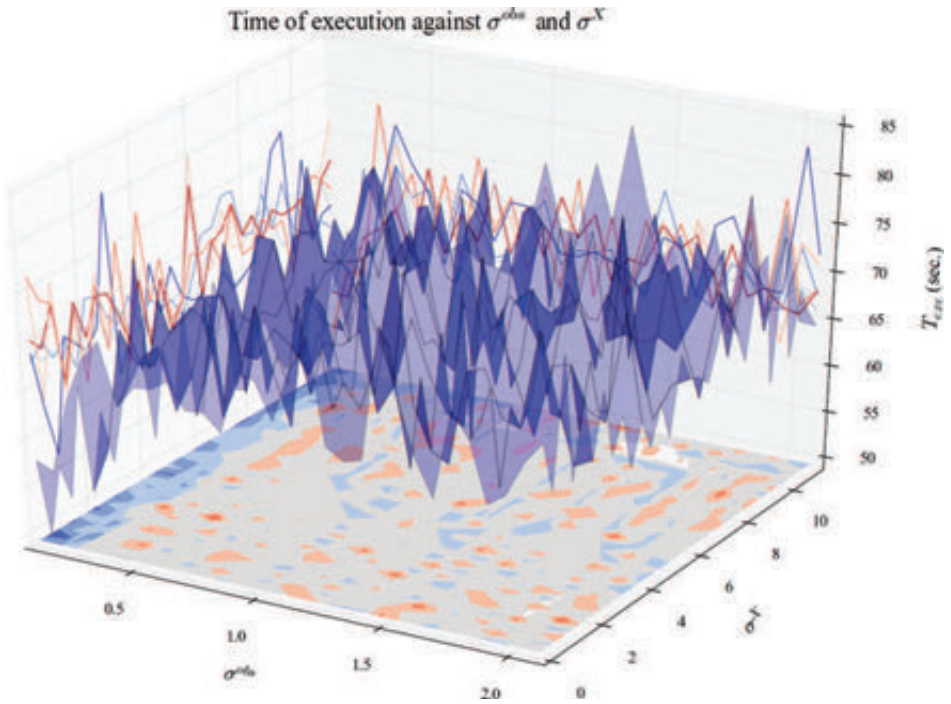


Figure C.61 – Execution time when σ^{obs} and σ^X vary.

C.2.13 σ^{obs} and τ

In this experiment, only σ^{obs} and τ vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

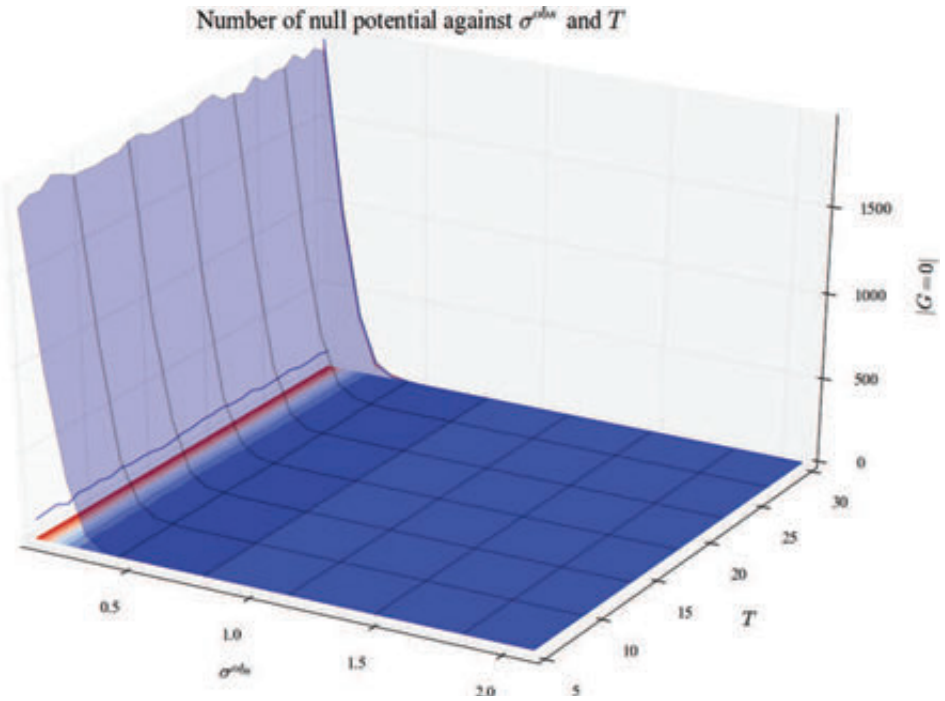


Figure C.62 – Number of null potential when σ^{obs} and τ vary.

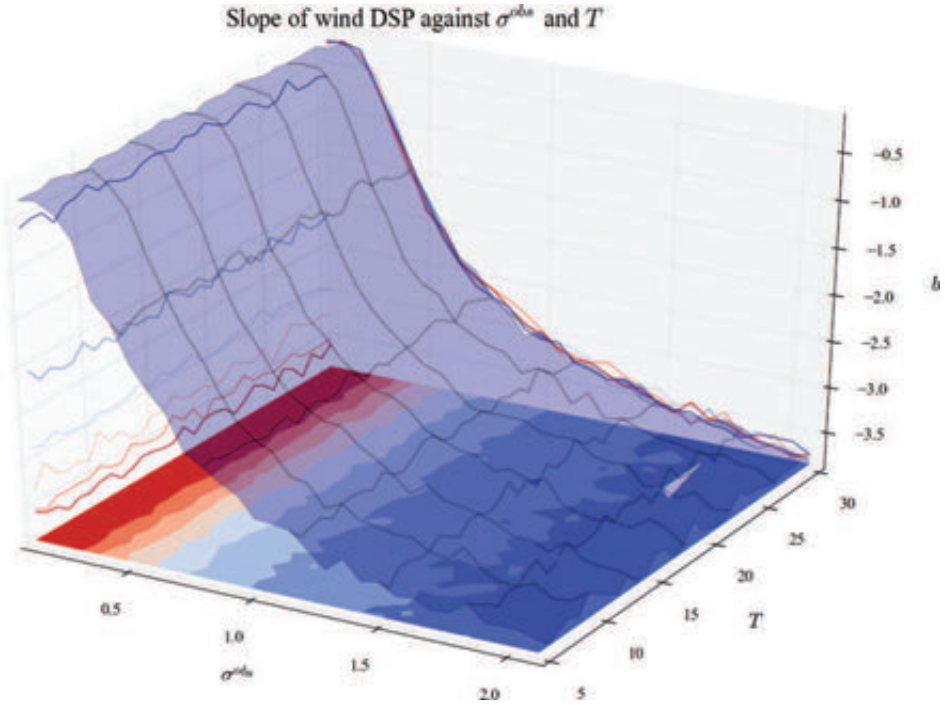


Figure C.63 – PSD slope when σ^{obs} and τ vary.

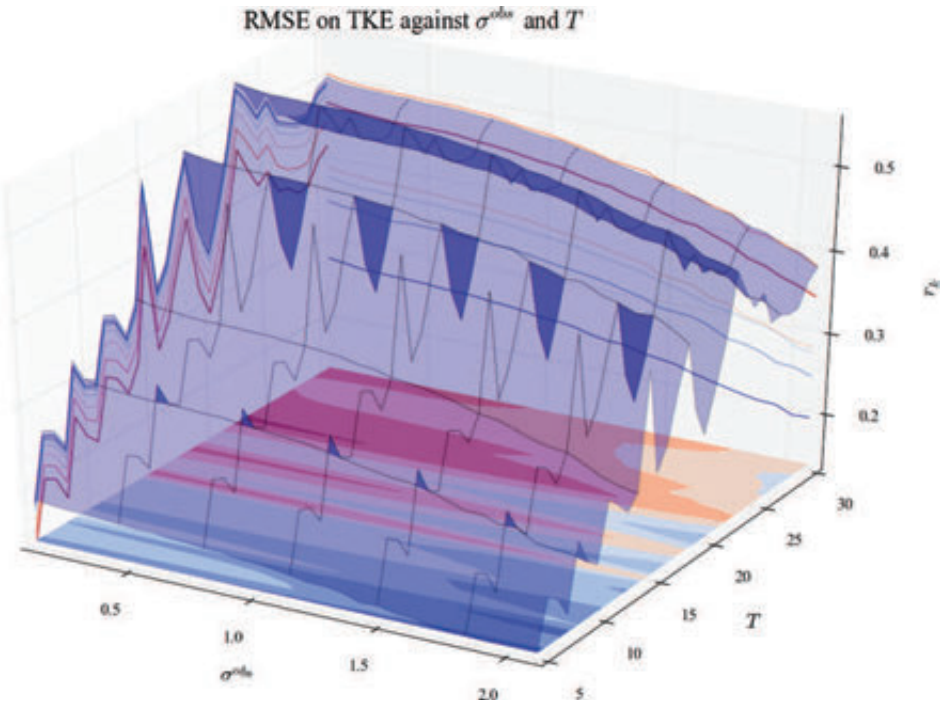


Figure C.64 – RMSE on TKE when σ^{obs} and τ vary.

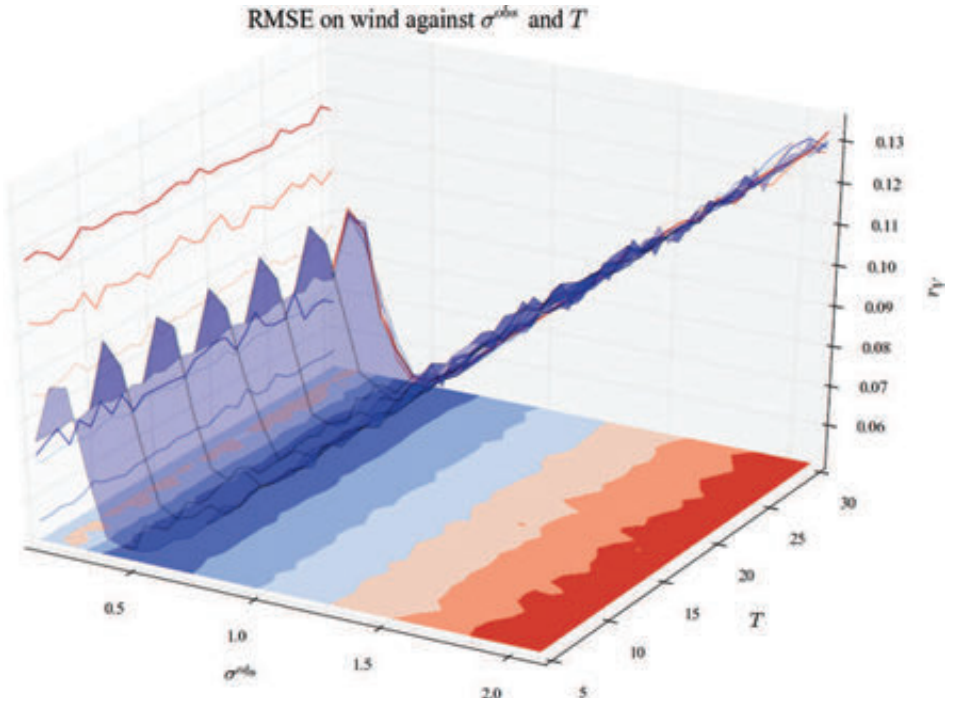


Figure C.65 – RMSE on wind when σ^{obs} and τ vary.

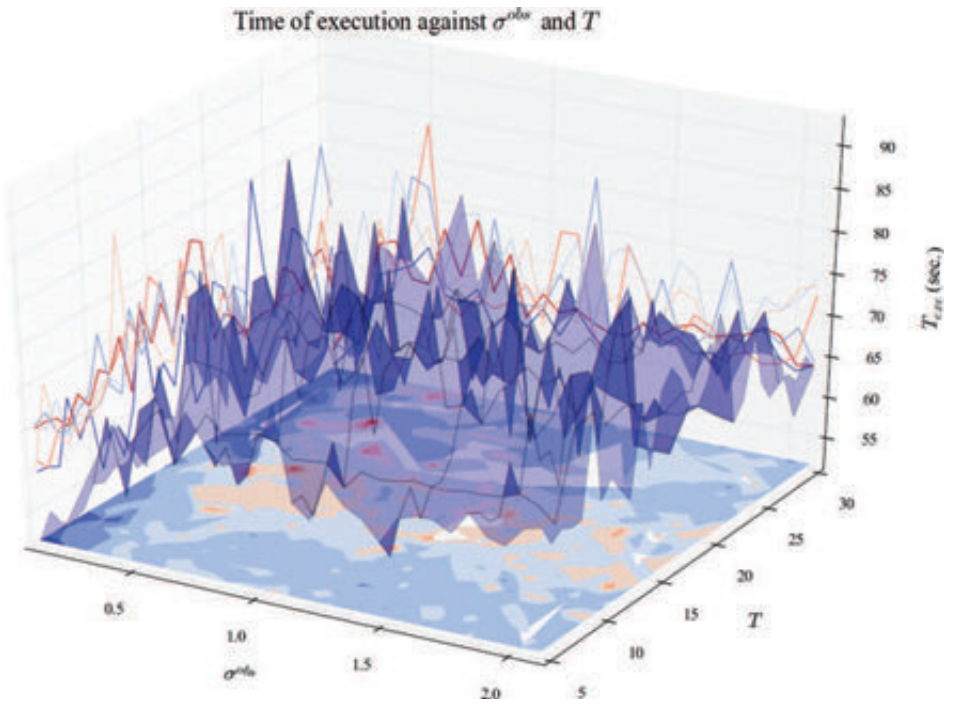


Figure C.66 – Execution time when σ^{obs} and τ vary.

C.2.14 σ^V and σ^X

In this experiment, only σ^V and σ^X vary. To check another experiment, go to table C.3 (hyperlinks in the table). To have more explanation about these inputs, go to table C.1.

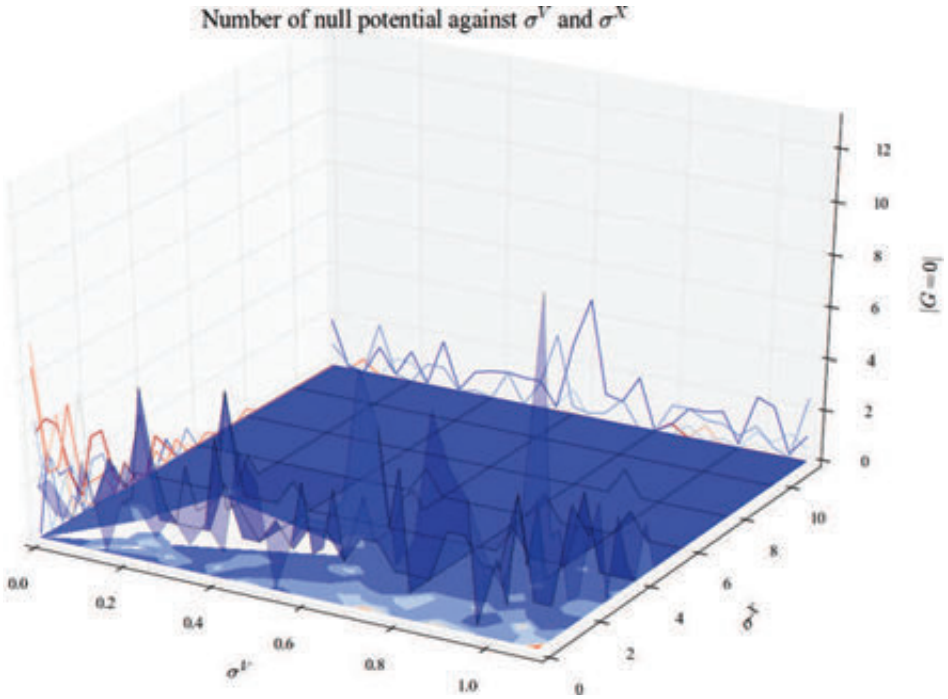


Figure C.67 – Number of null potential when σ^V and σ^X vary.

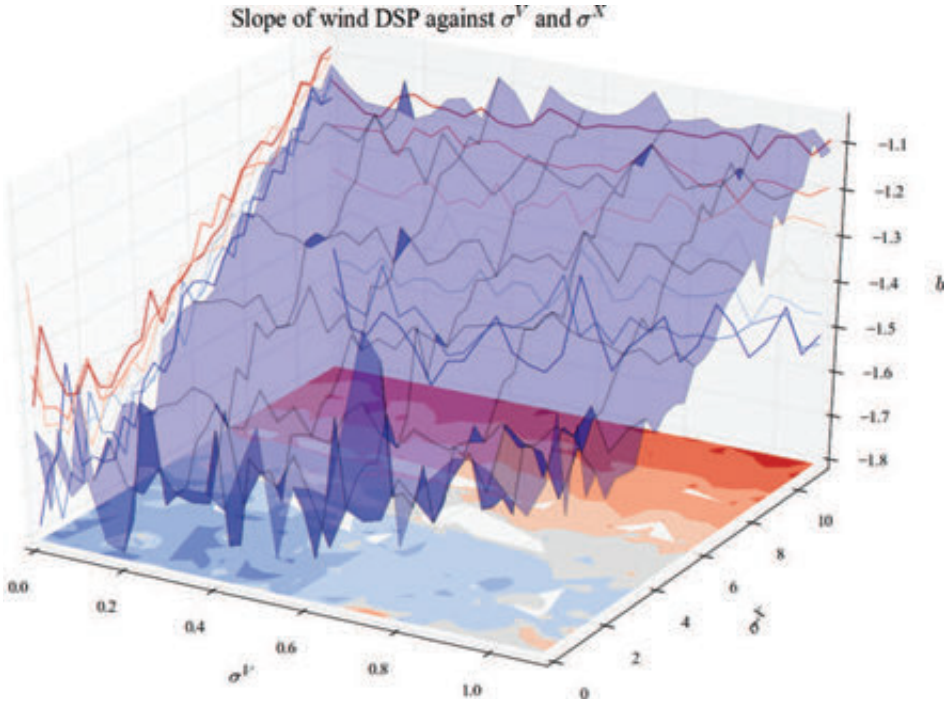


Figure C.68 – PSD slope when σ^V and σ^X vary.

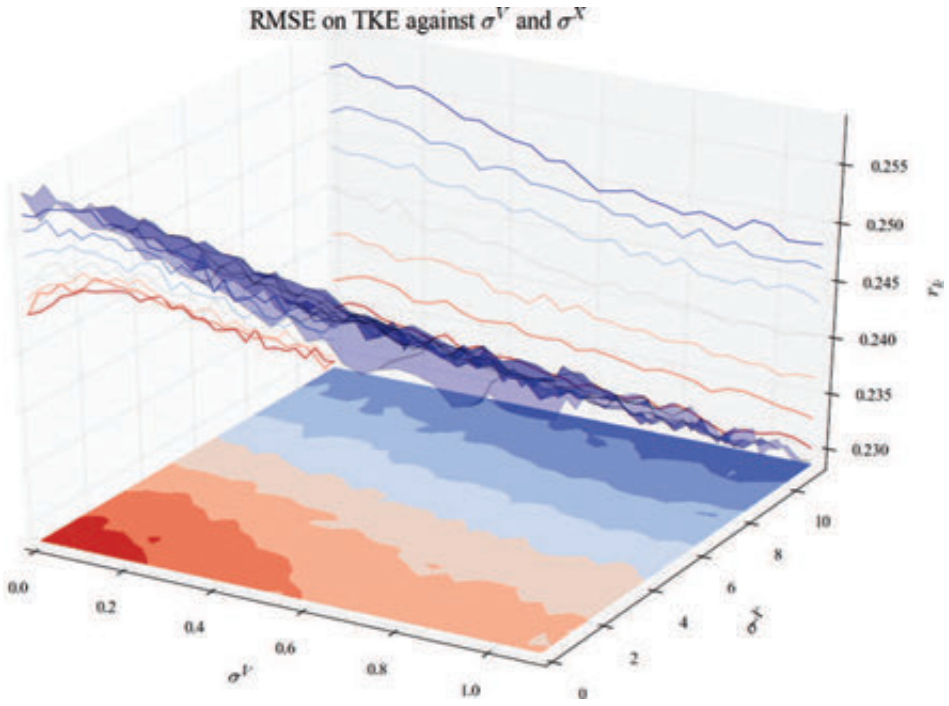


Figure C.69 – RMSE on TKE when σ^V and σ^X vary.

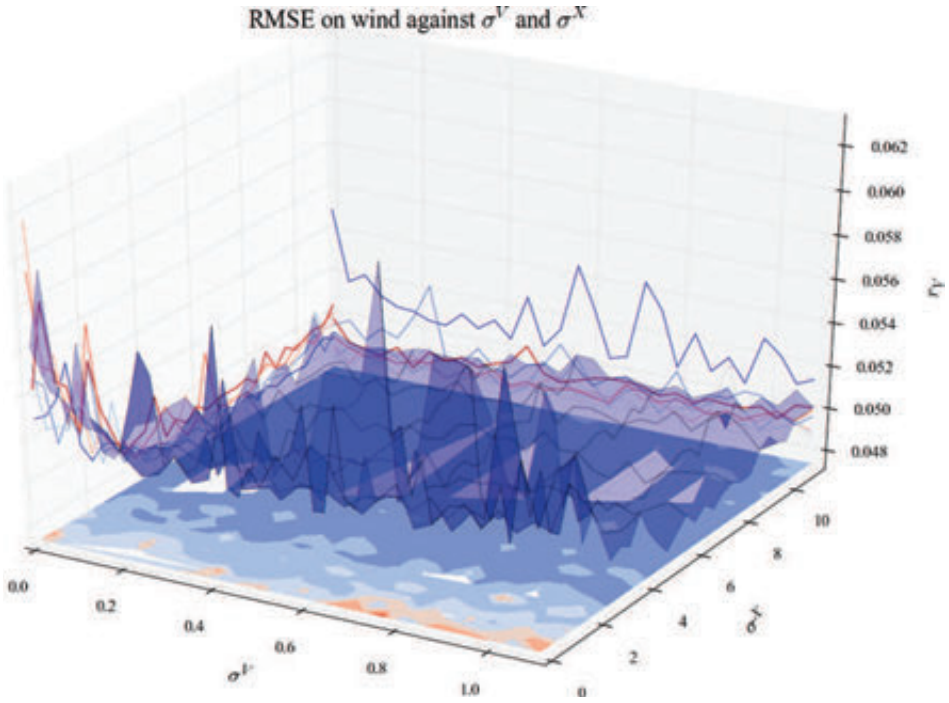


Figure C.70 – RMSE on wind when σ^V and σ^X vary.

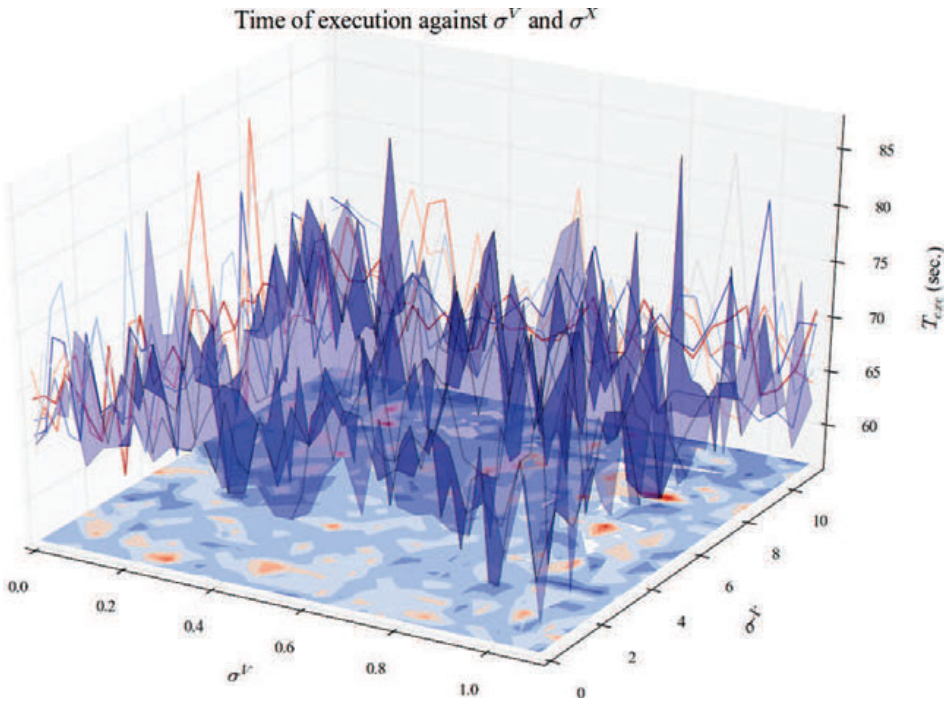


Figure C.71 – Execution time when σ^V and σ^X vary.

C.3 Results by output

C.3.1 Number of null potential

In this subsection are shown only the influence on the number of null potential N_{G0}. First are shown the Sobol indices for this output the most informative experiment and then the rest of experiments. To browse by experiment, use the hyperlinks in table C.3 (all outputs are computed for each experiment). To check another output, go back to table C.2. There is also a list of figures page 384.

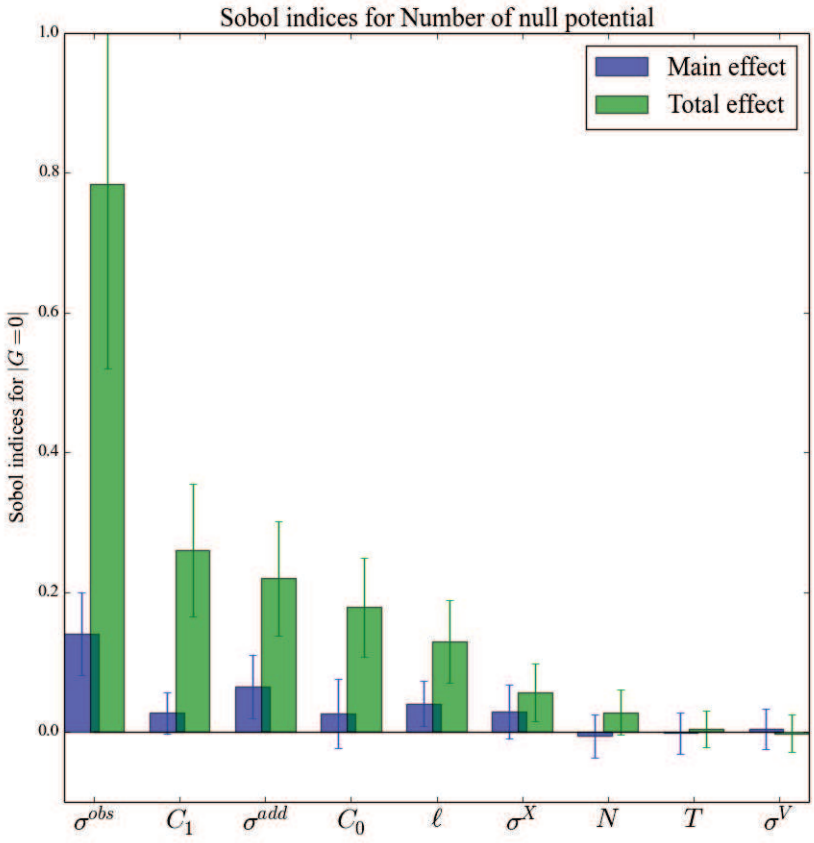


Figure C.72 – Sobol indices (score of influence) for number of null potential. Main effect in blue (effect of input alone), total effect in green (effect of input with all its interactions).

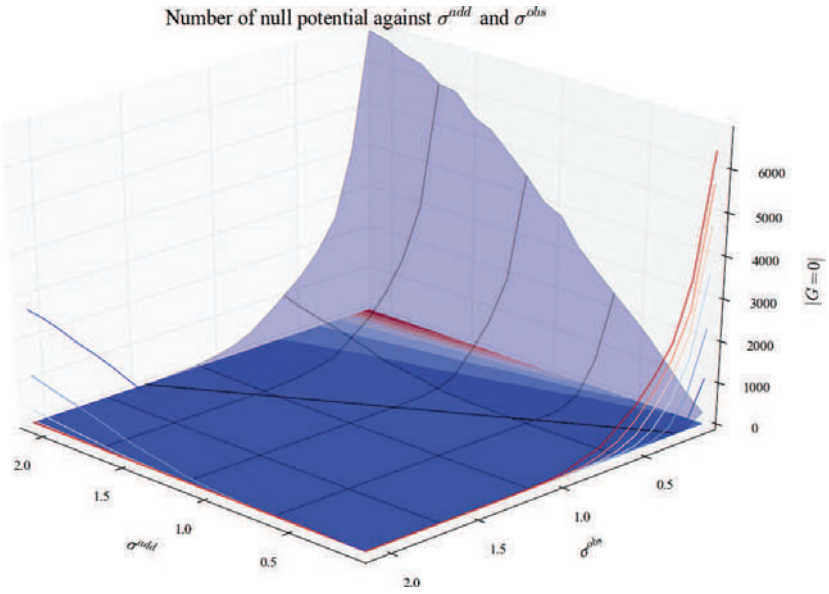


Figure C.73 – Number of null potential when σ^{add} and σ^{obs} vary.

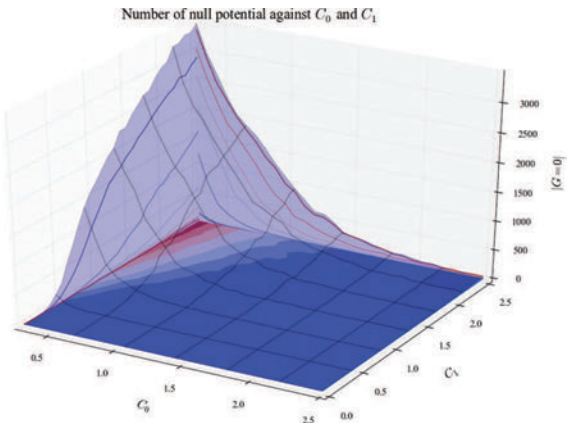


Figure C.74 – Number of null potential when C_0 and C_1 vary.

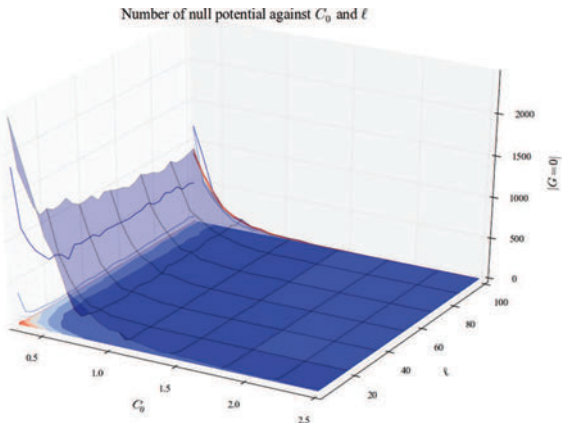


Figure C.75 – Number of null potential when C_0 and ℓ vary.

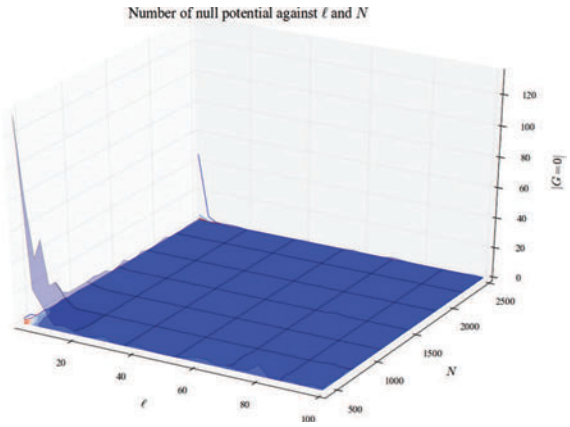


Figure C.76 – Number of null potential when ℓ and N vary.

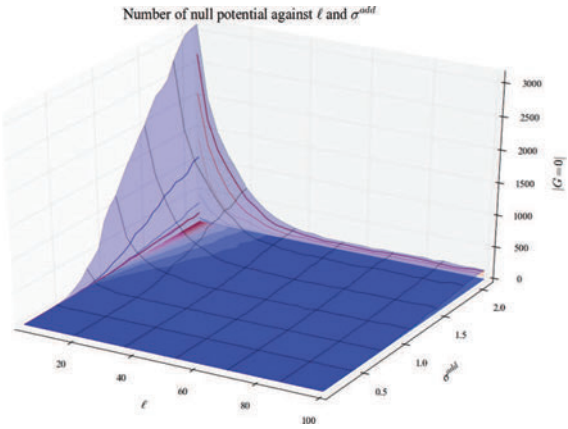


Figure C.77 – Number of null potential when ℓ and σ^{add} vary.

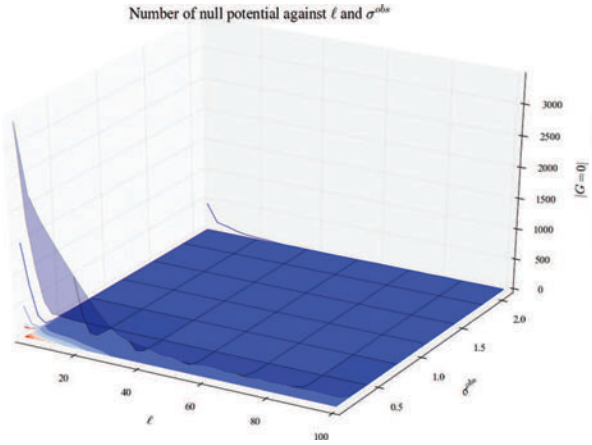


Figure C.78 – Number of null potential when ℓ and σ^{obs} vary.

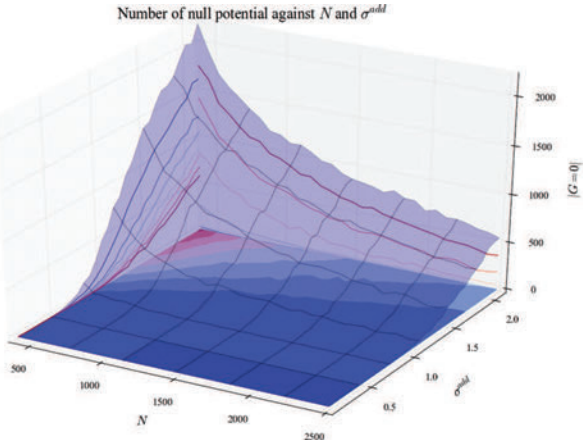


Figure C.79 – Number of null potential when N and σ^{add} vary.

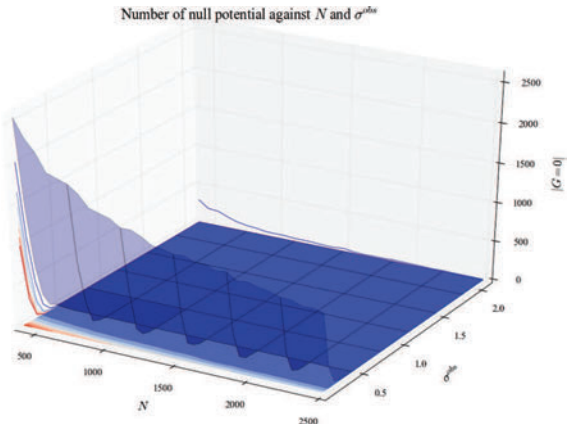


Figure C.80 – Number of null potential when N and σ^{obs} vary.

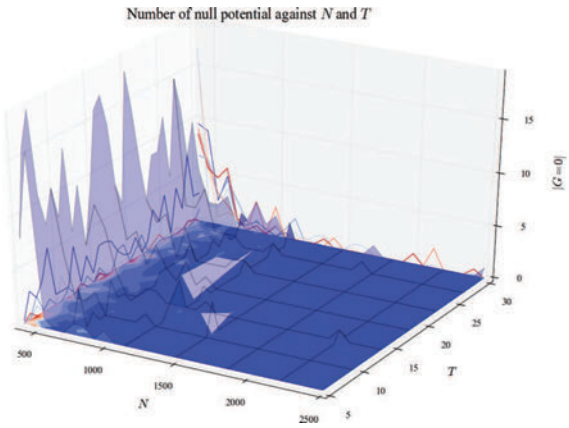


Figure C.81 – Number of null potential when N and τ vary.

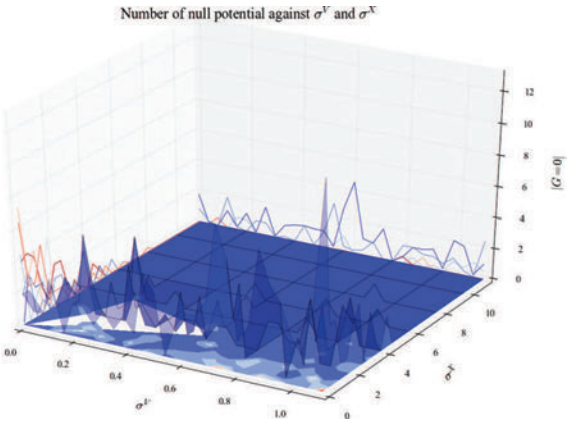


Figure C.82 – Number of null potential when σ^V and σ^X vary.

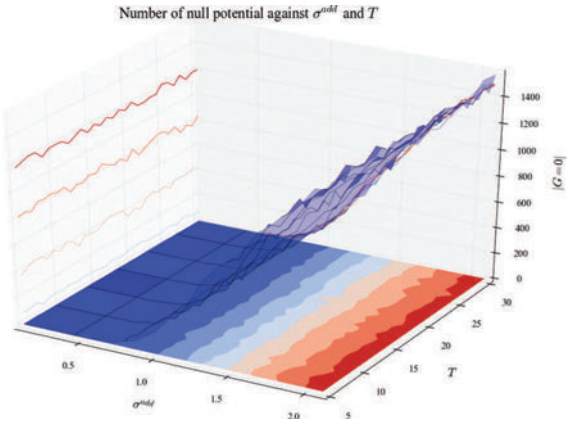


Figure C.83 – Number of null potential when σ^{add} and τ vary.

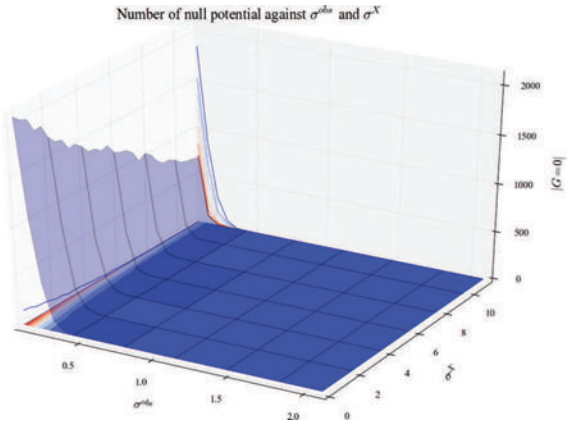


Figure C.84 – Number of null potential when σ^{obs} and σ^X vary.

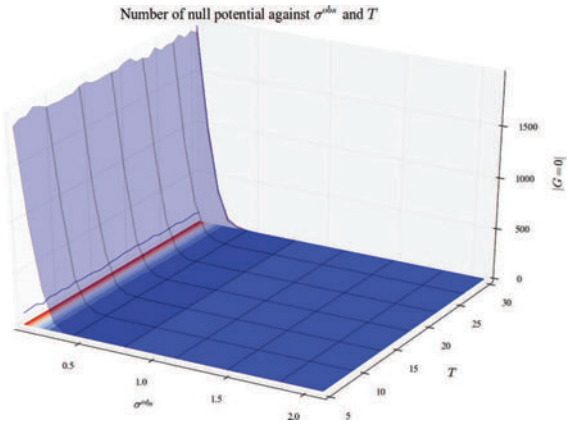


Figure C.85 – Number of null potential when σ^{obs} and τ vary.

C.3.2 Slope of PSD of wind

In this subsection are shown only the influence on the slope of the estimated wind power density spectrum b. First are shown the Sobol indices for this output the most informative experiment and then the rest of experiments. To browse by experiment, use the hyperlinks in table C.3 (all outputs are computed for each experiment). To check another output, go back to table C.2. There is also a list of figures page 384.

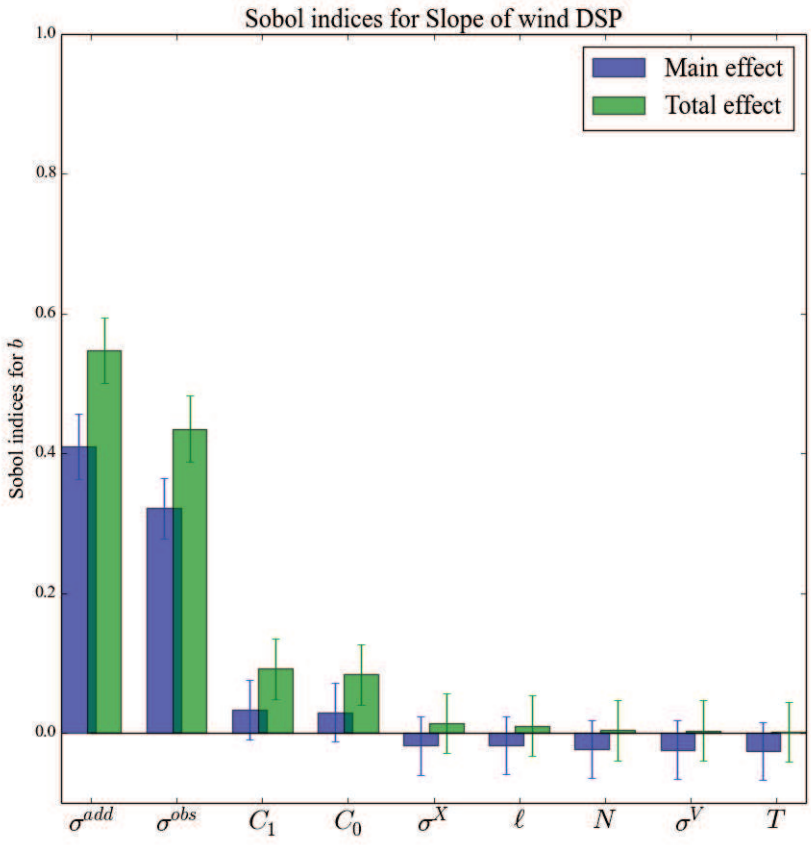


Figure C.86 – Sobol indices (score of influence) for PSD slope. Main effect in blue (effect of input alone), total effect in green (effect of input with all its interactions).

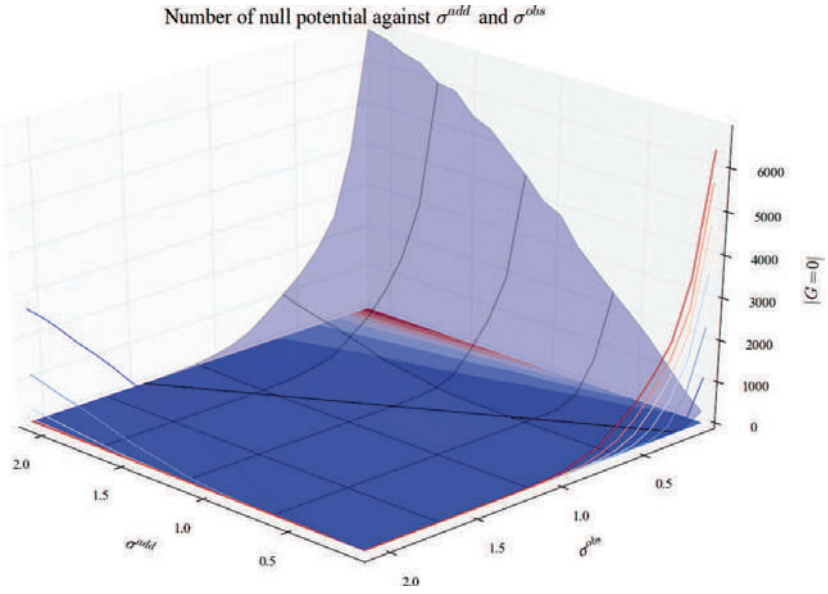


Figure C.87 – Slope of PSD when σ^{add} and σ^{obs} vary.

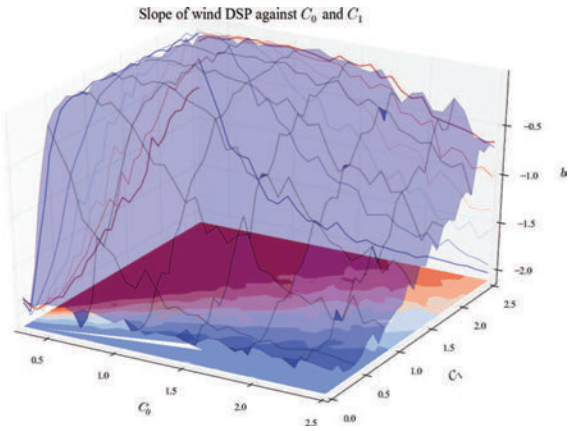


Figure C.88 – Slope of PSD when C_0 and C_1 vary.

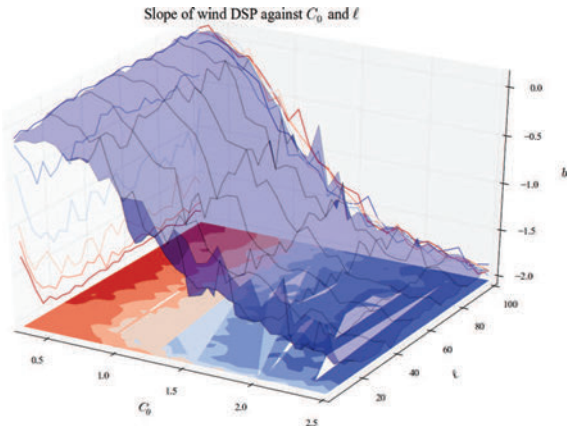


Figure C.89 – Slope of PSD when C_0 and ℓ vary.

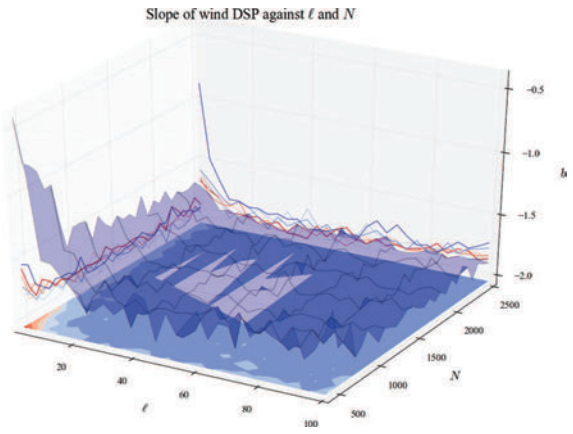


Figure C.90 – Slope of PSD when ℓ and N vary.

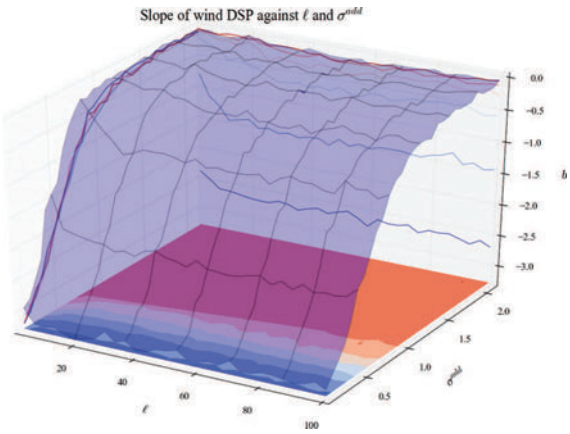


Figure C.91 – Slope of PSD when ℓ and σ^{add} vary.

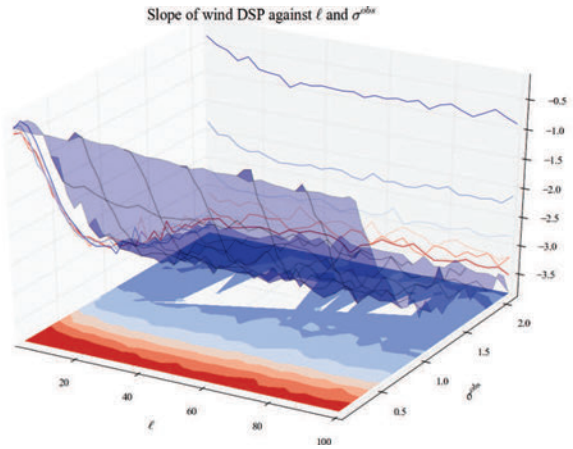


Figure C.92 – Slope of PSD when ℓ and σ^{obs} vary.

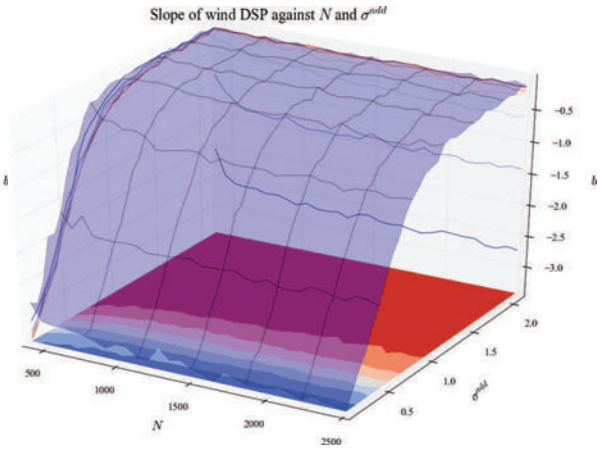


Figure C.93 – Slope of PSD when N and σ^{add} vary.

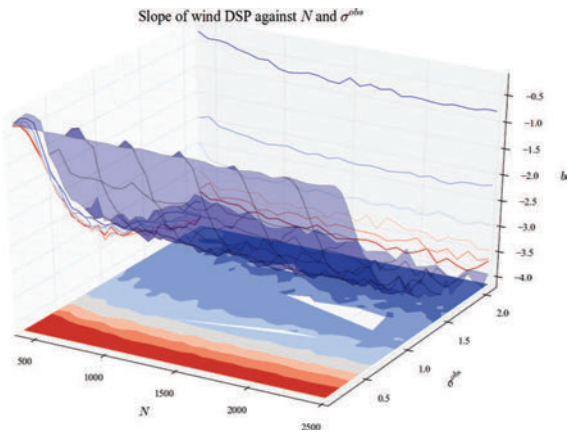


Figure C.94 – Slope of PSD when N and σ^{obs} vary.

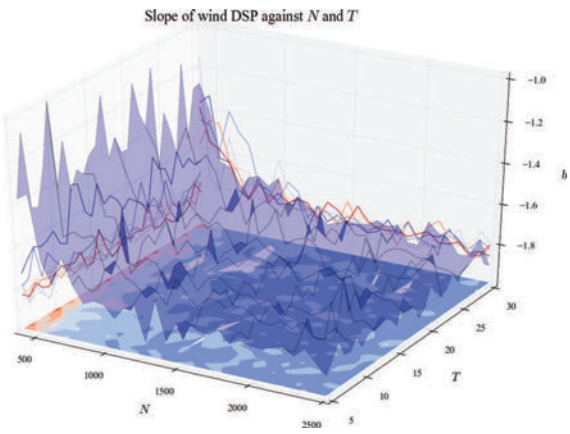


Figure C.95 – Slope of PSD when N and τ vary.

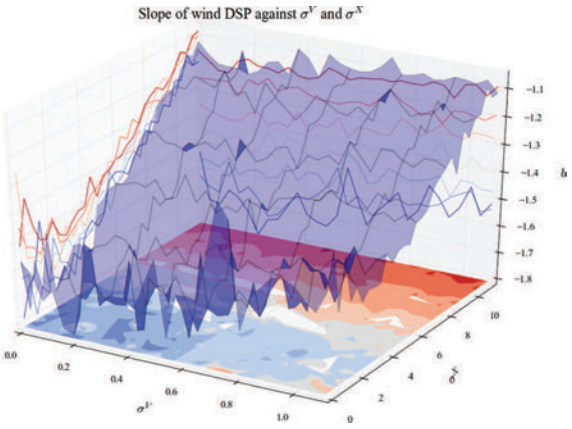


Figure C.96 – Slope of PSD when σ^V and σ^X vary.

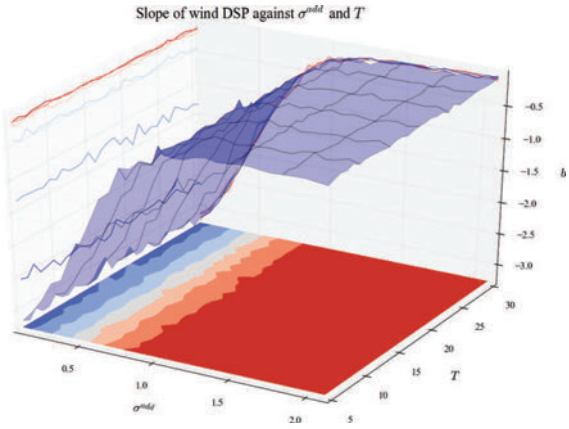


Figure C.97 – Slope of PSD when σ^{add} and τ vary.

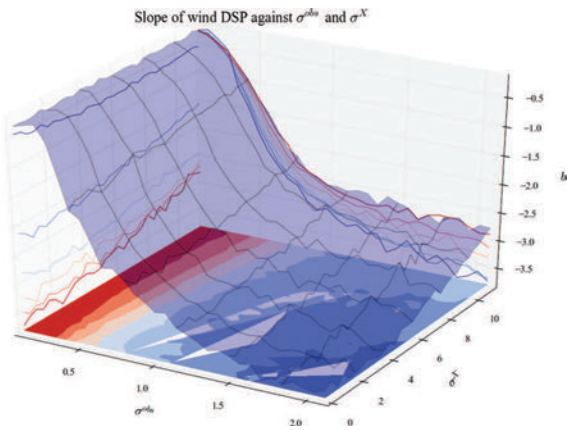


Figure C.98 – Slope of PSD when σ^{obs} and σ^X vary.

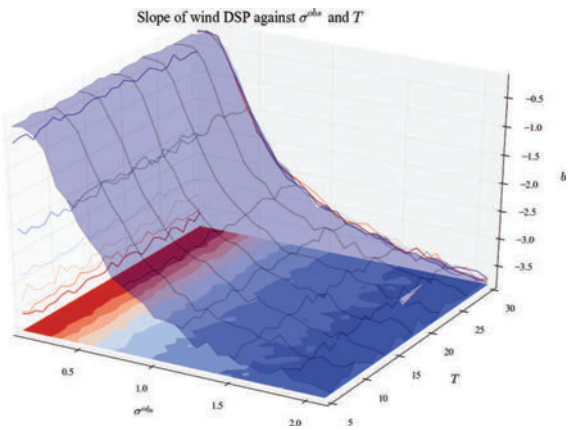


Figure C.99 – Slope of PSD when σ^{obs} and τ vary.

C.3.3 RMSE on TKE

In this subsection are shown only the influence on the root-mean-squared error on turbulent kinetic energy r_k. First are shown the Sobol indices for this output the most informative experiment and then the rest of experiments. To browse by experiment, use the hyperlinks in table C.3 (all outputs are computed for each experiment). To check another output, go back to table C.2. There is also a list of figures page 384.

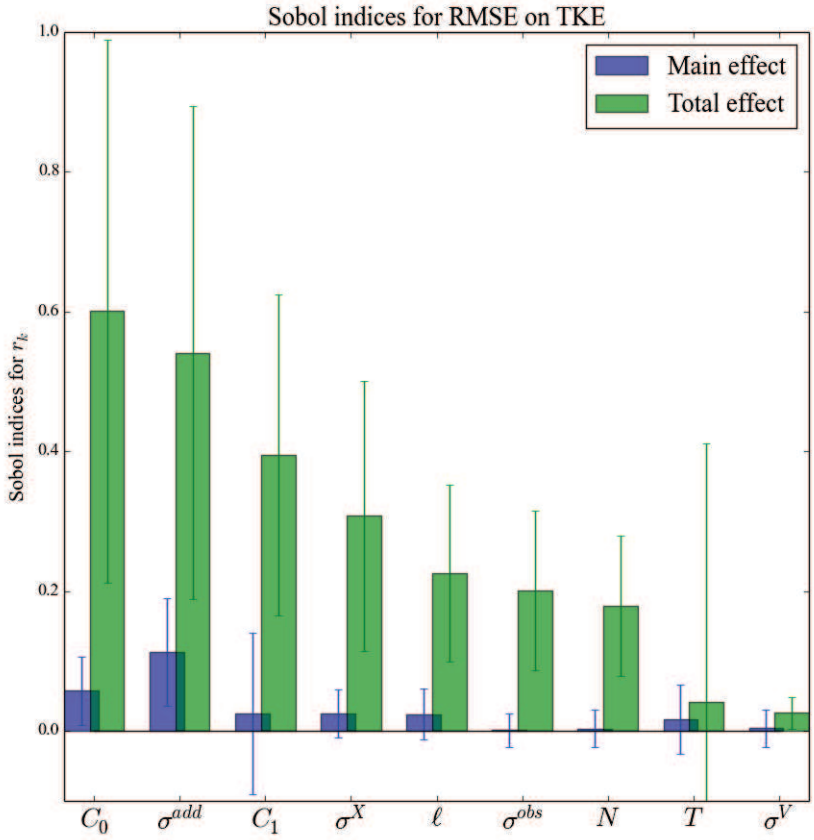


Figure C.100 – Sobol indices (score of influence) for TKE RMSE. Main effect in blue (effect of input alone), total effect in green (effect of input with all its interactions).

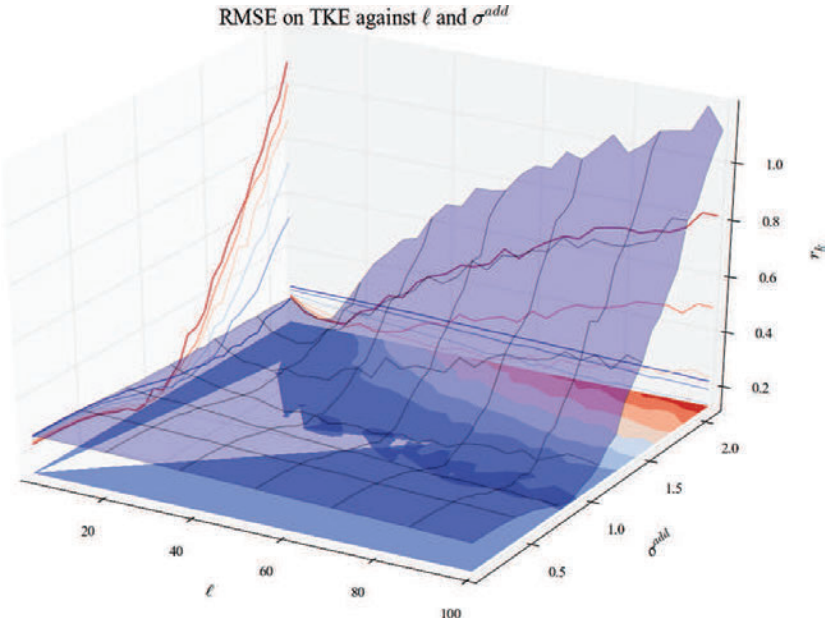


Figure C.101 – TKE RMSE when ℓ and σ^{add} vary.

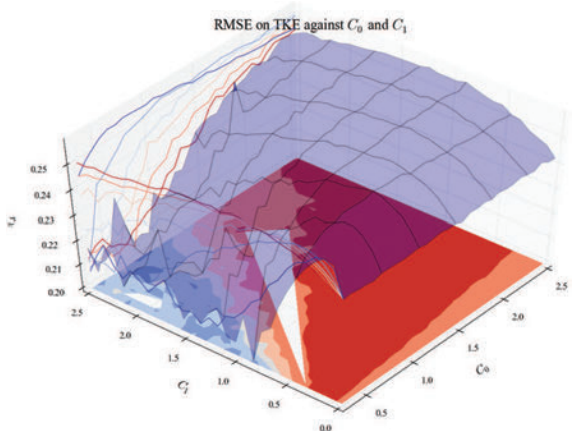


Figure C.102 – TKE RMSE when C_0 and C_1 vary.

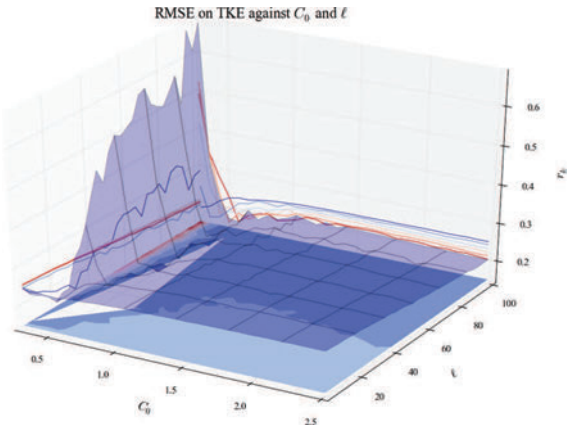


Figure C.103 – TKE RMSE when C_0 and ℓ vary.

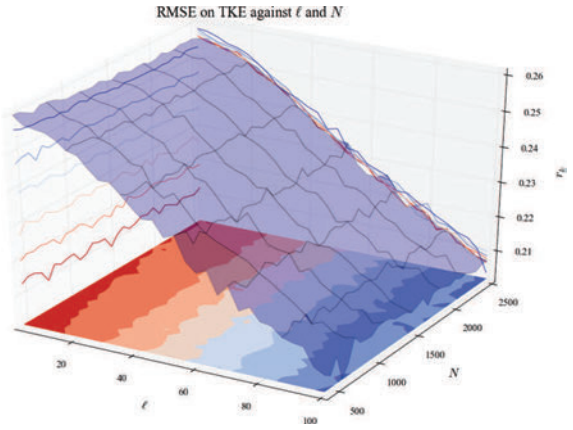


Figure C.104 – TKE RMSE when ℓ and N vary.

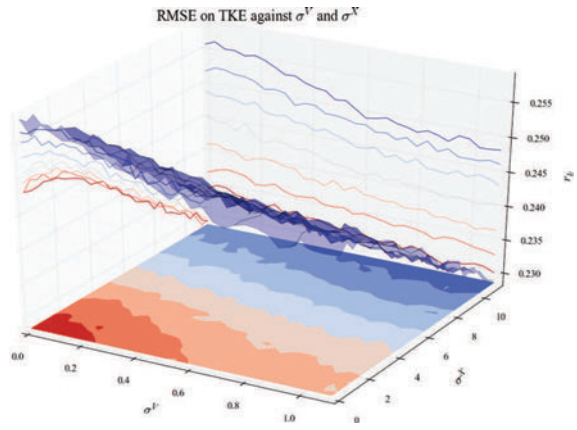


Figure C.105 – TKE RMSE when σ^V and σ^X vary.

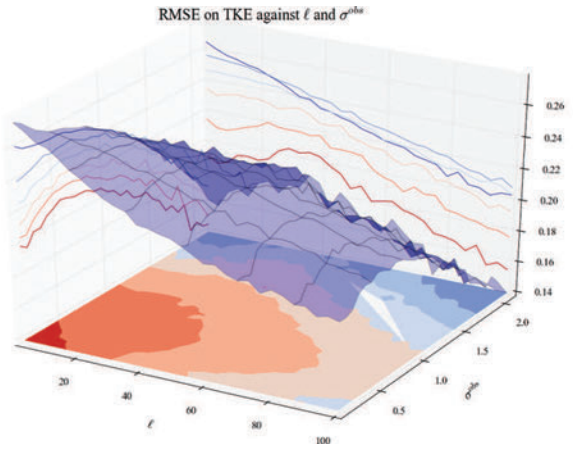


Figure C.106 – TKE RMSE when ℓ and σ^{obs} vary.

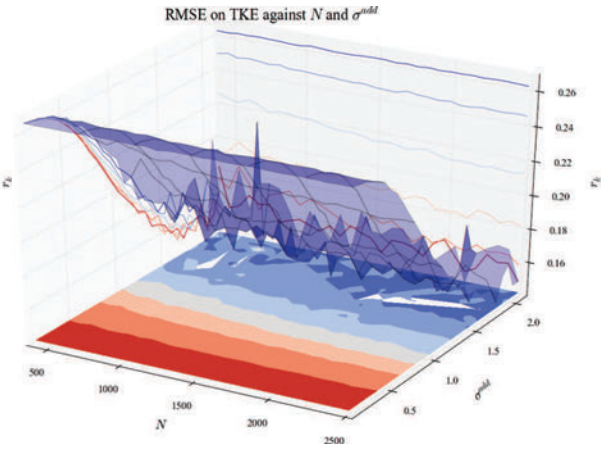


Figure C.107 – TKE RMSE when N and σ^{add} vary.

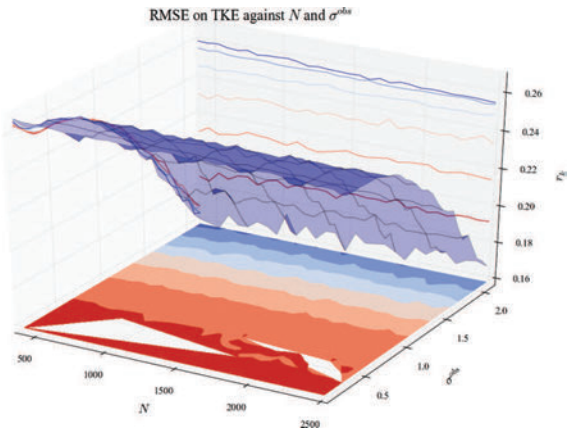


Figure C.108 – TKE RMSE when N and σ^{obs} vary.

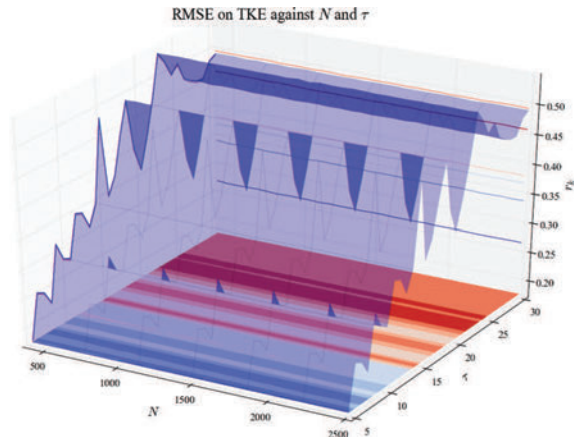


Figure C.109 – TKE RMSE when N and τ vary.

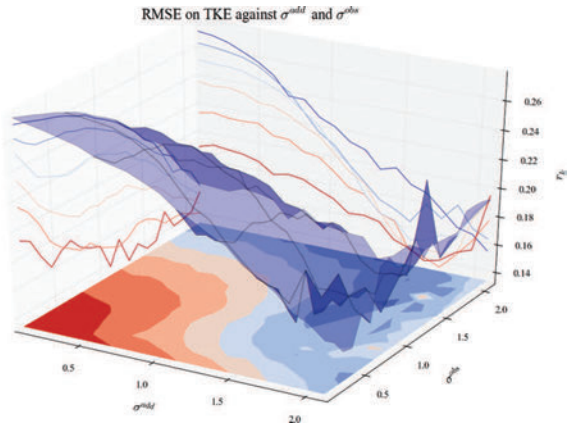


Figure C.110 – TKE RMSE when σ^{add} and σ^{obs} vary.

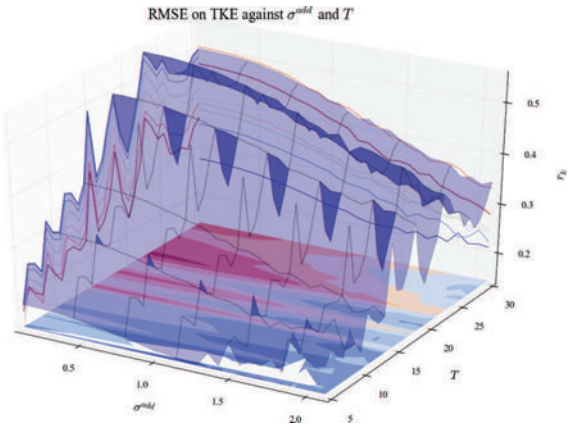


Figure C.111 – TKE RMSE when σ^{add} and τ vary.

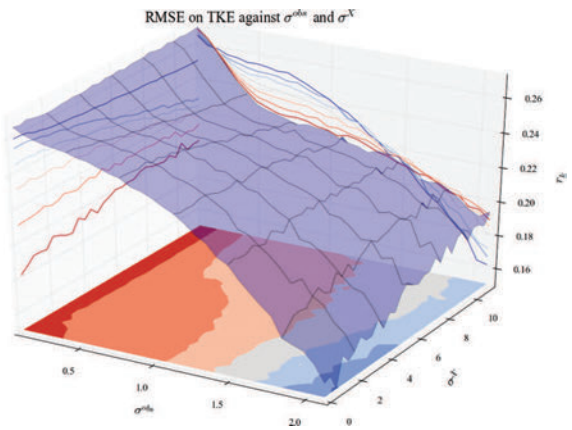


Figure C.112 – TKE RMSE when σ^{obs} and σ^{X} vary.

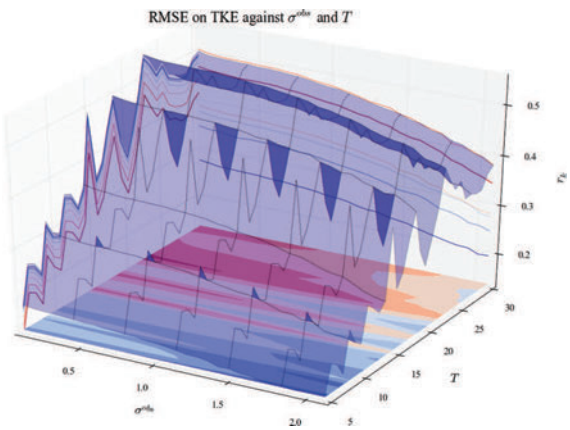


Figure C.113 – TKE RMSE when σ^{obs} and τ vary.

C.3.4 RMSE on wind

In this subsection are shown only the influence on the root-mean-squared error on vertical wind r_V. First are shown the Sobol indices for this output the most informative experiment and then the rest of experiments. To browse by experiment, use the hyperlinks in table C.3 (all outputs are computed for each experiment). To check another output, go back to table C.2. There is also a list of figures page 384.

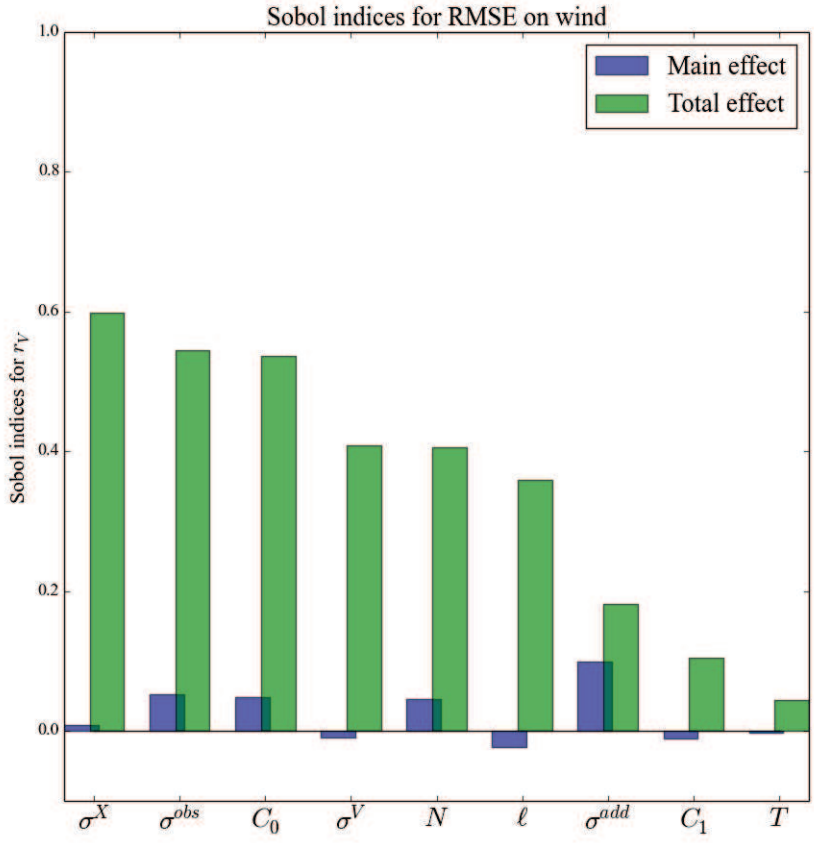


Figure C.114 – Sobol indices (score of influence) for wind RMSE. Main effect in blue (effect of input alone), total effect in green (effect of input with all its interactions).

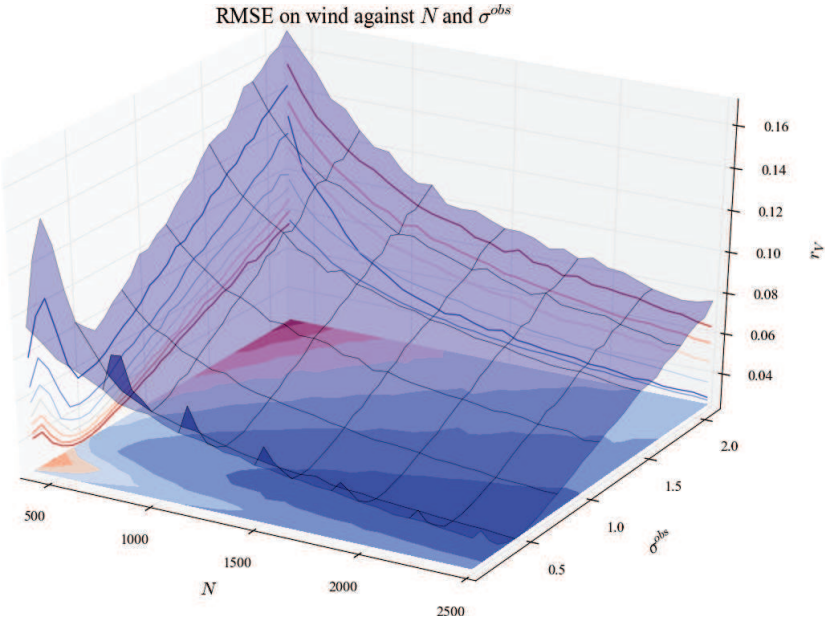


Figure C.115 – Wind RMSE when N and σ^{obs} vary.

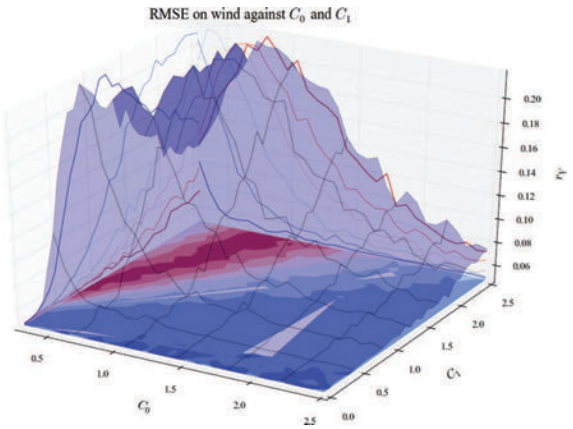


Figure C.116 – Wind RMSE when C_0 and C_1 vary.

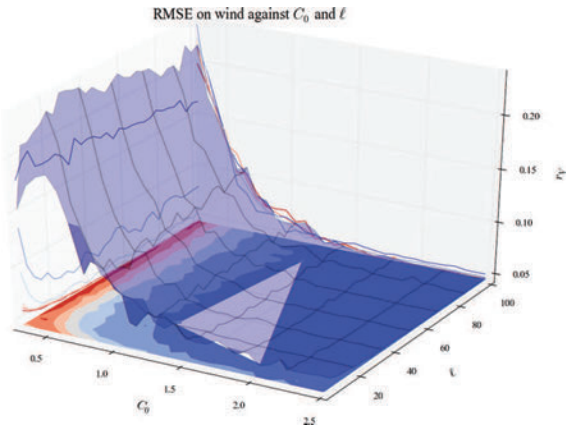


Figure C.117 – Wind RMSE when C_0 and ℓ vary.

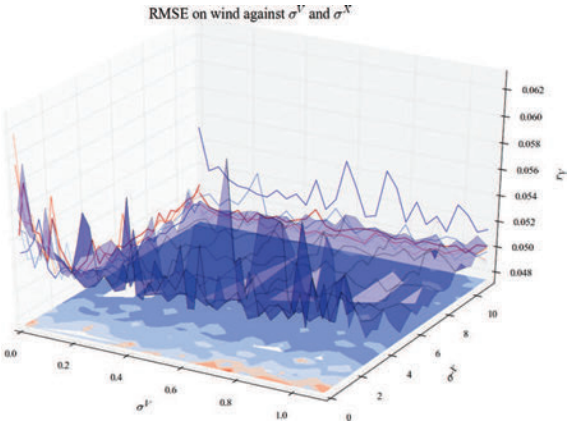


Figure C.118 – Wind RMSE when σ^V and σ^X vary.

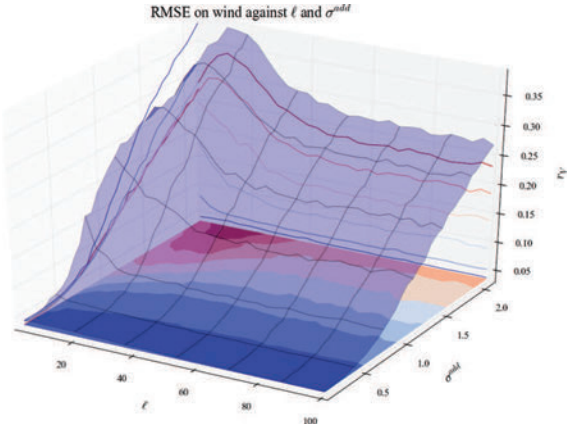


Figure C.119 – Wind RMSE when ℓ and σ^{add} vary.

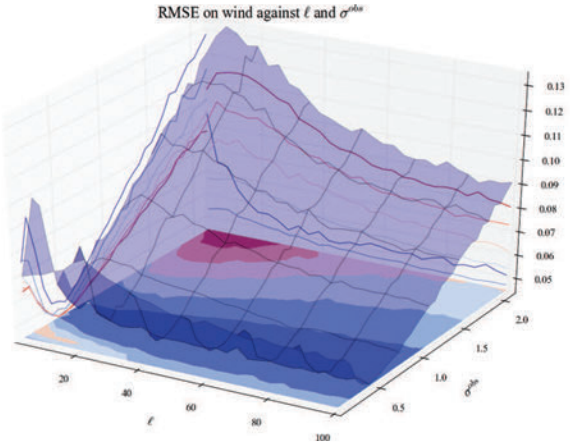


Figure C.120 – Wind RMSE when ℓ and σ^{obs} vary.

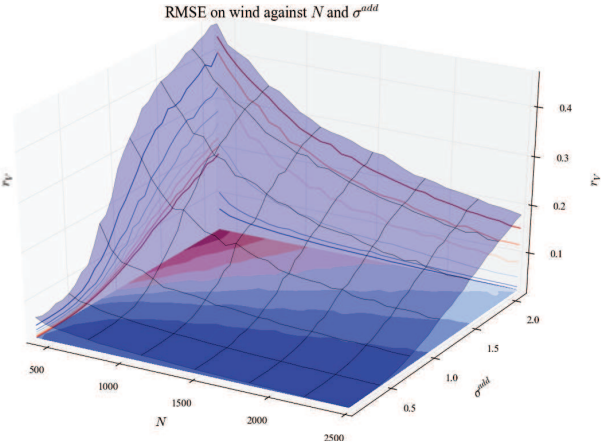


Figure C.121 – Wind RMSE when N and σ^{add} vary.

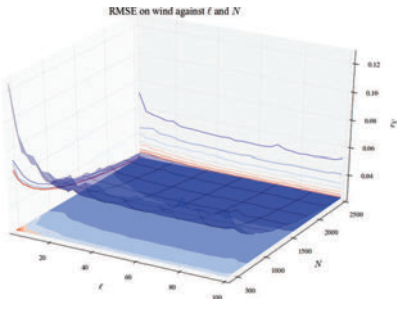


Figure C.122 – Wind RMSE when ℓ and N vary.

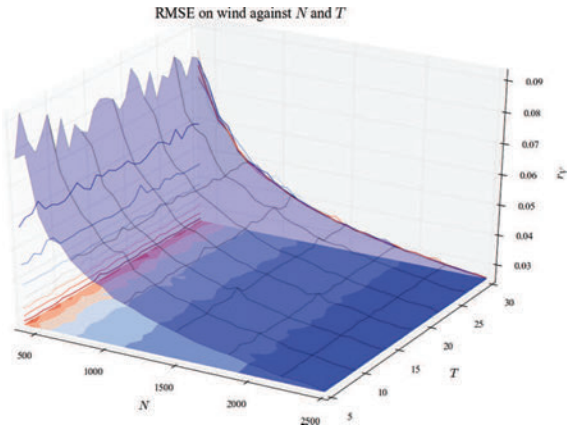


Figure C.123 – Wind RMSE when N and τ vary.

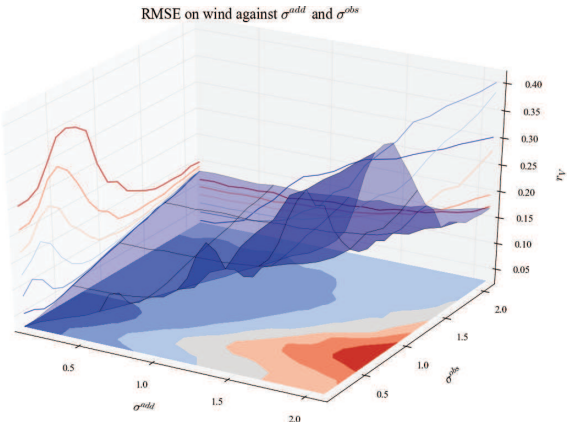


Figure C.124 – Wind RMSE when σ^{add} and σ^{obs} vary.

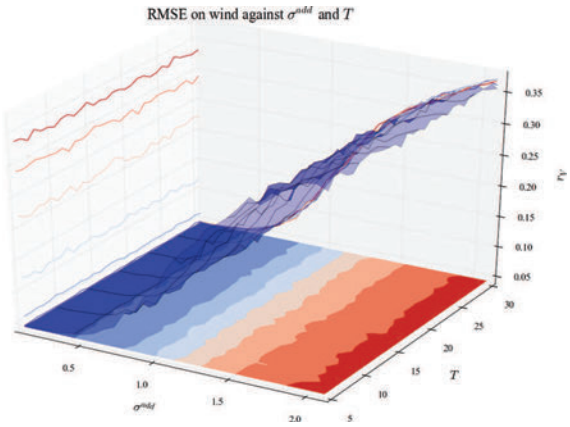


Figure C.125 – Wind RMSE when σ^{add} and τ vary.

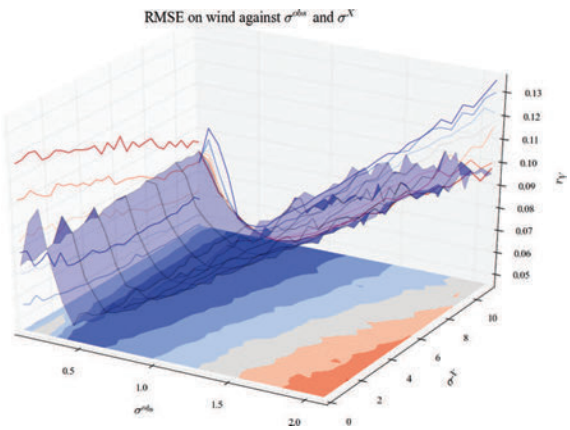


Figure C.126 – Wind RMSE when σ^{obs} and σ^{X} vary.

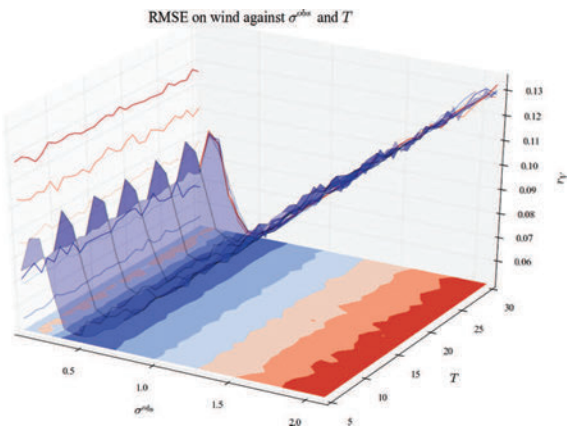


Figure C.127 – Wind RMSE when σ^{obs} and τ vary.

C.3.5 Execution time

In this subsection are shown only the influence on the execution time T_{exe}. First are shown the Sobol indices for this output the most informative experiment and then the rest of experiments. To browse by experiment, use the hyperlinks in table C.3 (all outputs are computed for each experiment). To check another output, go back to table C.2. There is also a list of figures page 384.

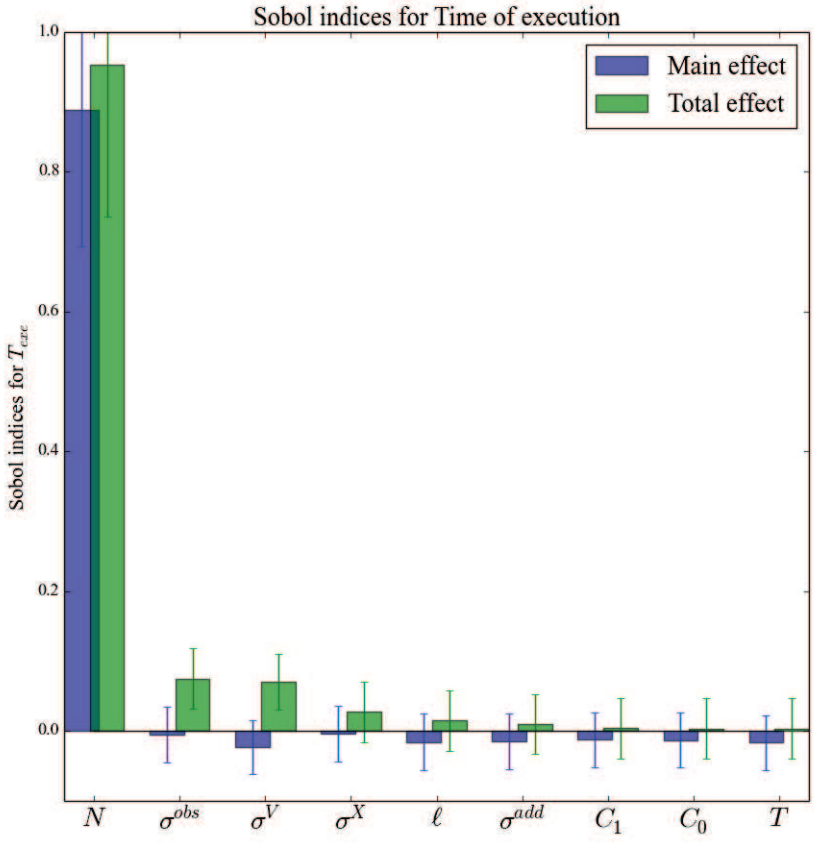


Figure C.128 – Sobol indices (score of influence) for execution time. Main effect in blue (effect of input alone), total effect in green (effect of input with all its interactions).

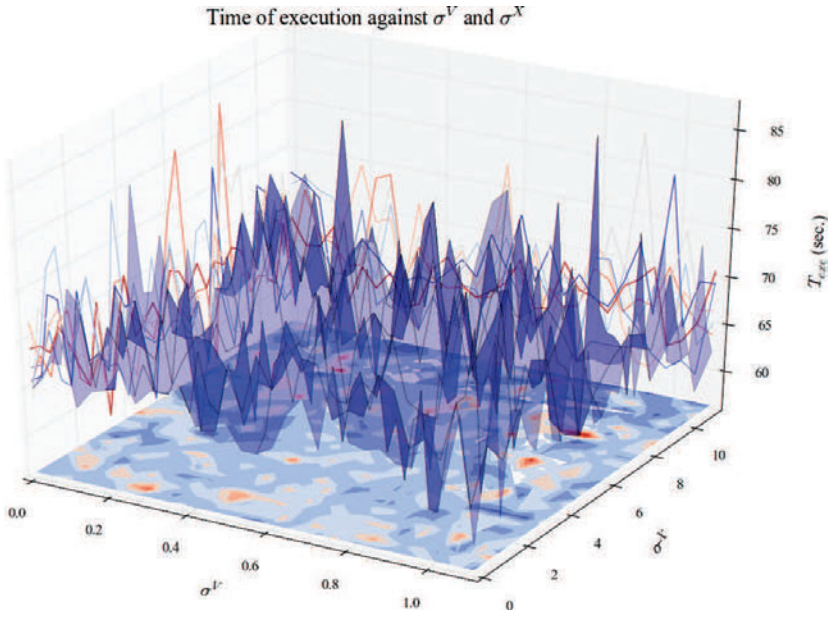


Figure C.129 – Execution time when σ^V and σ^X vary.

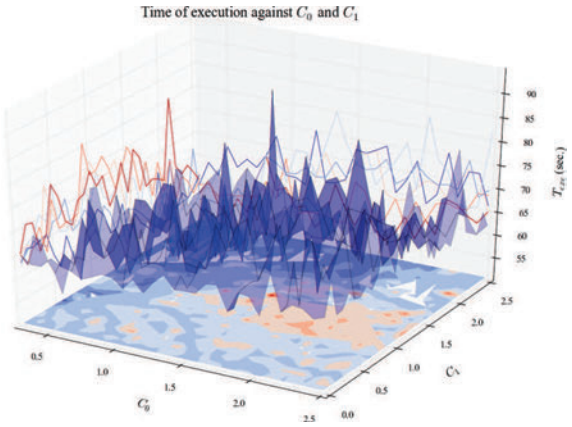


Figure C.130 – Execution time when C_0 and C_1 vary.

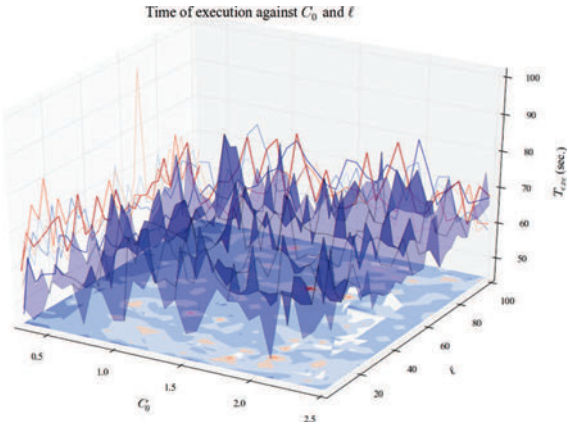


Figure C.131 – Execution time when C_0 and ℓ vary.

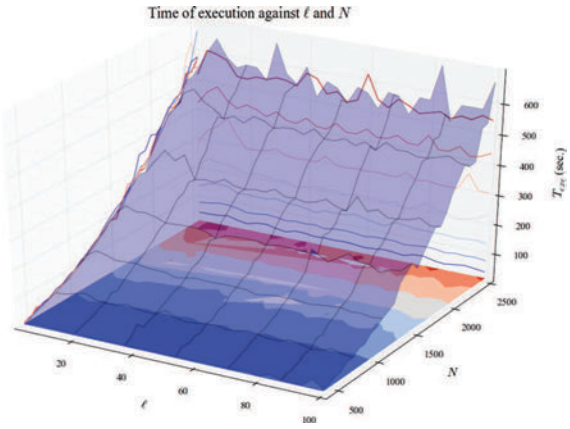


Figure C.132 – Execution time when ℓ and N vary.

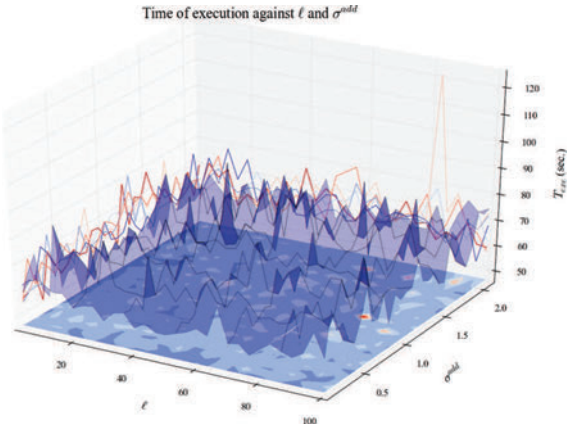


Figure C.133 – Execution time when ℓ and σ^{add} vary.

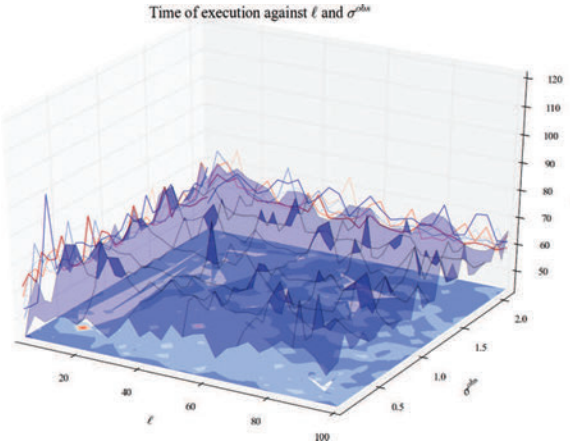


Figure C.134 – Execution time when ℓ and σ^{obs} vary.

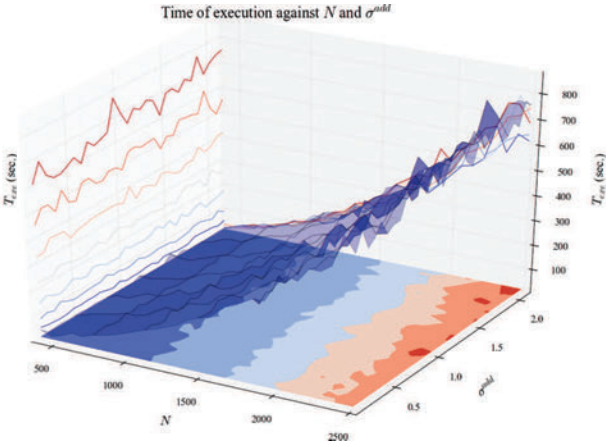


Figure C.135 – Execution time when N and σ^{add} vary.

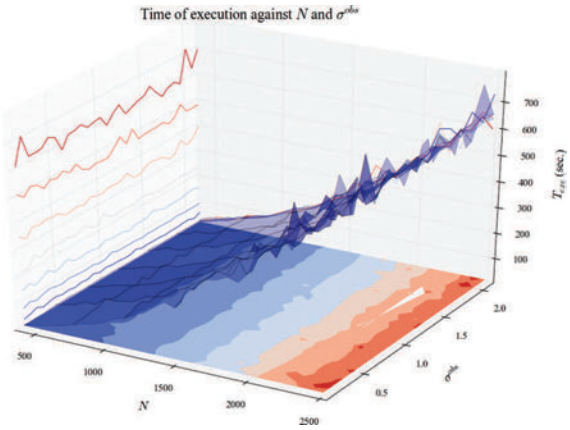


Figure C.136 – Execution time when N and σ^{obs} vary.

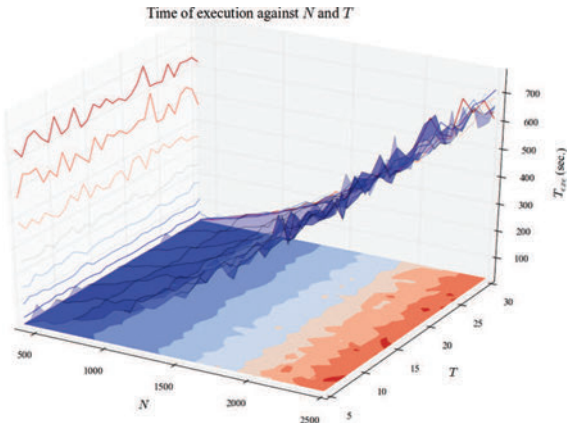


Figure C.137 – Execution time when N and τ vary.

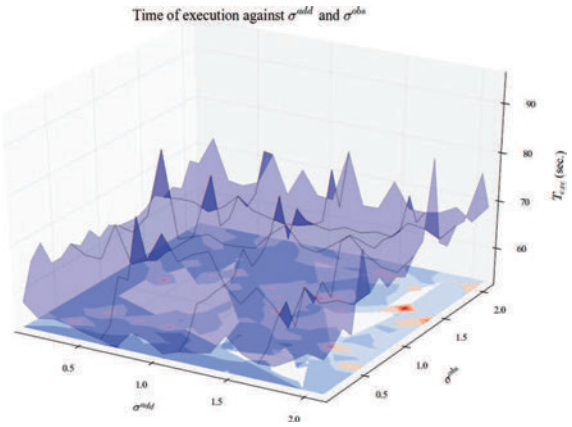


Figure C.138 – Execution time when σ^{add} and σ^{obs} vary.

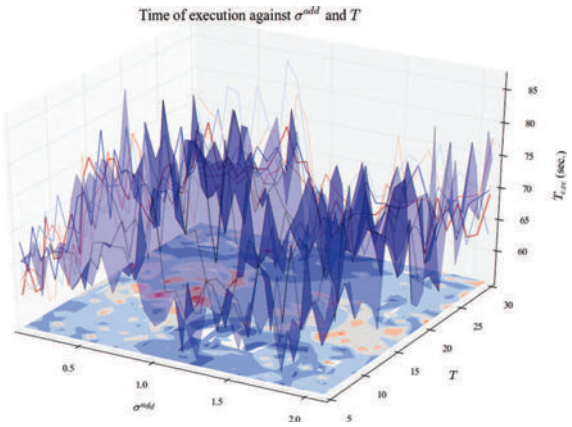


Figure C.139 – Execution time when σ^{add} and τ vary.

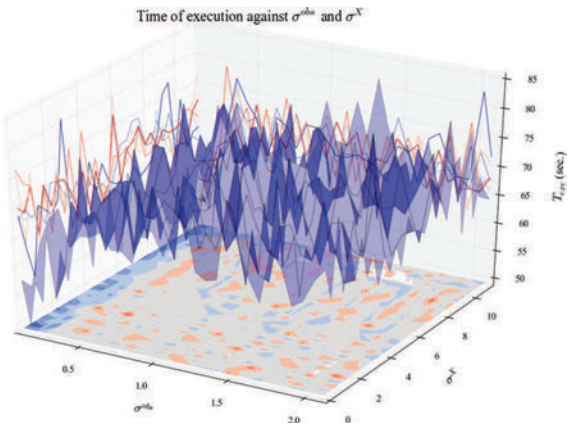


Figure C.140 – Execution time when σ^{obs} and σ^{X} vary.

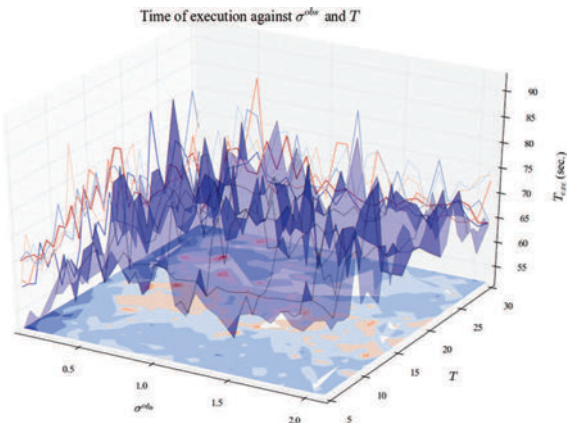


Figure C.141 – Execution time when σ^{obs} and τ vary.

Bibliography

(2010). Weather-related aviation accident study.

Akaike, H. (1998). Information theory and an extension of the maximum likelihood principle. In *Selected Papers of Hirotugu Akaike*, pages 199–213. Springer.

Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on signal processing*, 50(2):174–188.

Baehr, C. (2008). *Modélisation probabiliste des écoulements atmosphériques turbulents afin d'en filtrer la mesure par approche particulaire.* PhD thesis, Université Paul Sabatier-Toulouse III.

Baehr, C. (2010). Nonlinear filtering for observations on a random vector field along a random path. Application to atmospheric turbulent velocities. *ESAIM. Mathematical Modelling and Numerical Analysis*, 44(5):921.

Baehr, C., Beigbeder, C., Couvreux, F., Dabas, A., and Piguet, B. (2011). Retrieval of the turbulent and backscattering properties using a non-linear filtering technique applied to doppler lidar observations.

Baehr, C., Pannekoucke, O., et al. (2010). Some issues and results on the enkf and particle filters for meteorological models. *Chaotic Systems: Theory and Applications*, pages 27–24.

Baehr, C., Rottner, L., Suzat, F., and Rieutord, T. (2016). Procédé de construction d'une représentation tridimensionnelle d'une atmosphère, dispositif et programme correspondant.

Banta, R. M., Pichugina, Y. L., Kelley, N. D., Hardesty, R. M., and Brewer, W. A. (2013). Wind energy meteorology: insight into wind properties in the turbine-rotor layer of the atmosphere from high-resolution doppler lidar. *Bulletin of the American Meteorological Society*, 94(6):883–902.

Barbe, P. and Ledoux, M. (1998). *Probabilité (L3M1).* EDP Sciences.

Bengtsson, T., Bickel, P., Li, B., et al. (2008). Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems. In *Probability and statistics: Essays in honor of David A. Freedman*, pages 316–334. Institute of Mathematical Statistics.

Bennett, M., Christie, S., Graham, A., and Raper, D. (2010). Lidar observations of aircraft exhaust plumes. *Journal of Atmospheric and Oceanic Technology*, 27(10):1638–1651.

Bernardin, F., Bossy, M., Chauvin, C., Drobinski, P., Rousseau, A., and Salameh, T. (2009). Stochastic downscaling method: application to wind refinement. *Stochastic Environmental Research and Risk Assessment*, 23(6):851–859.

Bernardin, F., Bossy, M., Chauvin, C., Jabir, J.-F., and Rousseau, A. (2010). Stochastic lagrangian method for downscaling problems in computational fluid dynamics. *ESAIM: Mathematical Modelling and Numerical Analysis*, 44(05):885–920.

Bettonvil, B. and Kleijnen, J. P. (1997). Searching for important factors in simulation models with many factors: Sequential bifurcation. *European Journal of Operational Research*, 96(1):180–194.

Borgonovo, E. (2007). A new uncertainty importance measure. *Reliability Engineering & System Safety*, 92(6):771–784.

Bougeault, P. and Lacarrere, P. (1989). Parameterization of orography-induced turbulence in a mesobeta–scale model. *Monthly Weather Review*, 117(8):1872–1890.

Box, G. E., Muller, M. E., et al. (1958). A note on the generation of random normal deviates. *The annals of mathematical statistics*, 29(2):610–611.

Breiman, L. (1995). Better subset regression using the nonnegative garrote. *Technometrics*, 37(4):373–384.

Bromiley, P. (2003). Products and convolutions of gaussian probability density functions. *Tina-Vision Memo*, 3.

Burgers, G., Jan van Leeuwen, P., and Evensen, G. (1998). Analysis scheme in the ensemble kalman filter. *Monthly weather review*, 126(6):1719–1724.

Cariou, J.-P., Sauvage, L., Thobois, L., Gorju, G., Machta, M., Lea, G., and Duboué, M. (2011). Long range scanning pulsed coherent lidar for real time wind monitoring in the planetary boundary layer. *16th CLRC*.

Chan, P. (2011). Generation of an eddy dissipation rate map at the hong kong international airport based on doppler lidar data. *Journal of atmospheric and oceanic technology*, 28(1):37–49.

Chan, P., Shun, C., and Wu, K. (2006). Operational lidar-based system for automatic windshear alerting at the hong kong international airport. In *12th Conference on Aviation, Range, and Aerospace Meteorology*.

Chan, P. W. and Shao, A. M. (2007). Depiction of complex airflow near hong kong international airport using a doppler lidar with a two-dimensional wind retrieval technique. *Meteorologische Zeitschrift*, 16(5):491–504.

Chassaing, P. (2000). *Turbulence en mécanique des fluides*.

Chastaing, G. (2013). *Indices de Sobol généralisés pour variables dépendantes*. PhD thesis, Université de Grenoble.

Chastaing, G., Gamboa, F., and Prieur, C. (2015). Generalized sobol sensitivity indices for dependent variables: numerical methods. *Journal of Statistical Computation and Simulation*, 85(7):1306–1333.

Chastaing, G., Gamboa, F., Prieur, C., et al. (2012). Generalized hoeffding-sobol decomposition for dependent variables-application to sensitivity analysis. *Electronic Journal of Statistics*, 6:2420–2448.

Cohn, S. A. (1995). Radar measurements of turbulent eddy dissipation rate in the troposphere: A comparison of techniques. *Journal of Atmospheric and Oceanic Technology*, 12(1):85–95.

Cukier, R., Levine, H., and Shuler, K. (1978). Nonlinear sensitivity analysis of multiparameter model systems. *Journal of computational physics*, 26(1):1–42.

Da Veiga, S. (2015). Global sensitivity analysis with dependence measures. *Journal of Statistical Computation and Simulation*, 85(7):1283–1305.

Dabas, A. (1999). Semiempirical model for the reliability of a matched filter frequency estimator for doppler lidar. *Journal of Atmospheric and Oceanic Technology*, 16(1):19–28.

Dabas, A. M., Drobinski, P., and Flamant, P. H. (1998). Chirp-induced bias in velocity measurements by a coherent doppler co2 lidar. *Journal of Atmospheric and Oceanic Technology*, 15(2):407–415.

Dabas, A. M., Drobinski, P., and Flamant, P. H. (1999). Adaptive filters for frequency estimate of heterodyne doppler lidar returns: Recursive implementation and quality control. *Journal of Atmospheric and Oceanic Technology*, 16(3):361–372.

Dabas, A. M., Drobinski, P., and Flamant, P. H. (2000). Velocity biases of adaptive filter estimates in heterodyne doppler lidar measurements. *Journal of Atmospheric and Oceanic Technology*, 17(9):1189–1202.

Darbieu, C., Lohou, F., Lothon, M., Vilà-Guerau de Arellano, J., Couvreux, F., Durand, P., Pino, D., Patton, E. G., Nilsson, E., Blay-Carreras, E., et al. (2015). Turbulence vertical structure of the boundary layer during the afternoon transition. *Atmospheric Chemistry and Physics*, 15(17):10071–10086.

Das, S. and Durbin, P. (2005). A lagrangian stochastic model for dispersion in stratified turbulence. *Physics of Fluids (1994-present)*, 17(2):025109.

de Moor, G. (2006). *Couche limite atmospherique et turbulence*. Ministère des transports Direction de la météorologie.

Del Moral, P. (2004). *Feynman-Kac Formulae*. Springer.

Doucet, A., Godsill, S., and Andrieu, C. (2000). On sequential monte carlo sampling methods for bayesian filtering. *Statistics and computing*, 10(3):197–208.

DU, S. (1997). Universality of the lagrangian velocity structure function constant (c0) across different kinds of turbulence. *Boundary-Layer Meteorology*, 83(2):207–219.

Du, S., Sawford, B. L., Wilson, J. D., and Wilson, D. J. (1995). Estimation of the kolmogorov constant (c0) for the lagrangian structure function, using a second-order lagrangian model of grid turbulence. *Physics of Fluids (1994-present)*, 7(12):3083–3090.

Einstein, A. (1905). Un the movement of small particles suspended in statiunary liquids required by the molecular-kinetic theory 0f heat. *Ann. d. Phys*, 17:549–560.

Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans*, 99(C5):10143–10162.

Evensen, G. (2003). The ensemble kalman filter: Theoretical formulation and practical implementation. *Ocean dynamics*, 53(4):343–367.

Fong, W., Godsill, S. J., Doucet, A., and West, M. (2002). Monte carlo smoothing with application to audio signal enhancement. *Signal Processing, IEEE Transactions on*, 50(2):438–449.

Frehlich, R. (1997). Effects of wind turbulence on coherent doppler lidar performance. *Journal of Atmospheric and Oceanic Technology*, 14(1):54–75.

Frehlich, R. (2001). Estimation of velocity error for doppler lidar measurements. *Journal of Atmospheric and Oceanic Technology*, 18(10):1628–1639.

Frehlich, R. and Yadlowsky, M. (1994). Performance of mean-frequency estimators for doppler radar and lidar. *Journal of atmospheric and oceanic technology*, 11(5):1217–1230.

Frisch, U. (1995). *Turbulence: the legacy of AN Kolmogorov*. Cambridge university press.

Frisch, U., Sulem, P.-L., and Nelkin, M. (1978). A simple dynamical model of intermittent fully developed turbulence. *Journal of Fluid Mechanics*, 87(04):719–736.

Fruth, J., Roustant, O., and Kuhnt, S. (2011). Total interaction indices for the decomposition of functions with high complexity. *HAL: http://hal. archives-ouvertes. fr/hal-00631066/en*.

Fruth, J., Roustant, O., and Kuhnt, S. (2014). Total interaction index: A variance-based sensitivity index for second-order interaction screening. *Journal of Statistical Planning and Inference*, 147:212–223.

Fruth, J., Roustant, O., and Muehlenstaedt, T. (2013). The fanovagraph package: Visualization of interaction structures and construction of block-additive kriging models.

Furnival, G. M. and Wilson, R. W. (1974). Regressions by leaps and bounds. *Technometrics*, 16(4):499–511.

Gamboa, F., Janon, A., Klein, T., Lagnoux, A., et al. (2014). Sensitivity analysis for multidimensional and functional outputs. *Electronic Journal of Statistics*, 8(1):575–603.

Gardiner, C. (2009). *Stochastic methods*. Number ISBN 978-3-540-70712-7. Springer Berlin, 4th edition.

Germano, M. (1992). Turbulence: the filtering approach. *Journal of Fluid Mechanics*, 238:325–336.

Ginsbourger, D. (2009). *Multiples metamodeles pour l'approximation et l'optimisation de fonctions numeriques multivariables*. PhD thesis, Ecole Nationale Supérieure des Mines de Saint-Etienne.

Grund, C. J., Banta, R. M., George, J. L., Howell, J. N., Post, M. J., Richter, R. A., and Weickmann, A. M. (2001). High-resolution doppler lidar for boundary layer and cloud research. *Journal of Atmospheric and Oceanic Technology*, 18(3):376–393.

Gryning, S.-E., Mikkelsen, T., Baehr, C., Dabas, A., Gómez, P., O'Connor, E., Rottner, L., Sjöholm, M., Suomi, I., and Vasiljevi, N. (2017). *Renewable Energy Forecasting*. Number ISBN 978-0-08-100504-0. Elsevier.

Gustafsson, F., Gunnarsson, F., Bergman, N., Forssell, U., Jansson, J., Karlsson, R., and Nordlund, P.-J. (2002). Particle filters for positioning, navigation, and tracking. *Signal Processing, IEEE Transactions on*, 50(2):425–437.

Gustafsson, F. and Hendeby, G. (2012). Some relations between extended and unscented kalman filters. *IEEE Transactions on Signal Processing*, 60(2):545–555.

Hall, F. F., Huffaker, R. M., Hardesty, R. M., Jackson, M., Lawrence, T. R., Post, M. J., Richter, R., and Weber, B. (1984). Wind measurement accuracy of the noaa pulsed infrared doppler lidar. *Applied optics*, 23(15):2503–2506.

Hamby, D. (1994). A review of techniques for parameter sensitivity analysis of environmental models. *Environmental monitoring and assessment*, 32(2):135–154.

Hesterberg, T., Choi, N. H., Meier, L., Fraley, C., et al. (2008). Least angle and l1 penalized regression: A review. *Statistics Surveys*, 2:61–93.

Hinton, D. A., Charnock, J. K., and Bagwell, D. R. (2000). Design of an aircraft vortex spacing system for airport capacity improvement.

Holton, J. R. and Hakim, G. J. (2012). *An introduction to dynamic meteorology*, volume 88. Academic press.

Homma, T. and Saltelli, A. (1996). Importance measures in global sensitivity analysis of nonlinear models. *Reliability Engineering & System Safety*, 52(1):1–17.

Hull, T. E. and Dobell, A. R. (1962). Random number generators. *SIAM review*, 4(3):230–254.

Iooss, B. and Lemaître, P. (2015). A review on global sensitivity analysis methods. In *Uncertainty Management in Simulation-Optimization of Complex Systems*, pages 101–122. Springer.

Jacques, J. (2005). *Contributions à l'analyse de sensibilité et à l'analyse discriminante généralisée*. PhD thesis, Université Joseph-Fourier-Grenoble I.

Jacques, J., Lavergne, C., and Devictor, N. (2006). Sensitivity analysis in presence of model uncertainty and correlated inputs. *Reliability Engineering & System Safety*, 91(10):1126–1134.

Janon, A. (2012). *Réduction de dimension et Analyse de sensibilité*. PhD thesis, Université de Grenoble.

Janon, A., Klein, T., Lagnoux, A., Nodet, M., and Prieur, C. (2014). Asymptotic normality and efficiency of two sobol index estimators. *ESAIM: Probability and Statistics*, 18:342–364.

Javaheri, A., Lautier, D., and Galli, A. (2003). Filtering in finance. *Wilmott*, 3:67–83.

Jennings, S. (1988). The mean free path in air. *Journal of Aerosol Science*, 19(2):159–166.

Julier, S. J. (2002). The scaled unscented transformation. In *American Control Conference, 2002. Proceedings of the 2002*, volume 6, pages 4555–4559. IEEE.

Julier, S. J. and Uhlmann, J. K. (1997). A new extension of the kalman filter to nonlinear systems. In *Int. symp. aerospace/defense sensing, simul. and controls*, volume 3, pages 3–2. Orlando, FL.

Julier, S. J. and Uhlmann, J. K. (2004). Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422.

Kalman, R. E. et al. (1960). A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45.

Kolmogorov, A. N. (1941). The local structure of turbulence in incompressible viscous fluid for very large reynolds numbers. In *Dokl. Akad. Nauk SSSR*, volume 30, pages 299–303.

Köpp, F., Schwiesow, R., and Werner, C. (1983). Remote measurements of boundary-layer wind profiles using a cw doppler lidar. *Journal of climate and applied meteorology*, 23(1):148–154.

Kouznetsov, R., Kramar, V., Beyrich, F., and Engelbart, D. (2004). Sodar-based estimation of tke and momentum flux profiles in the atmospheric boundary layer: Test of a parameterization model. *Meteorology and Atmospheric Physics*, 85(1):93–99.

Kurowicka, D. and Cooke, R. M. (2006). *Uncertainty analysis with high dimensional dependence modelling*. John Wiley & Sons.

Kusiak, A. and Song, Z. (2010). Design of wind farm layout for maximum wind energy capture. *Renewable Energy*, 35(3):685–694.

Launder, B. E. and Spalding, D. (1974). The numerical computation of turbulent flows. *Computer methods in applied mechanics and engineering*, 3(2):269–289.

Le Gland, F. (2009). Introduction au filtrage en temps discret-filtre de kalman, filtrage particulaire, modeles de markov cachés. *Ecole Nationale Supérieure de Techniques Avancées*.

Le Gland, F., Monbet, V., Tran, V.-D., et al. (2009). Large sample asymptotics for the ensemble kalman filter.

Lenschow, D., Mann, J., and Kristensen, L. (1994). How long is long enough when measuring fluxes and other turbulence statistics? *Journal of Atmospheric and Oceanic Technology*, 11(3):661–673.

Levin, M. (1965). Power spectrum parameter estimation. *IEEE Transactions on Information Theory*, 11(1):100–107.

Lin, D., Foster, D. P., and Ungar, L. H. (2010). A risk ratio comparison of l0 and l1 penalized regressions. *University of Pennsylvania, techical report*.

Lin, D., Pitler, E., Foster, D. P., and Ungar, L. H. (2008). In defense of l0. In *Workshop on Feature Selection,(ICML 2008)*.

Lopes, H. F. and Tsay, R. S. (2011). Particle filters and bayesian inference in financial econometrics. *Journal of Forecasting*, 30(1):168–209.

Lothon, M., Lohou, F., Pino, D., Couvreux, F., Pardyjak, E., Reuder, J., Vilà-Guerau de Arellano, J., Durand, P., Hartogensis, O., Legain, D., et al. (2014). The bllast field experiment: Boundary-layer late afternoon and sunset turbulence. *Atmospheric chemistry and physics*, 14(20):10931–10960.

Maalouf, C. B. (2010). *Etude des phénomenes tourbillonnaires dans le sillage éolien*. PhD thesis, Arts et Métiers ParisTech.

Mahmoud, M. S. and Khalid, H. M. (2013). Distributed kalman filtering: a bibliographic review. *IET Control Theory & Applications*, 7(4):483–501.

Malardel, S. (2005). *Fondamentaux de Météorologie)*. Cépaduès Éditions. ISBN 2.85428.851.3.

Mandel, J., Cobb, L., and Beezley, J. D. (2011). On the convergence of the ensemble kalman filter. *Applications of Mathematics*, 56(6):533–541.

Marrel, A., Iooss, B., Da Veiga, S., and Ribatet, M. (2012). Global sensitivity analysis of stochastic computer models with joint metamodels. *Statistics and Computing*, 22(3):833–847.

Marrel, A., Iooss, B., Laurent, B., and Roustant, O. (2009). Calculations of sobol indices for the gaussian process metamodel. *Reliability Engineering & System Safety*, 94(3):742–751.

Marsaglia, G. et al. (2003). Xorshift rngs. *Journal of Statistical Software*, 8(14):1–6.

Maskell, S. and Gordon, N. (2001). A tutorial on particle filters for on-line nonlinear/non-gaussian bayesian tracking. In *Target tracking: algorithms and applications (Ref. No. 2001/174), IEE*, pages 2–1. IET.

Meneveau, C. and Sreenivasan, K. (1987). Simple multifractal cascade model for fully developed turbulence. *Physical review letters*, 59(13):1424.

Menzies, R. T. and Hardesty, R. M. (1989). Coherent doppler lidar for measurements of wind fields. *Proceedings of the IEEE*, 77(3):449–462.

Miller, S. and Childers, D. (2012). *Probability and random processes: With applications to signal processing and communications*. Academic Press.

Monin, A. S. and Yaglom, A. M. (1963). On the laws of small-scale turbulent flow of liquids and gases. *Russian Mathematical Surveys*, 18(5):89–109.

Mordant, N., Metz, P., Michel, O., and Pinton, J.-F. (2001). Measurement of lagrangian velocity in fully developed turbulence. *Physical Review Letters*, 87(21):214501.

Morris, M. D. (1991). Factorial sampling plans for preliminary computational experiments. *Technometrics*, 33(2):161–174.

Natarajan, B. K. (1995). Sparse approximate solutions to linear systems. *SIAM journal on computing*, 24(2):227–234.

Neiman, P. J., Hardesty, R., Shapiro, M., and Cupp, R. (1988). Doppler lidar observations of a downslope windstorm. *Monthly weather review*, 116(11):2265–2275.

O'Connor, E. J., Illingworth, A. J., Brooks, I. M., Westbrook, C. D., Hogan, R. J., Davies, F., and Brooks, B. J. (2010). A method for estimating the turbulent kinetic energy dissipation rate from a vertically pointing doppler lidar, and independent evaluation from balloon-borne in situ measurements. *Journal of atmospheric and oceanic technology*, 27(10):1652–1664.

Øksendal, B. (2013). *Stochastic differential equations: an introduction with applications*. Number ISBN 3-540-60243-7. Springer Science & Business Media, 4th edition.

Owen, A. B. (2013). Better estimation of small sobol' sensitivity indices. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 23(2):11.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pham, D. T. (2001). Stochastic methods for sequential data assimilation in strongly nonlinear systems. *Monthly weather review*, 129(5):1194–1207.

Pichugina, Y. L., Banta, R. M., Brewer, W. A., Sandberg, S. P., and Hardesty, R. M. (2011). Doppler lidar-based wind-profile measurement system for offshore wind-energy and other marine boundary layer applications. *Journal of Applied Meteorology and Climatology*, 51(2):327–349.

Pichugina, Y. L., Tucker, S. C., Banta, R. M., Brewer, W. A., Kelley, N. D., Jonkman, B. J., and Newsom, R. K. (2008). Horizontal-velocity and variance measurements in the stable boundary layer using doppler lidar: Sensitivity to averaging procedures. *Journal of Atmospheric and Oceanic Technology*, 25(8):1307–1327.

Pope, S. (1985). Pdf methods for turbulent reactive flows. *Progress in Energy and Combustion Science*, 11(2):119–192.

Pope, S. (1994). Lagrangian pdf methods for turbulent flows. *Annual review of fluid mechanics*, 26(1):23–63.

Pope, S. B. (2000). *Turbulent flows*. Cambridge university press.

Priestley, M. B. (1981). *Spectral analysis and time series*. Academic press.

Reif, K., Gunther, S., Yaz, E., and Unbehauen, R. (1999). Stochastic stability of the discrete-time extended kalman filter. *IEEE Transactions on Automatic control*, 44(4):714–728.

Rieutord, T., Baehr, C., and Rottner, L. (2016). Doppler lidar based reconstruction for fast estimation of turbulence: contribution and validation. In *18th International Symposium for the Advancement of Boundary-Layer Remote Sensing*.

Rotach, M. W. (2001). Simulation of urban-scale dispersion using a lagrangian stochastic dispersion model. *Boundary-Layer Meteorology*, 99(3):379–410.

Rottner, L. (2015). *Reconstruction de l'atmosphère turbulente à patir d'un lidar Doppler 3D et étude du couplage avec Meso-NH.* PhD thesis, Université Paul Sabatier-Toulouse III.

Rottner, L. and Baehr, C. (2014). 3D wind reconstruction and turbulence estimation in the boundary layer from doppler lidar measurements using particle method. http://adsabs.harvard.edu/abs/2014AGUFMNG23B3806R.

Rottner, L. and Baehr, C. (2017). Stochastic method for turbulence estimation from doppler lidar measurements. *Atmospheric Measurement Techniques*. submitted.

Rottner, L., Baehr, C., Couvreux, F., Canut, G., and Rieutord, T. (2017). A new downscaling method for sub-grid turbulence modeling. *Atmospheric Chemistry and Physics*, 17(11):6531–6546.

Rottner, L., Suomi, I., Rieutord, T., Baehr, C., and Gryning, S.-E. (2016). Real time turbulence and wind gust estimation from wind lidar observations using the turbulence reconstruction method. In *18th International Symposium for the Advancement of Boundary-Layer Remote Sensing*.

Rousseau, A., Bernardin, F., Bossy, M., Salameh, T., and Drobinski, P. (2007). Stochastic particle method applied to local wind simulation. In *ICCEP'07-International Conference for Clean Electrical Power*, pages 526–528.

Roustant, O., Ginsbourger, D., and Deville, Y. (2012). Dicekriging, diceoptim: Two r packages for the analysis of computer experiments by kriging-based metamodeling and optimization.

Rubin, D. B. (1976). Inference and missing data. *Biometrika*, 63(3):581–592.

Saltelli, A. (2002a). Making best use of model evaluations to compute sensitivity indices. *Computer Physics Communications*, 145(2):280–297.

Saltelli, A. (2002b). Sensitivity analysis for importance assessment. *Risk Analysis*, 22(3):579–590.

Saltelli, A. and Annoni, P. (2010). How to avoid a perfunctory sensitivity analysis. *Environmental Modelling & Software*, 25(12):1508–1517.

Saltelli, A., Annoni, P., Azzini, I., Campolongo, F., Ratto, M., and Tarantola, S. (2010). Variance based sensitivity analysis of model output. design and estimator for the total sensitivity index. *Computer Physics Communications*, 181(2):259–270.

Saltelli, A., Tarantola, S., and Campolongo, F. (2000). Sensitivity analysis as an ingredient of modeling. *Statistical Science*, pages 377–395.

Saporta, G. (2006). *Probabilités, analyse des données et statistique*. Editions Technip.

Sathe, A. and Mann, J. (2013). A review of turbulence measurements using ground-based wind lidars. *Atmospheric Measurement Techniques*, 6.

Schafer, J. L. and Graham, J. W. (2002). Missing data: our view of the state of the art. *Psychological methods*, 7(2):147.

Schlipf, D. (2014). Lidar assisted control of wind turbines. *Struttgart Wind Energy. Universitat Stuttgart*.

Schmid, H. P. (2002). Footprint modeling for vegetation atmosphere exchange studies: a review and perspective. *Agricultural and Forest Meteorology*, 113(1):159–183.

Smalikho, I., Köpp, F., and Rahm, S. (2005). Measurement of atmospheric turbulence by 2-μ m doppler lidar. *Journal of Atmospheric and Oceanic Technology*, 22(11):1733–1747.

Smith, A., Doucet, A., de Freitas, N., and Gordon, N. (2013). *Sequential Monte Carlo methods in practice*. Springer Science & Business Media.

Snyder, C., Bengtsson, T., Bickel, P., and Anderson, J. (2008). Obstacles to high-dimensional particle filtering. *Monthly Weather Review*, 136(12):4629–4640.

Sobol, I. M. (1976). Uniformly distributed sequences with an additional uniform property. *USSR Computational Mathematics and Mathematical Physics*, 16(5):236–242.

Sobol', I. M. (1990). On sensitivity estimation for nonlinear mathematical models. *Matematicheskoe Modelirovanie*, 2(1):112–118.

Sobol, I. M. (1993). Sensitivity estimates for nonlinear mathematical models. *Mathematical Modelling and Computational Experiments*, 1(4):407–414.

Sobol, I. M. (2001). Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. *Mathematics and computers in simulation*, 55(1):271–280.

Sommerfeld, M. (2001). Validation of a stochastic lagrangian modelling approach for inter-particle collisions in homogeneous isotropic turbulence. *International Journal of Multiphase Flow*, 27(10):1829–1858.

Stull, R. B. (1988). *An introduction to boundary layer meteorology*, volume 13. Springer.

Sudret, B. (2008). Global sensitivity analysis using polynomial chaos expansions. *Reliability Engineering & System Safety*, 93(7):964–979.

Sun, X., Huang, D., and Wu, G. (2012). The current state of offshore wind energy technology development. *Energy*, 41(1):298–312.

Suzat, F., Baehr, C., and Dabas, A. (2011). A fast atmospheric turbulent parameters estimation using particle filtering. application to lidar observations. In *Journal of Physics: Conference Series*, volume 318, page 072019. IOP Publishing.

Thomson, D. and Wilson, J. (2013). History of lagrangian stochastic models for turbulent dispersion. *Lagrangian Modeling of the Atmosphere*, pages 19–36.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288.

Tucker, S. C., Senff, C. J., Weickmann, A. M., Brewer, W. A., Banta, R. M., Sandberg, S. P., Law, D. C., and Hardesty, R. M. (2009). Doppler lidar estimation of mixing height using turbulence, shear, and aerosol profiles. *Journal of Atmospheric and Oceanic Technology*, 26(4):673–688.

van de Geer, S. A. (2016). *Estimation and testing under sparsity*. Springer.

van Leeuwen, P. J. (2003). A variance-minimizing filter for large-scale applications. *Monthly Weather Review*, 131(9):2071–2084.

van Leeuwen, P. J. (2009). Particle filtering in geophysical systems. *Monthly Weather Review*, 137(12).

Welch, G. and Bishop, G. (1995). An introduction to the kalman filter.

Wharton, S. and Lundquist, J. K. (2012). Atmospheric stability affects wind turbine power collection. *Environmental Research Letters*, 7(1):014005.

Wilson, J. D. and Sawford, B. L. (1996). Review of lagrangian stochastic models for trajectories in the turbulent atmosphere. *Boundary-layer meteorology*, 78(1-2):191–210.

Yakhot, V. (2008). Lagrangian structure functions in turbulence: Scaling exponents and universality. *arXiv preprint arXiv:0810.2955*.

List of figures

List of tables

List of definitions

List of theorems

Sensitivity analysis of a filtering algorithm for wind lidar measurements

Thomas Rieutord

Centre National de Recherches Météorologiques (UMR 3589)
Météo-France, CNRS
42 avenue Gaspard Coriolis, Toulouse, France.

Abstract — Wind energy industry and airport safety are in need of atmospheric observations. Remote sensors, such as lidars, are well proven and common technology to provide wind measurements in the first hundreds of meters of altitude. However, acquisition abilities of lidars are polluted by measurement noise. Using non-linear filtering techniques, we took part at the development of an algorithm improving wind and turbulence estimations. The process is based on a representation of the atmosphere with fluid particles. It uses a stochastic Lagrangian model of turbulence and a genetic selection filtering technique. Its efficiency depends of the setting of various parameters. Their values were fixed experimentally during the development phase. But their influence has never been assessed. This work addresses this question with a variance-based sensitivity analysis. New estimators of Sobol indices, using penalized regression have been tested. These estimators ensure the lowest Sobol indices automatically go to zero so the overall interpretation is simplified. The sensitivity analysis allows to reduce the system from 5 outputs and 9 inputs to 3 inputs (number of particles, real observation noise, observation noise given to the filter) and 2 outputs (wind spectrum slope, root-mean-squared error on wind). With this reduced system we determined a procedure to correctly set the most important parameters. The observation noise given to the filter is well set when the wind spectrum slope has the expected value of -5/3. Once it is set correctly, the error on wind is minimum and its expression is known.

Keywords — *sensitivity analysis, non-linear filtrering, Doppler lidar, turbulence*

PhD delivered by Toulouse INP in November 2017

PhD supervisors – Fabrice Gamboa, Alain Dabas
Speciality – Applied mathematics

Analyse de sensibilité d'un algorithme de filtrage pour les mesures de vent par lidar

Thomas Rieutord

Centre National de Recherches Météorologiques (UMR 3589)
Météo-France, CNRS
42 avenue Gaspard Coriolis, Toulouse, France.

Résumé — L'industrie éolienne et l'aéronautique ont des besoins importants en matière de mesure de vent dans les premières centaines de mètres de l'atmosphère. Les lidars sont des instruments répandus et éprouvés pour ce type de mesure. Cependant, leurs qualités d'acquisition sont atténuées par un bruit de mesure systématique. En utilisant des techniques sur le filtrage non-linéaire nous avons participé au développement d'un algorithme qui améliore l'estimation du vent et de la turbulence. Cet algorithme est basé sur une représentation de l'atmosphère par des particules fluides. Il utilise un modèle lagrangien stochastique de turbulence et un filtrage par sélection génétique. Son efficacité dépend du réglage de certains paramètres, fixés à une valeur acceptable à l'issue de la phase de développement. Mais l'influence de ces paramètres n'a jamais été étudiée. Ce travail de thèse répond à cette question par une analyse de sensibilité basée sur la décomposition de variance. De nouveaux estimateurs pour les indices de Sobol, utilisant des régression pénalisées, ont été testés. Ces estimateurs mettent les indices de Sobol les plus petits automatiquement à zéro pour faciliter l'interprétation globale. L'analyse de sensibilité permet de réduire le système à 9 entrées et 5 sorties à un système de 3 entrées (le nombre de particules, le bruit d'observation réel et le bruit d'observation donné au filtre) et 2 sorties (la pente du spectre de vent et l'erreur sur le vent). Grâce à ce système réduit, nous mettons en évidence une méthode de réglage des paramètres d'entrée les plus importants. Le bruit d'observation donné au filtre est bien réglé lorsque la pente du spectre est à la valeur cible de -5/3. Une fois ce bruit réglé, l'erreur sur le vent est minimale avec une expression connue.

Mots clés — *analyse de sensibilité, filtrage non-linéaire, lidar Doppler, turbulence.*

Doctorat délivré par Toulouse INP en novembre 2017

Directeurs de thèse – Fabrice Gamboa, Alain Dabas
Discipline – Mathématiques appliquées