Reduced-order models under uncertainties for microscale atmospheric pollutant dispersion in urban areas : exploring learning algorithms for high-fidelity model emulation

Bastien Nony

Université de Toulouse

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par l'Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)

Présentée et soutenue le 20 janvier 2023 par :

Bastien Nony

Modèles réduits en présence d'incertitudes pour la dispersion atmosphérique micro-échelle de polluant en milieu urbain : exploration de méthodes d'apprentissage pour l'émulation de modèles haute-fidélité

JURY

Lionel Soulhac	LMFA/INSA Lyon	Rapporteur
Etienne Mémin	Inria	Rapporteur
Amandine Marrel	CEA	Rapporteure
Laure Raynaud	CNRM/Météo-France	Examinatrice
Fabrice Gamboa	IMT/Univ. de Toulouse	Examinateur
Mélanie Rochoux	CECI/CERFACS	Directrice de thèse
Didier Lucor	LISN/CNRS	Co-directeur
Thomas Jaravel	CECI/CERFACS	Co-encadrant

École doctorale et spécialité :
SDU2E : Océan, Atmosphère, Climat

Unité de Recherche :
CECI/Climat Environnement Couplages Incertitudes (UMR 5318)

Directeurs de Thèse :
Mélanie Rochoux et Didier Lucor

Rapporteurs :
Amandine Marrel, Étienne Mémin et Lionel Soulhac

Résumé

En cas de rejet accidentel de substances dangereuses en milieu urbain ou sur un site industriel, cartographier la concentration en polluants est un véritable défi pour évaluer l'exposition du public à des doses toxiques. Il s'agit d'une problématique à la fois opérationnelle et scientifique car l'interaction de la couche limite atmosphérique avec la canopée urbaine rend la dynamique de l'écoulement complexe et nécessite des outils de modélisation physique haute-fidélité. En résolvant explicitement l'essentiel du spectre de turbulence, l'approche de simulation aux grandes échelles (SGE) permet de représenter la variabilité spatio-temporelle de la concentration de polluants dans un environnement complexe. Concevoir une approche qui synthétise cette grande quantité d'informations est d'un grand intérêt pour ensuite l'injecter dans des modèles opérationnels de plus basse fidélité. Néanmoins, dans ce contexte accidentel, l'approche de SGE reste sujette à des incertitudes, à la fois atmosphériques et concernant la source d'émission, et nécessite un cadre de modélisation d'ensemble pour représenter l'éventail des scénarios plausibles de dispersion. Cependant, ce cadre multi-requêtes est inaccessible dans un contexte en temps réel car les simulations SGE reposent sur des moyens de calcul conséquents.

Dans cette thèse, nous explorons différentes approches d'apprentissage statistique pour construire un modèle réduit informé par l'approche de SGE afin d'obtenir des prévisions de concentration physiquement cohérentes, tout en diminuant considérablement le coût de calcul. Cette étude est effectuée sur un cas bidimensionnel de dispersion de traceur dans un écoulement turbulent de couche limite atmosphérique autour d'un obstacle isolé, pour lequel les conditions aux limites sur l'écoulement en entrée du domaine et la localisation de la source sont incertaines.

Dans un premier temps, nous mettons en œuvre une approche de modèle réduit basée sur les données de SGE pour prévoir les statistiques du champ de concentration du traceur. Nous comparons plusieurs approches *i)* de compression (décomposition orthogonale en modes propres/POD versus auto-encodeur) pour réduire la dimension les champs d'intérêt à un nombre limité de variables latentes, et *ii)* différents modèles de régression (e.g. chaos polynomial, processus gaussiens) pour représenter la réponse des variables latentes aux variations des paramètres incertains. La POD combinée à la régression par processus gaussiens permet d'obtenir de bonnes prévisions pour un grand jeu de données de SGE d'entraînement (composé de 450 solutions). L'hétérogénéité de la concentration proche de la source en amont de l'obstacle nécessite un grand nombre de modes POD pour être bien représentée. De plus, la capacité du modèle à réduire la dimension des champs peut être améliorée en remplaçant l'approche POD par un auto-encodeur convolutif.

En réduisant le nombre de données d'entraînement, nous observons que les prévisions du modèle réduit manquent de consistance avec les lois de la physique. Pour surmonter ce problème, dans un deuxième temps, nous mettons en œuvre une approche de modèle réduit hybride basée sur l'équation de transport de traceur RANS (Reynolds-averaged Navier-Stokes) informée par les données de SGE afin d'intégrer des contraintes physiques dans le processus d'apprentissage. L'idée-clé est de découpler les incertitudes atmosphériques des incertitudes de source, et de remplacer les termes classiques de fermeture de la turbulence dans l'approche RANS par des

modèles sur l'écoulement d'air émulés à partir des données de SGE. Cette approche nécessite beaucoup moins de données de SGE (seulement 50 solutions) que le modèle réduit directement émulé à partir des données de SGE. Nous montrons finalement qu'une approche multi-fidélité (combinant un petit nombre de solutions de SGE avec un grand nombre de prévisions du modèle réduit hybride) offre une perspective de recherche intéressante pour optimiser la performance du modèle réduit.

Mots-clés. Dispersion atmosphérique, Modélisation numérique, Mécanique des fluides numériques, SGE, RANS, Incertitudes paramétriques, Obstacle isolé, Apprentissage statistique, Réduction de dimension, Régression, Science des données guidée par la physique, Multi-fidélité

Résumé de vulgarisation. En cas de rejet accidentel de substances dangereuses dans l'atmosphère, certaines zones urbaines peuvent être sujettes à une forte dégradation de la qualité de l'air, ce qui peut avoir des conséquences sur la santé et l'environnement. Identifier ces zones nécessite de recourir à des modèles représentant finement les interactions entre le vent et les bâtiments. Ces modèles reposent sur les équations fondamentales de la physique des écoulements, et un ensemble de simulations est requis pour estimer l'enveloppe des scénarios possibles lors d'un évènement. En raison des moyens de calcul qu'ils nécessitent, ces modèles ne sont pas directement applicables à des cas de grande échelle. Dans ces travaux, nous cherchons à identifier les outils statistiques apparentés à l'intelligence artificielle les plus adaptés pour concevoir un modèle simplifié capable d'intégrer les données venant des modèles physiques, tout en permettant de réaliser des prévisions d'ensemble en temps réel de l'évènement.

Abstract

In the event of an accidental release of hazardous substances in an urban area or on an industrial site, tracking pollutant concentration is particularly important for assessing public exposure to toxic doses. This is an operational but also a scientific challenge, as the interaction of the atmospheric boundary layer with the urban canopy makes the near-surface flow dynamics complex and requires high-fidelity physics modelling tools. By explicitly solving for most of the turbulence spectrum, the large-eddy simulation (LES) approach has the potential to represent the spatial and temporal variability of pollutant concentration in a complex environment. Finding a way to synthesise this large amount of information to inject into lower-fidelity operational models is particularly appealing. Still, in this accidental context, the LES approach remains subject to atmospheric and emission source uncertainties, and requires an ensemble modelling framework to represent the range of plausible dispersion scenarios. But this multi-query framework is far out of reach in a real-time context as LES simulations require very large computational resources.

In this thesis, we explore different statistical learning approaches to design a reduced-order model informed by LES to produce physically consistent concentration predictions, while substantially decreasing computational cost. This study is carried out on a two-dimensional tracer dispersion case in a turbulent atmospheric boundary-layer flow over an isolated obstacle, in which both the inflow boundary condition and source location are uncertain.

In a first step, we design a data-driven reduced-order model approach based on LES data to predict tracer concentration field statistics. We compare several dimension reduction approaches (proper orthogonal decomposition/POD versus autoencoder) to reduce the field statistics to a limited number of latent variables. We also compare several regression models (e.g. polynomial chaos, Gaussian processes) to represent the response of the latent variables to changes in the uncertain parameters. POD combined with Gaussian process regression provides fairly good predictions for a large LES training dataset (made of 450 snapshots). Near-source concentration heterogeneity upstream of the obstacle requires a large number of POD modes to be well captured. Moreover, the field dimension reduction capability of the model can be improved by replacing POD with a convolutional autoencoder.

By reducing the number of training snapshots, we observe a loss of consistency with physics principles in the reduced-order model predictions. To overcome this issue, in a second step, we design a hybrid reduced-order model approach based on a LES-informed Reynolds-averaged Navier-Stokes (RANS) tracer transport equation to integrate physical constraints in the training process. The key idea is to decouple the atmospheric uncertainties from the source location uncertainties and to replace the classical RANS turbulent closure terms with data-driven airflow models emulated from LES data. This approach requires much less LES data (only 50 snapshots) than the LES data-driven reduced-order model. We finally show that a multi-fidelity approach (combining a small number of LES snapshots with a large number of hybrid model predictions) offers an interesting avenue of research to optimise the reduced-order model performance.

Keywords. Atmospheric dispersion, Numerical modelling, Computational fluid dynamics, LES, RANS, Parametric uncertainties, Isolated obstacle, Statistical learning, Dimension reduction, Regression, Physics-guided data science, Multi-fidelity

Plain language summary. In the event of an accidental release of hazardous substances into the atmosphere, some urban areas may be subject to severe air quality degradation, which may have public health and environmental impacts. Mapping these areas requires the use of models that accurately represent the interactions between the wind and the buildings. These models rely on the fundamental equations of fluid dynamics, and an ensemble of simulations is required to assess the envelope of plausible situations during an event. Due to the high computational resources they require, these models cannot be directly applied to large-scale cases. In this work, we seek to identify the statistical tools related to artificial intelligence that are best suited to designing a simplified model able to integrate physical model data while allowing for real-time predictions of the event.

Remerciements

Ce manuscrit fait état de trois années d'apprentissage, et quelle curieuse sensation que d'arriver au bout de cette aventure. Aussi interminables qu'éphémères, quelques trois années d'échanges, de lectures et de lignes de code pour contribuer, je l'espère, aux futures lectures des aventuriers-doctorants à venir. C'est si peu de choses finalement, mais quelle aventure ! Les lignes qui suivent rappelleront, je l'espère, à ceux qui m'ont épaulé des instants précieux que nous avons partagés.

Mélanie, Thomas et Didier, il revient à ma mémoire notre tout premier échange, sous forme d'entretien d'embauche empreint de quelque nervosité pour ma part. Trois ans plus tard, nous voilà au bout du chemin et je tiens à vous exprimer ma gratitude pour la confiance que vous m'avez accordée et pour l'énergie que vous avez déployée pour me former. Didier, avec une touche de modernité, tu as endossé le rôle d'encadrant à distance, la faute aux restrictions imposées par le Covid. Malgré les contraintes, tu as su jongler habilement entre bad cop et grand Manitou pour faire avancer cette thèse. Thomas, tu n'as jamais baissé les bras face à mes yeux vitreux lorsque tu m'initiais aux fondamentaux de la mécanique des fluides. Grâce à toi, cette science m'est aujourd'hui un peu moins mystérieuse. Enfin, Mélanie, tu m'as soutenu, motivé jusque dans les instants difficiles et surtout aidé à grandir. De Toulouse à Oslo, merci d'avoir veillé sur le jeune chercheur Poucet que j'ai été. Ton dévouement professionnel force à l'admiration, et aujourd'hui je te dois beaucoup.

Mes pensées vont à l'ensemble de l'équipe CECI, véritable village peuplé d'irréductibles chercheurs. Pour nos escapades, du Néouvielle à Gruissan en passant par Cauterets, je remercie les copains-doctorants et assimilés qui m'ont servi de guides de montagne et, pour certains m'ont vu remonter sur des skis après dix ans d'absence pour terminer avec une jolie entorse, Susanne, Juliette, Quentin, François, Théo, Mohamed, Svenya, Eliott, Victor, Maxime, Abel, William, Gabriel, Anthony et Suzanne. Un grand merci à Emilia, Christophe et Laure, qui ont transformé le bureau en véritable café bavard. Je tiens également à exprimer ma gratitude envers Laurent, Olivier, Marie-Pierre, Sophie, Margot, et Julien, pour les précieux moments que nous avons partagés. Mention spéciale à Than-Huy à qui j'attribue sans conteste le titre de Castafiore. Chacun de vous apporte une contribution unique à cette équipe. Vous avez joué un rôle essentiel dans mon parcours, et je vous remercie du fond du cœur pour votre expertise et votre bienveillance.

Enfin, il y a de ces relations qui doucement se transforment en amitié. Saloua et Aurélien, je ne sais pas si Michèle avait raison lorsqu'elle nous qualifiait de « piliers de globc », mais vous aurez certainement été les miens. Vous aurez partagé mes moments de joie, de doute et surtout, vous aurez supporté vaillamment ces longues marches qui quotidiennement faisaient le tour du météopôle. Sans oublier Timothée, évidemment, je garderai un souvenir particulièrement doux et amusé de notre Bulle de Saint-Cyprien, de la façon dont Serge Pey a mangé le feu, et de cette phrase curieusement inspirée qui orne désormais l'entrée du bureau : « PLOTO ERGO SUM ». Pierre-Alexandre, toi et moi avons su recréer quelques temps une parodie de Big Bang Theory. Tes manies d'apprenti-cuisinier/diététicien/sociologue/politologue m'ont d'abord laissé

perplexe, puis fasciné. Tu es à toi seul ce curieux mélange entre un biathlète et un physicien un peu fêlé, mais c'est toi alors ne change rien.

Voilà déjà sept ans depuis que j'ai posé le pied à Toulouse pour la première fois. C'est l'occasion de saluer les amis toulousains qui continuent de me soutenir avec une énergie apparemment inépuisable. Mes pensées vont à mon trio nouvellement parisien, Cécile et Claude-Chantal, à mon binôme d'études et confident, Alban, à Robin qui continue de partager mon affection pour les contrepétries du Canard, à Solène qui m'attend toujours pour courir le raid'INSA, à Elodie et à son énergie badinesque, à Colette et à son accent toujours rempli de soleil, à Guillaume et Aurélien pour leurs cours transverses sur War Thunder et la drum and bass, à Julie et Lucas avec qui j'ai vibré sur cette finale manquée des Bleus, à Alexis et Jessica que je reste déterminé à croiser, à Londres, à New-York ou ailleurs (promis Jessica, je ne raterai pas mon avion cette fois-ci) et à Albane qui doit en ce moment-même contempler les merveilles de la Cordillère des Andes.

Un immense merci à mes amis de Monaco, loin des yeux mais près du cœur, Giorgia, Mathieu, Vincent, Sylvie, Ambroise, Stéphane, Jean-Pierre et Françoise, au boys band sophipolitain, Johann, Loïc et Thomas, aux bisounours, Robin, Lilas, Andréa, Maxime, et Mathieu, dont je tairai ici les plus grands exploits (coucou Maxime), à Livio qui après presque vingt ans d'amitié n'est toujours pas à court de canulars absurdes. Enfin, merci à toi Clarisse qui au fil des années est devenue ma tac (ou ma tic à toi de voir). Merci pour ton amitié si précieuse aujourd'hui.

A tous ces amis qu'il me tarde de revoir, à Mari-Carmen qui m'a guidé dans les ruelles asséchées de Grenade, à Juliette et à Serrié & Co. qui participent à rendre la Réunion un petit paradis terrestre, à Éric et à la famille Marchand, fiers représentants du pays roannais, à Eduardo, philosophe sud-américain, grand amateur de café-vapeur.

Je n'oublie pas non plus ces professeurs qui ont su rendre compte de leur passion et me la transmettre, peut-être avec un certain goût du décalage pour certains, Edmond, Pascal, Florence et Thierry. Je garde un doux souvenir de vos enseignements qui ont finalement contribué à ce que je m'attarde jusqu'au doctorat.

Enfin, mes remerciements vont à ma famille. D'abord à mes parents, Marie-Pierre et Christophe, pour vos conseils et votre soutien si précieux. Aujourd'hui, je partage avec vous toute la satisfaction de franchir la ligne d'arrivée car sans vous rien n'aurait été possible. Merci également à mon frère Natan qui vieillis comme le bon vin à une vitesse qui force au respect, à mes grands-parents Marie-France, Régine, Michel et Xavier, qui veillez sur moi d'où vous êtes, à mon oncle et à tes talents humoristiques parfois douteux, à ma tante et à tes paroles apaisantes. Enfin, à Anaïs pour la place que tu occupes aujourd'hui. De confidente à avocate en passant par maître-pancake et styliste, tout y est et j'en suis heureux.

Merci à vous tous, qui m'êtes proches et qui avez participé à ce que ces instants me soient aujourd'hui si doux. Je vous souhaite un bel avenir et que celui-ci fasse sens en cette période où le monde entier semble douter.

Table of contents

Introduction générale

La dispersion atmosphérique des polluants désigne l'évolution spatiale et temporelle d'espèces (gaz, particules solides/liquides) rejetées dans l'atmosphère, transportées et dispersées dans un environnement complexe sous l'effet de l'écoulement de l'air ambiant. Comme l'a récemment mis en évidence l'accident de l'usine Lubrizol à Rouen en 2019, maîtriser les rejets accidentels de substances dangereuses constitue un enjeu majeur pour les pouvoirs publics. Avec la présence de sites industriels à haut risque à proximité de zones urbaines densément peuplées, les conséquences sanitaires et environnementales peuvent être potentiellement sévères[1] [Brunekreef and Holgate, 2002]. Les effets toxiques sont particulièrement centraux pour la prévention des risques technologiques (par exemple, la directive Seveso-3 et la loi "RISQUES" adoptée à la suite de l'accident industriel d'AZF à Toulouse en 2001).

Le suivi des seuils de pollution dans un environnement urbain ou sur un site industriel constitue un véritable défi en raison de la nature complexe des écoulements urbains [Britter and Hanna, 2003; Dauxois et al., 2021]. À la micro-échelle urbaine, l'écoulement atmosphérique est fortement turbulent et couvre une large gamme d'échelles spatio-temporelles allant de la circulation météorologique de grande échelle jusqu'aux échelles de Kolmogorov. En particulier, une forte turbulence est engendrée par les fortes interactions entre l'écoulement du vent de surface et la topographie urbaine composée d'obstacles de nature, taille et forme variables (bâtiments, arbres, etc.). Par exemple, l'organisation des rues a un impact non-négligeable sur la forme et la concentration du panache toxique puisque les bâtiments perturbent l'écoulement atmosphérique pour donner naissance à des structures complexes de dispersion telles que des gradients de pression négatifs ou des flux de cisaillement, conduisant à des phénomènes locaux de séparation et de recirculation. Pour obtenir une représentation précise de ces structures d'écoulement et de la variabilité de la concentration qui en résulte à l'échelle d'une rue ou d'un quartier, des approches de modélisation haute-fidélité basées sur la mécanique des fluides numérique (CFD pour *Computational Fluid Dynamics* en anglais) sont nécessaires. Elles constituent une approche complémentaire aux expériences de terrain pour mieux comprendre les écoulements urbains et la dispersion atmosphérique à micro-échelle [Franke et al., 2011]. En effet, les expériences de terrain ne peuvent couvrir qu'un nombre restreint de scénarios en raison de coûts élevés de mise en œuvre. De plus, elles ne fournissent qu'une vision partielle, quoique précise, d'un scénario donné car les capteurs prennent généralement des mesures ponctuelles et ne fournissent pas de cartes complètes des quantités d'intérêt (même si les progrès récents sur les drones ouvrent la possibilité d'améliorer la couverture et la résolution des données acquises).

Le principe de la mécanique des fluides numérique est de résoudre numériquement les équations fondamentales de la dynamique des fluides (c'est-à-dire les équations de Navier-Stokes pour la dynamique de l'écoulement de l'air et l'équation de transport d'un scalaire pour la dispersion du traceur), tout en étant capable de traiter une géométrie complexe [Tominaga and Stathopoulos, 2013]. La mécanique des fluides numérique pour les écoulements urbains englobe

[1] `https://www.ecologie.gouv.fr/risques-technologiques-directive-seveso-et-loi-risques`

une grande variété d'approches qui diffèrent par leur manière d'appréhender l'effet des grandes échelles turbulentes sur l'écoulement moyen et le transport du traceur. Le choix du modèle de turbulence résulte d'un compromis entre complexité, précision et coût de calcul. D'une part, les approches de type moyenne de Reynolds (ou RANS pour *Reynolds-averaged Navier-Stokes* en anglais) modélisent entièrement la turbulence et constituent une option basse fidélité. D'autre part, les simulations aux grandes échelles (ou LES pour *large-eddy simulation* en anglais) considèrent les équations de Navier-Stokes filtrées: elles résolvent explicitement les grandes échelles de turbulence, tandis que l'impact des plus petits tourbillons sur les quantités d'intérêt est modélisé par un modèle de turbulence sous-maille. Les simulations LES sont reconnues pour fournir des solutions plus haute-fidélité que les simulations RANS pour les écoulements fortement instationnaires et anisotropes, que l'on trouve typiquement dans le sillage d'obstacles urbains [Blocken, 2018; García-Sanchez et al., 2018]. Elles donnent également accès à des informations plus détaillées sur l'écoulement et la concentration de traceur (y compris des statistiques de second ordre telles que les flux de masse) que les approches RANS (qui ne fournissent que des quantités de premier ordre moyennées en temps). Cependant, la précision, en général accrue des simulations LES par rapport aux simulations RANS, s'accompagne d'un coût de calcul beaucoup plus important.

Pour garantir la capacité de prévision à grande échelle des approches LES et RANS dans le contexte de dispersion micro-échelle, il est nécessaire de mettre en œuvre un cadre probabiliste pour la mécanique des fluides numérique, dans lequel un ensemble de simulations peut être réalisé pour refléter les incertitudes inhérentes à un événement donné. Ces incertitudes sont en partie dues à la variabilité naturelle des écoulements de couche limite atmosphérique [Dauxois et al., 2021], qui rend difficile la caractérisation des conditions aux limites et le fonctionnement des modèles de mécanique des fluides numérique. Dans le contexte accidentel, des incertitudes sont également liées au manque d'informations disponibles sur la source d'émission du polluant. Cependant, l'échantillonnage de ces incertitudes requiert la génération d'un grand ensemble de données de simulations, dans lequel chaque simulation représente un scénario probable différent (par exemple, différentes conditions atmosphériques – vitesse du vent, stabilité – et sources d'émission – position, débit, espèces). La génération d'un grand ensemble de données peut être hors de portée pour les cas de dispersion à grande échelle, en particulier pour les simulations LES qui sont beaucoup plus coûteuses que les simulations RANS. Il est donc nécessaire de concevoir un modèle simplifié, également appelé modèle d'ordre réduit, qui soit capable de restituer les processus physiques essentiels identifiés dans les simulations RANS ou LES, tout en permettant une prévision rapide des scénarios possibles durant un évènement. La prise en compte des incertitudes dans les modèles physiques d'ordre réduit est citée comme l'un des trois défis pour les simulations d'écoulement urbain dans l'étude de Dauxois et al. [2021]. C'est l'objectif principal poursuivi dans ce travail de thèse.

Les approches d'apprentissage automatique et profond offrent une piste de recherche intéressante pour représenter de façon efficace et précise la réponse des modèles physiques aux variations des paramètres d'entrée incertains. Cette réponse peut être complexe en raison des

non-linéarités présentes dans les données RANS ou LES dans les applications d'écoulement urbain à micro-échelle. Les modèles orientés données n'intègrent pas d'hypothèses physiques, mais au contraire apprennent des structures statistiques extraites de grands volumes de données. Ces modèles d'apprentissage sont des fonctions d'approximation très flexibles, capables de s'adapter à des réponses complexes non-linéaires. Cette flexibilité provient du grand nombre de paramètres (les poids du modèle) à optimiser à partir des données disponibles. Une fois que les poids du modèle sont calibrés, un modèle d'apprentissage peut être considéré comme une fonction analytique rapide à évaluer, qui peut produire des prévisions en temps réel des quantités d'intérêt. Malgré leur grand potentiel, la mise en œuvre de ces approches d'apprentissage pour la dispersion micro-échelle n'est pas évidente et se heurte à un certain nombre de problématiques : *i)* de multiples sources d'incertitudes sont en jeu ; *ii)* les grandes valeurs du nombre de Reynolds caractéristiques des écoulements urbains requièrent des maillages fins pour les modèles de mécanique des fluides numérique, ce qui augmente de manière considérable la dimension des statistiques de sortie à apprendre par les modèles orientés données ; et *iii)* le lien entre les entrées incertaines et les statistiques de sortie est fortement non-linéaire en raison des interactions entre le vent et les obstacles dans la canopée urbaine. Un tel problème nécessiterait un très grand volume de données pour bien explorer l'espace d'incertitudes. Cependant, comme les approches de mécanique des fluides numérique sont coûteuses, l'apprentissage de modèles réduits basés sur les données ne peut se faire que sous la contrainte de données disponibles parcimonieuses.

La littérature sur le couplage entre apprentissage automatique et mécanique des fluides numérique pour les écoulements urbains et la dispersion à micro-échelle est en plein essor. Toutefois, de notre point de vue, un certain nombre de questions restent ouvertes afin d'identifier les approches statistiques les plus pertinentes pour construire des modèles réduits capables de faire des prédictions à la fois robustes et physiquement cohérentes d'un évènement donné. Les études déjà parues dans la littérature sont difficilement comparables par manque d'homogénéité dans le choix du cas d'étude (topographie du terrain, sources d'incertitude, etc.), dans l'approche de modélisation utilisée pour générer la base de données d'entraînement, ou dans le choix des métriques de performance. Par la difficulté à appréhender l'ensemble des disciplines, elles ne traitent qu'une fraction des difficultés inhérentes à l'apprentissage automatique pour la dispersion à la micro-échelle urbaine, à savoir : le grand nombre d'incertitudes en entrée, la grande dimension des quantités d'intérêt en sortie, l'explicabilité du modèle réduit, et la cohérence physique des prévisions du modèle réduit. Par exemple, García-Sánchez et al. [2017] ont élaboré une expansion en chaos polynomial (i.e. une approche par régression polynomiale) à partir de données issues de 729 simulations RANS pour relier les incertitudes sur le profil du vent en entrée au champ de concentration moyen du traceur dans le centre-ville d'Oklahoma City. Cette étude traite des questions de non-linéarité et d'incertitudes, mais n'aborde pas la haute dimension des sorties (un modèle d'apprentissage est formé pour chaque nœud de maillage dans le domaine de calcul). Sur la base du même cas test du centre-ville d'Oklahoma City, Margheri and Sagaut [2016] ont mis en œuvre une approche par krigeage basée sur la décomposition ANOVA pour isoler la dépendance des sorties aux entrées incertaines et sur la décomposition orthogonale propre (ou POD pour *proper orthogonal decomposition* en anglais) afin de réduire la dimension des

sorties d'un modèle physique de type Lattice Boltzmann. Leur méthodologie est plus avancée que l'étude de García-Sánchez et al. [2017] mais les résultats sont difficilement comparables.

Objectifs de la thèse

Dans cette thèse, nous explorons et comparons une diversité d'approches de modélisation orientée données, gravitant autour de l'apprentissage automatique et profond, afin de mettre en œuvre un modèle réduit informé par les simulations LES pour produire des prévisions de concentration physiquement consistantes, tout en diminuant substantiellement leur coût de calcul. Cette étude porte sur un cas bidimensionnelle de dispersion d'un traceur dans un écoulement turbulent de couche limite atmosphérique autour d'un obstacle isolé, dans lequel les conditions limites d'entrée et la position de la source d'émission sont incertaines. Il convient de mentionner que dans ce travail, les quantités d'intérêt sont des statistiques de champ où la dimension temporelle n'est pas considérée et où l'objectif est d'émuler la variabilité spatiale des quantités d'intérêt. De plus, nous ne considérons que des entrées incertaines scalaires. Néanmoins, ce cas canonique permet une analyse approfondie de la précision et de la robustesse des modèles réduits vis-à-vis de trois problématiques énoncées ci-dessous :

- **Réduction de dimension.** Comment réduire la dimension des statistiques de champs LES d'intérêt pour identifier un espace latent de petite taille qui minimise la perte d'information ?
- **Non-linéarité.** Quels modèles réduits fournissent le cadre le plus approprié et le plus flexible pour construire un processus d'apprentissage capable de représenter les différentes échelles portées par les variables de l'espace latent ?
- **Base d'apprentissage réduite.** Comment mettre en œuvre un modèle réduit capable de prévoir des statistiques de champs LES physiquement cohérentes, en particulier dans le sillage de l'obstacle et dans les zones de recirculation, alors que la base de données d'entraînement est limitée ?

Plan du manuscrit

Ce manuscrit est divisé en cinq chapitres. Afin de bien positionner ce travail de thèse par rapport à la littérature, le chapitre I présente une introduction aux processus de dispersion en milieu urbain à micro-échelle, aux approches de modélisation associées et aux opportunités offertes par les méthodes d'apprentissage automatique et profond pour la mécanique des fluides numérique. Le chapitre II présente l'approche par modèle réduit proposée dans ce travail, incluant une composante de réduction de la dimension et une composante de régression. L'étude de cas bidimensionnel ainsi que le choix des incertitudes et des outils de modélisation physiques et statistiques utilisés dans ce travail sont présentés au chapitre III. Le chapitre IV examine une série de modèles réduits par apprentissage entraînés à partir de données LES, et identifie la meilleure approche pour émuler les champs de concentration de traceur moyennés en temps. Le chapitre V présente et évalue les performances d'un modèle réduit hybride pour pallier les limitations des modèles orientés données discutés au chapitre IV. Le nouveau modèle réduit

permet d'inclure des contraintes physiques dans le processus d'apprentissage et donc dans les prévisions de concentration de traceur par le biais d'une équation RANS de transport de scalaire informée par la LES. Cette approche ouvre la voie à la construction de modèles réduits entraînés sur des données de type multi-fidélité.

General introduction

Pollutant atmospheric dispersion refers to the spatial and temporal evolution of species (gas, solid/liquid particles) released into the atmosphere, transported and dispersed in a complex environment by the effect of the atmospheric airflow. As highlighted by the recent Lubrizol industrial accident in Rouen in 2019, monitoring the accidental release of hazardous substances is a major issue for public authorities with the presence of many high-risk industrial facilities near densely-populated urban areas and the immediate severe consequences on human health and environment[2] [Brunekreef and Holgate, 2002]. Toxic effects are particularly central to the prevention of technological risks (e.g. Seveso-3 directive and the "RISQUES" law that was enacted following the AZF industrial accident in Toulouse in 2001).

Tracking the pollutant concentration in an urban environment or on an industrial site remains a challenge due to the complexity of urban flows [Britter and Hanna, 2003; Dauxois et al., 2021]. At the urban microscale, the atmospheric flow is highly turbulent and covers a broad range of spatio-temporal scales ranging from the large-scale weather circulation to Kolmogorov scales. In particular, strong turbulence is induced by the complex interactions between near-surface wind flow and urban topography made of obstacles of varying nature, size and shape (e.g. buildings, trees). For instance, the street layout has a significant impact on the shape and concentration of the toxic plume since buildings disrupt the atmospheric flow to give rise to complex patterns such as adverse pressure gradients or shear flows, leading to local separation and recirculation phenomena. To obtain an accurate representation of these complex flow patterns and of the subsequent pollutant concentration variability at the scale of a street or a neighbourhood, high-fidelity modelling approaches based on computational fluid dynamics (CFD) are required. They offer a complementary approach to field experiments to provide further insights into microscale urban flow and atmospheric dispersion [Franke et al., 2011]. Indeed, field experiments can only cover a limited number of urban flow scenarios because of their high setup costs and can only provide a partial but accurate view of a given scenario, as sensors usually take point-wise measurements and do not yet provide complete maps of the quantities of interest (even though recent progress on UAVs is promising to improve the coverage and resolution of the acquired data).

CFD numerically solves for the fundamental equations of fluid dynamics (i.e. Navier-Stokes equations for air flow dynamics and a scalar transport equation for tracer dispersion) while being able to deal with complex geometry [Tominaga and Stathopoulos, 2013]. Urban flow CFD encompasses a variety of approaches that differ in the way they represent the effect of large-range turbulence scales on the mean flow and tracer transport. The choice of the turbulence model results from a trade-off between complexity, accuracy and computational cost. On the one hand, Reynolds-averaged Navier-Stokes (RANS) simulation approaches fully model turbulence and offer a low-fidelity option. On the other hand, large-eddy simulations (LES) solve for the filtered Navier-Stokes equations: they explicitly solve for the large turbulence scales, while

[2] `https://www.ecologie.gouv.fr/risques-technologiques-directive-seveso-et-loi-risques`

the impact of the smallest eddies on the quantities of interest are modelled by a subgrid-scale turbulence model. LES are known to provide more high-fidelity solutions than RANS for highly unsteady flows with strong anisotropy, typically found in the wake of urban obstacles [Blocken, 2018; García-Sanchez et al., 2018]. They also give access to more detailed flow and tracer concentration information (including second-order statistics such as mass fluxes) than RANS approaches (which only deliver first-order time-averaged quantities). However, this increased accuracy of LES over RANS in certain situations comes with a much higher computational cost.

To ensure full-scale predictive capability of LES and RANS approaches in the context of microscale pollutant dispersion, it is necessary to design a probabilistic CFD framework where an ensemble of simulations can be carried out to account for the uncertainties involved in a given event. These uncertainties are partly due to the inherent variability of the atmospheric boundary-layer flow [Dauxois et al., 2021], which makes it difficult to characterise the boundary and operating conditions that are necessary to configure CFD models for a given case study. They are also due to the limited information available on the pollutant emission source in the accidental context. However, sampling these uncertainties requires the generation of a large dataset of CFD simulations, in which each simulation represents a different possible scenario (e.g. different atmospheric conditions – wind velocity, stability – and emission sources – position, flow rate, species). Generating a large dataset may be out of reach for large-scale dispersion cases, in particular for LES that are much more computationally intensive than RANS. There is therefore a need to design a simplified model, also referred to as a reduced-order model, that is able to reproduce key physical processes learned from RANS or LES simulations, while allowing for quick predictions of possible scenarios during an event. "Accounting for uncertainty in reduced-order physics models" is cited as one of the three challenges for urban flow simulations in Dauxois et al. [2021]. This is the main objective pursued in this PhD thesis work.

Machine and deep learning approaches are an interesting avenue for accurately and efficiently representing the response of physics-based models to variations in uncertain inputs. Such mapping can be complex due to the nonlinearities present in the RANS or LES data in microscale urban flow applications. Data-driven models based on machine learning do not incorporate physical assumptions but instead learn statistical patterns extracted from large volumes of data. These learning models stand as strongly flexible approximation functions that are able to fit nonlinear complex mappings. This flexibility comes from the large number of parameters (the weights) to be optimised based on the available data. Once its model weights are calibrated, a learning model can be seen as a fast-evaluating analytical function that can produce real-time predictions of the quantities of interest. Despite its great potential, the implementation of learning approaches for microscale dispersion problems is not straightforward and faces a number of issues: *i)* there are multiple sources of uncertainty at play; *ii)* the large Reynolds number of the urban flows requires fine CFD meshes that dramatically increase the dimension of the output statistics to be learned from the data-driven models; and *iii)* the mapping between uncertain inputs and output statistics is strongly nonlinear due to wind-building interactions in the urban

canopy. Such a problem would require a large amount of data to densely explore the uncertainty space. However, because CFD is computationally expensive, learning data-driven reduced-order models can only be done along with sparse data.

The literature on this coupling between machine learning and CFD models for microscale urban flows and dispersion is emerging, but from our point of view, there are still many points to study in order to identify the most suitable approaches to building a robust reduced-order modelling approach that produces physically consistent predictions of a given event. Studies already published in the literature are difficult to compare due to lack of homogeneity in the choice of the case study (e.g. terrain topography, sources of uncertainty), in the modelling approach used for generating the training database, or in the choice of the performance metrics. Necessarily, they only address a fraction of all machine learning issues for microscale urban flow and dispersion, namely: the large number of input uncertainties, the high dimension of the output quantities of interest, the reduced-order model explicability, and the physical consistency of the data-driven model predictions. For instance, García-Sánchez et al. [2017] designed a polynomial chaos expansion (i.e. a polynomial regression approach) to map uncertainties on the inlet wind profile to the mean tracer concentration field in downtown Oklahoma City using a training database made of 729 RANS snapshots. This study deals with the issues of nonlinearity and uncertainty but does not address the high-dimensionality of the outputs (one learning model is trained for each mesh node of the computational domain). Based on the same test-case of downtown Oklahoma City, Margheri and Sagaut [2016] designed a kriging approach based on ANOVA decomposition to isolate the outputs' dependency on each uncertain input and on proper orthogonal decomposition (POD) to reduce the output dimension for a Lattice Boltzmann physics-based model. Such a framework is more advanced compared to the study of García-Sánchez et al. [2017] but the results are difficult to compare.

Objectives of the thesis

In this thesis, we explore and compare a variety of data-driven modelling approaches based on machine and deep learning to design a reduced-order model informed by LES to produce physically consistent concentration predictions while substantially decreasing their computational cost. This study is carried out on a two-dimensional tracer dispersion case in a turbulent atmospheric boundary-layer flow over an isolated obstacle, in which both the inflow boundary condition and source location are uncertain. It is worth mentioning that in this work, the quantities of interest are field statistics, implying that the temporal dimension is not considered and that we focus on emulating the spatial variability of the quantities of interest. Moreover, we only consider uncertain inputs that are scalars. Still, this canonical case allows for an in-depth analysis of the reduced-order model's accuracy and robustness with respect to the following three issues:

- **Dimension reduction.** How to properly reduce the dimension of the LES field statistics of interest to identify a small latent space that minimises information loss?
- **Nonlinearity.** Which reduced-order modelling approach provides the most appropriate and

flexible framework for fitting the learning process to the different length-scales of the latent variables?

– **Limited training database.** How to design a reduced-order modelling approach that is able to produce physically consistent LES field statistics predictions, in particular in the wake of the obstacle and in the recirculation regions, while the training database is limited?

Manuscript outline

The manuscript is divided into five chapters. Chapter I provides an introduction to microscale urban dispersion processes, related modelling approaches and the possibilities offered by machine learning and deep learning for CFD, in order to properly position this PhD thesis work in relation to the literature. Chapter II presents the reduced-order modelling approach proposed in this work, including a dimensionality reduction component and a regression model component. The two-dimensional case study, along with the choice of the uncertainties and the physical/statistical modelling tools used in this work, is presented in Chapter III. Chapter IV compares a variety of reduced-order modelling approaches directly obtained from LES data and identifies the best approach to emulate the time-averaged tracer concentration fields. Chapter V presents and evaluates the performance of a hybrid reduced-order modelling approach to overcome the purely data-driven model limitations observed in Chapter IV. This new reduced-order model allows for the introduction of physical constraints in the learning process and thereby in the tracer concentration prediction through the RANS scalar transport equation informed by LES data. This approach paves the way towards reduced-order modelling approaches based on multi-fidelity data.

Chapter I

Introduction

This first chapter provides an overview of the fundamental concepts related with the multidisciplinary subject of reduced-order modelling for microscale urban flow and tracer dispersion. More details are given on urban wind flow and dispersion processes in Sect. I.1, on the modelling approaches available for atmospheric dispersion with a focus on RANS and LES approaches in Sect. I.2, and on the state-of-the-art statistical learning applications for CFD problems in Sect. I.3. In particular, data science interconnections are discussed in order to study the opportunities and challenges arising from the use of machine and deep learning techniques combined with CFD. This allows us to position the framework of this PhD thesis and its objectives in Sect. I.4.

I.1 Physical processes in urban air dispersion

Urban atmospheric dispersion refers to the spatial and temporal evolution of species (aerosols, gases) released into the atmosphere, which are transported and dispersed in a complex urban-type environment. This urban environment is made of buildings, trees and other elements of varying size. These roughness elements impact the flow structures and the pollutant dispersion in the lowest layers of the atmosphere, known as the atmospheric boundary-layer, through fluxes of momentum, heat, water vapour, pollutant species, etc. The study of the interactions between the urban surface and the atmosphere at fine scales (≤ 1 km), including the diffusion and near-range pollutant transport, belongs to the field of micrometeorology.

Pollutant concentrations are difficult to track in urban areas due to the complex interactions between the pollutant plume, the atmospheric dynamics and the urban topography [Britter and Hanna, 2003; Belcher, 2005]. For instance, the street layout combined with the inflow wind direction has a significant impact on the plume shape and concentration as highlighted in Fig. I.1. Pollutant dispersion depends on pollutant physical properties (e.g. chemical composition, density, diffusivity), emission characteristics (continuous/intermittent, smooth release/high speed leak), atmospheric conditions (e.g. wind field, temperature, stability condition), mutual pollutant-atmospheric interactions (e.g. chemical reactions) as well as urban topography (nature of the ground, obstacles).

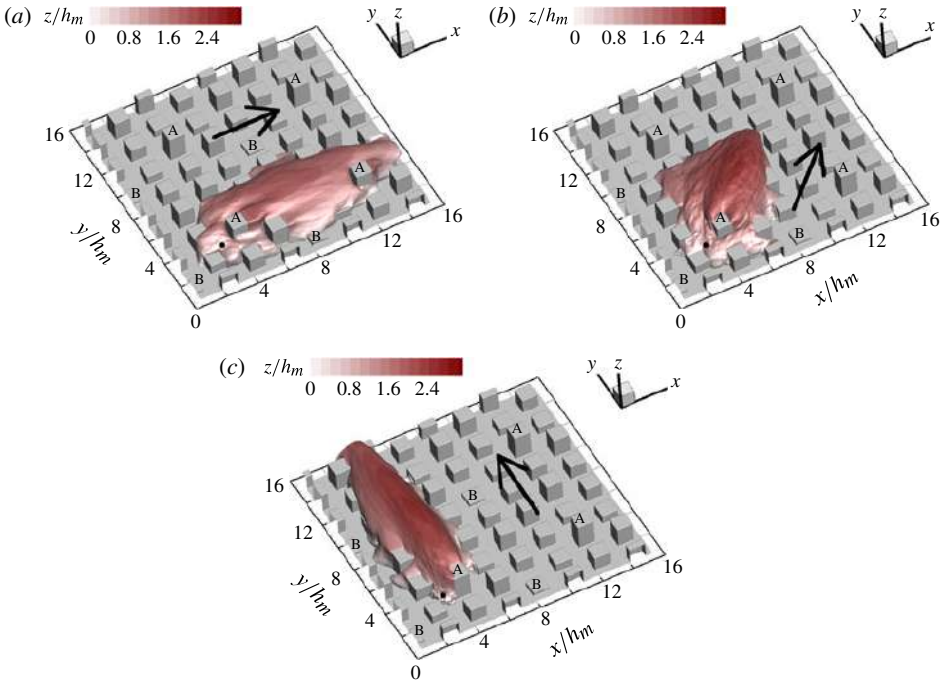

Figure I.1: Isosurface of mean scalar concentration shaded by height due to a point-source emission (black dot) obtained for three different flow directions (indicated by the black arrow) [Philips et al., 2013].

I.1.1 Pollutant physical and release properties

The pollutant plume characteristics depend on (1) the physical properties of the released species, (2) their chemical interactions with the surrounding air, and (3) the conditions under which they are released.

1. Chemical species can be in various physical states (e.g. soot is in solid form, carbon monoxide is in a gaseous state), which can affect the plume density and induce buoyancy effects (e.g. denser gases will tend to spread on the ground surface due to gravity).

2. Chemical reactions with the surrounding air might occur. The released products may be more or less stable, remain in their original state or undergo chemical transformations (e.g. reactions between sulphur dioxide – SO_2, nitrogen dioxide – NO_2, water, oxygen and other chemicals may lead to acid rains). These transformations may change the nature of the plume resulting in complex dispersion dynamics [Leelőssy et al., 2014].

3. The storage conditions, the type of release and the shape of the leaking opening also affect the pollutant initial temperature and momentum. Accidental emissions are usually associated with high initial pressure and tracer momentum.

Passive tracer assumption. All of these interactions might be challenging to represent. In many situations, assumptions on the pollutant properties could help simplify the modelling process. For instance, assuming the pollutant behaves as a passive tracer means that the tracer is nonreactive, does not induce density variation, and has no self momentum. This type of release occurs in the case of highly diluted stable gases and for emission conditions that are close to atmospheric conditions. In practice, this assumption is useful to decouple the flow dynamics

from the plume dynamics since passive tracer emission and dispersion have no feedback on the carrier flow.

I.1.2 Urban wind flow

At microscale, we are interested in relatively short time scales (of the order of a minute) and small spatial dimensions (i.e. at the scale of a building, a street or a district). Microscale flows result from multi-scale processes in the atmospheric boundary-layer: they depend on both large-scale (mesocale) meteorology governing long-term behaviour (>10 minutes, >10 kilometers), and small-scale processes like turbulence (< 1 minute, <100 meters) influenced by local urban topography and air flow dynamics.

Atmospheric-to-urban scales. The atmosphere is subject to a wide variety of phenomena, which are characterised by their own spatial and time scales (from jet stream at planetary scales to thunderstorms at mesoscale and tornadoes at microscale). Oke [2002] suggests a classification of observation scales reported in Table I.1, which is useful to define what is implied by the term "microscale". Plume dispersion generally occurs in the atmospheric boundary layer, whose depth can vary between 100 m and 3 km as it is directly influenced by the Earth's surface thermal effects and topography, resulting in large vertical wind shear, momentum, heat, and mass turbulent exchanges [Garratt, 1992; Oke et al., 2017]. The atmospheric boundary-layer over urbanised areas, named urban boundary layer (UBL), has a very specific structure. When it comes to dispersion over urban areas, it is typical to relate the different observation scales to the type and dimension of the obstacles. Accordingly, the plume dispersion evolves in specific atmospheric sublayers. For example, the mesoscale focuses on the dispersion up to the city scale (Fig. I.2c) in the UBL. Closer to the ground surface, the local scale relates to the dispersion in a street or a neighbourhood (Fig. I.2b) in the surface layer. The microscale relates to the dispersion around a few buildings (Fig. I.2c) in the lowest part of the UBL called the roughness sublayer, which is highly impacted by the roughness elements.

Table I.1: Atmospheric scales of motion adapted from Oke [2002].

Global/Macroscale	from 10^2 up to 10^5 km	weeks, centuries	polar jet stream
Mesoscale	from 10^1 up to 10^2 km	minutes, days	thunderstorm
Local scale	from 10^{-1} up to 10^1 km	seconds, minutes	large cumulus, tornado
Microscale	from 10^{-5} up to 10^0 km	seconds, minutes	small cumulus, dust-devil

Atmospheric turbulence and impact on dispersion. In reality, the different atmospheric scales form a continuum, each scale being influenced by the larger and smaller scales. Urban atmospheric conditions are controlled by global meteorological events, occurring on longer time scales, which drive the background flow conditions. Turbulence, by its multiscale and nonlinear nature, directly contributes to the coupling of the different scales: it can transfer energy back and forth across scales, from large atmospheric turbulence eddies (of several kilometres in diameter) to scales smaller than urban topology down to Kolmogorov dissipative scales (0.1-10 millimetres) where molecular mixing comes into play.

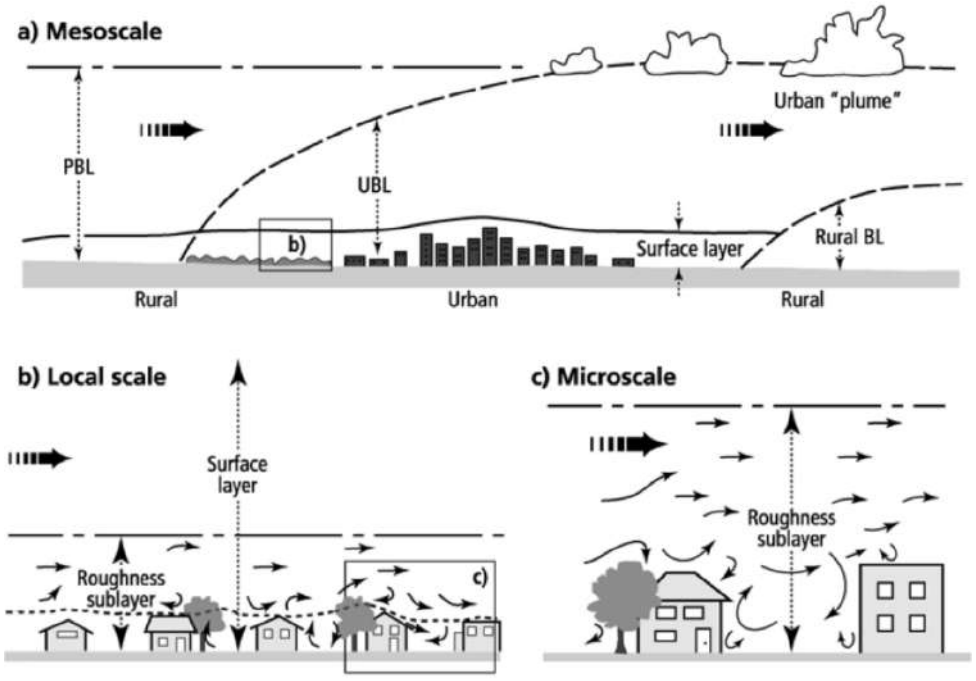

Figure I.2: Schematic of the atmospheric boundary-layer over an urban area. Three overlapping scales of observation, (a) mesoscale, (b) local scale, and (c) microscale, are represented according to the type and size of the urban obstacles (revised by Oke and Rotach after a figure in [Oke, 1997]).

Eddies of different sizes have a different impact on the dispersion patterns as illustrated in Fig. I.3. Turbulent eddies significantly smaller than the plume leads to micro-mixing, which essentially spreads the plume (Fig. I.3a). Eddies much larger than the plume structure advects the plume following the large-scale turbulent motion, without altering its internal structure (Fig. I.3b). Turbulent structures with a size comparable to the plume alter its shape and expand its contour (Fig. I.3c).

At microscale, all these processes occur simultaneously. The flow variability can be distinguished into (1) large-scale fluctuations induced by large-scale forcing variability (e.g. geostrophic wind) down to mesoscale; and (2) small-scale fluctuations related to turbulence production, primarily induced by buoyancy, wind shear and interaction with the obstacles in the surface layer, down to microscale. In microscale studies, turbulence refers to the second flow variability type. It is worth noting that large-scale variability is often not embedded in microscale studies, it is rather seen as an external source of variability.

Buoyancy effects. Thermal effects can play a major role in the generation of turbulent flow structures through buoyancy. For instance, the energy balance is altered by the city thermal properties. Ground materials (steel, cement, asphalt, etc.) increase the absorption of solar radiation and limit evaporation, raising the local air temperature [Couillet, 2002]. The ground ability to absorb and release heat via radiation can have a substantial impact on the vertical temperature distribution. If the ground heats up significantly quicker than air due to radiation, it can lead to an unstable atmosphere where buoyancy contributes to turbulence production. In contrast, a clear night with low wind speeds corresponds to a stable atmosphere where turbulence is essentially mechanical.

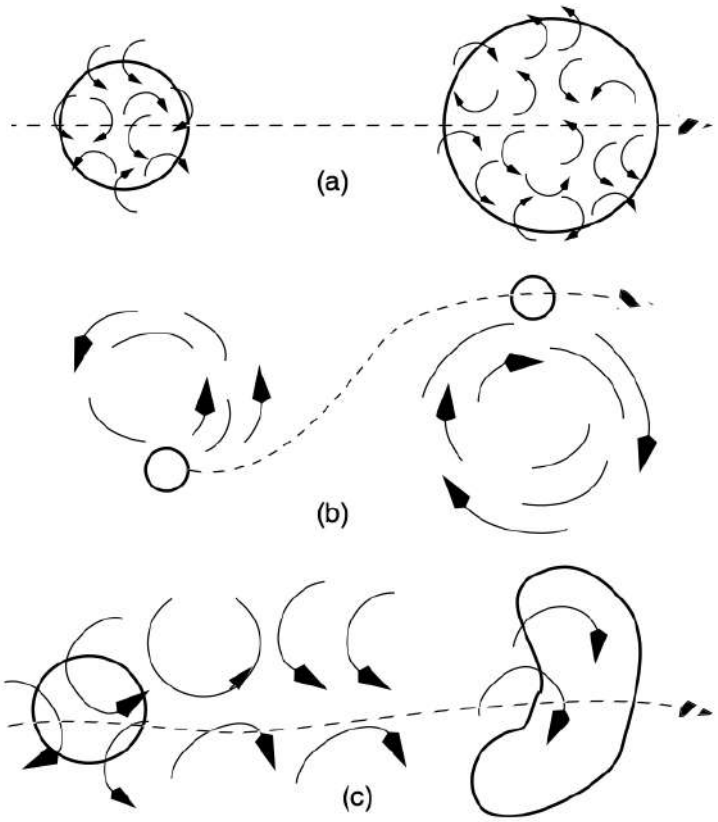

Figure I.3: Dispersion of scalar concentration under turbulent eddies of size (a) smaller, (b) larger and (c) comparable to the characteristic size of the plume [Seinfeld, 1986].

Atmospheric stability can be characterised by several criteria, e.g. the Richardson number [Richardson, 1921], the Monin-Obukhov theory [Obukhov, 1971], and the Pasquill classification [Pasquill, 1961].

– The Richardson number Ri measures the ratio between thermal and mechanical production of turbulent kinetic energy by comparing potential temperature (θ) gradient (characterising buoyancy) to wind shear

$$Ri = \frac{g}{T} \left(\frac{\frac{\partial \theta}{\partial z}}{\left(\frac{\partial \boldsymbol{u}}{\partial z}\right)^2 + \left(\frac{\partial \boldsymbol{v}}{\partial z}\right)^2} \right) , \tag{I.1}$$

where g is the gravity constant, T is the temperature, u and v are the horizontal wind velocity components, and z is the vertical coordinate. The sign of the Richardson number highlights the role of buoyancy: $Ri < 0$ indicates unstable conditions as there is turbulent production by buoyancy that becomes dominant for very large negative values; buoyancy has a stabilising effect for $Ri > 0$; and neutral conditions correspond to $Ri \approx 0$.

– The Monin-Obukhov length L_{MO} represents the height at which turbulence production by buoyancy becomes dominant compared to wind shear. It is expressed as:

$$L_{MO} = -\frac{u_\tau^3 \, \rho \, C_p \, T}{g \, k \, q}, \tag{I.2}$$

where u_τ is the friction velocity, ρ is the air density, C_p is the heat capacity, k is the von

Kármán constant, and q is the sensible heat flux. Neutral conditions are characterised by an infinite L_{MO}-ength, stable conditions by positive values and unstable conditions by negative values.

– The Pasquill classification discretizes atmospheric stability in six classes ranging from the most unstable condition (A) to the most stable condition (F) based on 10-m wind speed, incident solar radiation and total cloud cover.

Interaction with urban topography. Within the urban canopy, terrain complexity is described *i*) by obstacles of a size comparable to the observation scale, and *ii*) by the roughness elements, which correspond to small obstacles that act uniformly on the flow [Wiernga, 1993; Grimmond and Oke, 1999]. For instance, when observing the flow at the street scale, the paved road irregularities can be modelled on average by a roughness length, which represents the height at which the flow speed virtually becomes zero in the logarithmic profiles of the surface boundary layer. Typical roughness lengths for terrain irregularities are presented in Table I.2.

Table I.2: Typical surface roughness lengths for different types of terrain.

Type of terrain	Roughness length (m)
Sand	$10^{-4} - 10^{-3}$
Sea surface	5×10^{3}
Grass	$10^{-2} - 10^{-1}$
Forest and woodland	$10^{-1} - 10^{0}$
Suburban areas	$1 - 2$
City	$1 - 4$

The presence of obstacles disrupts the flow, generating counter gradient diffusion effects and vortex profiles around the obstacles and in its wake, resulting in local separation and recirculation regions. Figure I.4 illustrates the typical flow patterns observed around an isolated wall-mounted obstacle [Li and Meroney, 1983a; Martinuzzi and Tropea, 1993; Li and Stathopoulos, 1997; Tominaga et al., 1997; Meroney et al., 1999; Blocken et al., 2008a; Tominaga and Stathopoulos, 2008]. In this configuration, a certain amount of pollutant is periodically trapped and spread in the wake of the obstacle through vortex shedding [Louka et al., 2000].

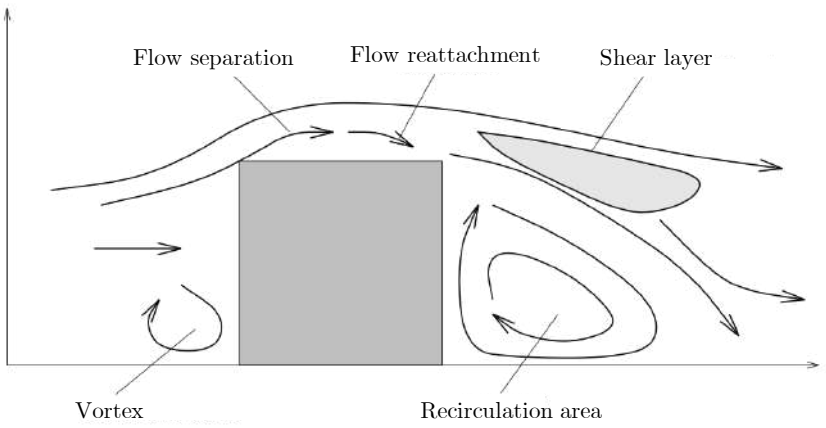

Figure I.4: Flow vertical cross-section around an isolated wall-mounted obstacle [Turbelin, 2000].

Within the urban canopy made by an ensemble of obstacles, the atmospheric flow perturbation depends on the distance between obstacles and on their individual heights. This has been extensively studied through canonical multi-obstacle configurations such as idealised street canyon [Leitl and Meroney, 1997; Chan et al., 2002; Baik and Kim, 2002; Kim and Baik, 2004; Gromke et al., 2008; Garbero, 2008] and idealised urban areas [Milliez and Carissimo, 2007; Philips et al., 2013; Carpentieri and Robins, 2015]. Based on the aerodynamic interactions between buildings, three distinct flow regimes illustrated in Fig. I.5 have been identified [Oke, 1988]. These regimes can be distinguished based on the obstacle aspect ratio H/W, where W corresponds to the typical distance between two obstacles and H corresponds to their typical height H.

- Lower ratio values, i.e. $H/W \in [0.15, 0.2]$, are associated with the isolated roughness regime (Fig. I.5a). Since the obstacles are widely separated, the flow field is simply a superposition of flow fields obtained for isolated buildings (Fig. I.4). This implies that the recirculation regions upstream and downstream of each obstacle are independent from the other obstacles.

- Intermediate ratio values, i.e. $0.2 < H/W < 0.65$, are associated with the wake interference regime (Fig. I.5b). When two buildings are sufficiently close each other, the resulting flow pattern becomes more complex: the wakes induced by one obstacle upstream interact with the downstream obstacles.

- Higher ratio levels, i.e. $H/W > 0.65$, correspond to the skimming flow regime in which the obstacles are densely packed. In this case, the flow dynamics in-between the obstacles is local (vortex frequency is directly influenced by the obstacle aspect ratio) and is relatively decoupled from the flow dynamics above the urban canopy.

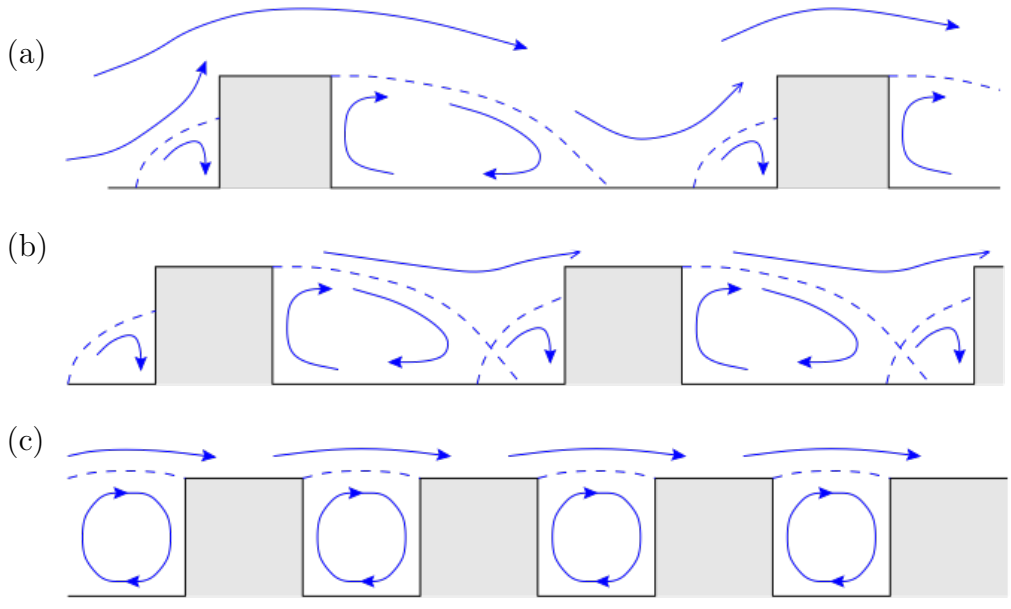

Figure I.5: Flow regimes for different building layout. (a) Isolated roughness regime. (b) Wake interference regime. (c) Skimming flow regime [Garbero, 2008].

In this work, we limit our study to the isolated roughness regime under neutral atmospheric conditions, implying that there are no buoyancy effects on the flow and plume dynamics. The tracer is assumed to be passive. Consequently, the plume behaviour is mainly driven by mechanical effects, i.e. by turbulent production from mean shear and interaction of the wind flow with an isolated obstacle. Our problem complexity is essentially linked to the consideration of uncertainties, and on their impact on the flow dynamics and tracer dispersion around the obstacle.

I.2 Modelling approaches for urban air dispersion

This section provides an overview of the main modelling approaches used for atmospheric dispersion and discusses their complementarity with the measurements (Fig. I.6). These models are not universal because they are not adapted to the same scales of observation. They do not feature the same level of accuracy, the same setup complexity or the same computational cost.

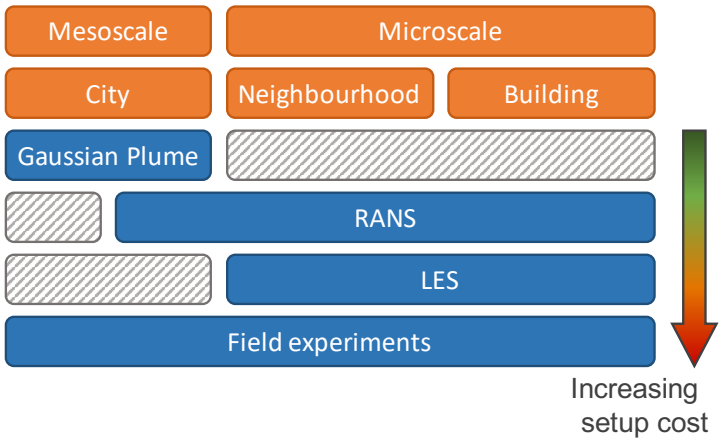

Figure I.6: Overview of available modelling approaches for pollutant dispersion according to the atmospheric scales of observation. Adapted from Philips [2012].

I.2.1 Advantages and limitations of experiments

I.2.1.a Field-scale experiments

On-site experiments provide information on the full complexity of dispersion processes in the atmospheric boundary-layer. They have been used to improve our understanding of the plume dynamics, first conducted in cities in the USA (Salt Lake City, Utah – Allwine et al., 2002; Oklahoma City Joint Urban 2003 Experiment, Oklahoma – Allwine et al., 2004; Allwine and Flaherty, 2006) or in idealised urban-like canopy (MUST/Mock Urban Setting Test in the desert of Utah – Biltoft, 2001). However, these experiments are not necessarily generalisable due to the wide variety of urban geometry and climate conditions that are present in the cities across the world. For this reason, many studies have been carried out, particularly in Europe, to investigate the impact of site specificity on dispersion: e.g. in Basel (Switzerland) [Rotach et al., 2004], or in Birmingham and London (UK) [Britter et al., 2002; Arnold et al., 2004; Dobre et al., 2005; Martin et al., 2008].

Still, large-scale field experiments remain expensive, time-consuming and only give access to sparse data on the flow and plume dynamics because of experimental limitations. It would require too many sensors and long acquisition times under a variety of weather conditions to provide a complete view of the possible plume dispersion scenarios. In this context, on-site measurements usually collect data at the street level but more rarely study vertical dispersion issues (e.g. tracer loss due to dispersion in the boundary-layer above roof level).

I.2.1.b Laboratory-scale experiments

In complement to field experiments, more controlled wind-tunnel experiments (usually carried out with a working scale of the order of 1:50 to 1:500) have been carried out to provide insights into the fundamental processes involved in air pollution dispersion [Saathof et al., 1995; Saathoff et al., 1998; Meroney et al., 1999; Stathopoulos, 2002; CEDVAL, 2022]. Some of these experiments focus on single-building configurations such as isolated bluff obstacles [Li and Meroney, 1983a; Tominaga et al., 1997; Li and Stathopoulos, 1997; Meroney et al., 1999; Tominaga and Stathopoulos, 2008; Blocken et al., 2008a; Gamel, 2015]. These configurations are useful for understanding localised flow structures around an obstacle as illustrated in Fig. I.4 [Murakami, 1993; Vinçont et al., 2000]. It would be very difficult to access such fine flow details in large-scale experiments, which are valuable for model validation.

Laboratory experiments take advantage of the turbulent flow scalability. To properly reproduce atmospheric boundary-layer conditions in a wind tunnel, geometric similarity must be ensured by consistent scaling. Dynamic similarity, involving similar flow patterns, is derived from the dimensional analysis of the Navier-Stokes equations. In particular, the Reynolds number Re is a key dimensionless number for characterising the level of turbulence in a flow defined as:

$$Re = \frac{U\,L}{\nu}, \tag{I.3}$$

where U is a reference velocity, L is a characteristic length-scale, and ν is the fluid kinematic velocity. The Reynolds number must be identical between the reduced-scale representation of the urban boundary-layer and the actual flow conditions that one seeks to reproduce at small scale. Stated differently, a given atmospheric flow in neutral conditions may be represented by another flow with the same Reynolds number. In general, the non-dimensional boundary conditions must be also identical to impose the same velocity and turbulence intensity profiles in the wind-tunnel boundary layer as in the actual flow conditions.

These small-scale experiments are valuable for several reasons. First, laboratory settings are favourable to reproductibility since they provide control over the boundary conditions, allowing the collection of long time-series to obtain converged flow statistics. Second, they give access to high-quality validation data and in particular to complex flow pattern observations through advanced diagnostics that are not applicable to field experiments. For instance, Garbero et al. [2010] studied passive tracer dispersion inside an idealised urban canopy based on velocity measurements obtained using a Laser Doppler Anemometer (LDA) and Hot Wire Anemometry (HWA) as well as concentration measurements obtained using flame ionisation detectors (FID). The issue with these instruments is to provide adequate spatial resolution of the velocity field

so as to detect high frequency fluctuations in the turbulent flow and to properly estimate plume concentrations without altering the flow.

However, wind-tunnel experiments suffer from similarity issues because of their reduced scale. Some atmospheric conditions such as low wind conditions and stable/unstable atmospheric stratification are difficult to reproduce in wind tunnels [Blocken, 2014] as the exact Reynolds similarity is impossible to respect in most practical case studies. Moreover, such experiments are still too costly and time-consuming to assist the design of a new building or urban area. In this regard, modelling approaches such as CFD are attractive options to study a wide range of dispersion conditions and to give access to the spatio-temporal variability of the quantities of interest (wind flow velocity components, tracer concentration) [Blocken, 2018], which are useful for a deeper analysis of microscale atmospheric dispersion processes in urban environment.

I.2.2 Introduction to modelling approaches

I.2.2.a Parametric Gaussian models

To be compatible with multi-query and real-time assessment, simple models based on phenomenological considerations of air dispersion have been developed. One representative example is the Gaussian model [Sutton, 1947]. This model relies on an analytical expression for the tracer concentration, obtained as the solution of the advection-diffusion equation for simplified boundary conditions [Stockie, 2011]. In its most simple formulation, the tracer concentration can be expressed as:

$$K(x,y,z) = \frac{Q_s}{2\pi\, U\, \sigma_y\, \sigma_z} \exp\left(-\frac{(y-y_{\mathrm{src}})^2}{2\,\sigma_y^2} - \frac{(z-z_{\mathrm{src}})^2}{2\,\sigma_z^2}\right), \tag{I.4}$$

where U is the uniform velocity in the streamwise direction (x-direction), Q_s is the pollutant release rate, σ_y and σ_z are the spanwise and vertical plume spread parameters that depend on the distance from the emission source $(x_{\mathrm{src}}, z_{\mathrm{src}})$.

The main difficulty in setting up a Gaussian model is to define appropriate values for the spread parameters σ_y and σ_z, which are related to turbulent diffusivity. As the turbulent diffusivity itself is generally unknown, these parameters are often computed from experimental measurements and observations. In particular, they depend on the atmospheric stability condition, which can be defined based on Pasquill stability classes [Holmes and Morawska, 2006; Turner, 2020]. For instance, Pasquill-Turner correlations formulate the standard deviations σ_y and σ_z in the following general form:

$$\sigma = a\, x^b + c, \tag{I.5}$$

where the constants a, b and c are determined from large-scale field measurements in an open flat terrain, and their values depend on the Pasquill atmospheric stability class. In addition, the Gaussian model is based on several strong assumptions related to tracer and flow conditions: homogeneous flat terrain without obstacle, homogeneous turbulent diffusion, uniform wind field of at least 1 to 2 $\mathrm{m\,s^{-1}}$ with a constant vertical profile. More advanced models can take into

account wind speed fluctuation, air temperature, atmospheric density with height, etc. In practice, Gaussian models are widely used for their ease of implementation and their cost-effective integration. They are valid for representing the far field of the emission source but are not appropriate for microscale and isolated barrier studies [Bluett et al., 2002; Couillet, 2002].

I.2.2.b Computational fluid dynamics

CFD is based on the numerical resolution of partial differential equations, including a transport equation for the scalar and the Navier-Stokes equations to represent the airflow motions. For a typical atmospheric dispersion simulation, the user has to set up the computational domain geometry and mesh, the boundary conditions and wall treatment, the numerical schemes, etc. Additional parameters may also be added such as the turbulence model depending on the adopted CFD approach (see Sect. I.2.3 for a discussion on LES and RANS approaches).

For passive tracer dispersion under neutral conditions, the system of equations for incompressible flow includes the continuity equation (or mass conservation equation), the momentum conservation equation, and the advection-diffusion equation for tracer conservation (or tracer transport equation):

$$\begin{aligned} \frac{\partial u_i}{\partial x_i} &= 0, \\ \frac{\partial u_i}{\partial t} + \frac{\partial u_i\, u_j}{\partial x_j} &= -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left(2\,\nu\, s_{ij}\right), \\ \frac{\partial \mathbf{K}}{\partial t} + \frac{\partial \mathbf{K}\, u_j}{\partial x_j} &= \frac{\partial}{\partial x_j}\left(D\,\frac{\partial \mathbf{K}}{\partial x_j}\right), \end{aligned} \tag{I.6}$$

where u_i denotes the ith component of the instantaneous airflow velocity field, x_i is the spatial position, p is the instantaneous pressure, t is the time, ρ is the air density, ν is the kinematic molecular viscosity, $\mathbf{K}$ is the scalar concentration, D is the molecular diffusion coefficient, and s_{ij} the strain tensor expressed as:

$$s_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right). \tag{I.7}$$

This system of equations simulates the entire turbulence spectrum, meaning that all spatial and temporal scales are resolved down to the lowest dissipative Kolmogorov microscales. This is referred to as direct numerical simulation (DNS). In this framework, the required number of mesh points N_h grows with respect to the Reynolds number (Eq. I.3), resulting in considerable resource requirements ($N > Re^{9/4}$, Pope 2000). Philips [2012] uses the example of a flow around a 12-m high building to illustrate this point: the Reynolds number would be around 3.8×10^6 on a mild wind day with a 5 $\mathrm{m\,s^{-1}}$ breeze. Such high Reynolds number is far out of reach for DNS. More affordable CFD approaches such as RANS and LES can be derived from Eq. (I.6), but a turbulence model should be introduced (Sect. I.2.3).

The CFD paradigm accurately represents tracer dispersion within a complex geometry as the obstacles can be directly embedded in the computational grid and as the interactions between the wind and the obstacles can be explicitly solved.
Specific challenges arise from wind flow in urban areas: (1) high Reynolds numbers require fine grid resolutions and accurate wall functions in near-wall regions; (2) complex flow field dynamics with localised vortex impingement, separation and shedding; (3) numerical issues related to numerical schemes for flow around bluff bodies with sharp edges; and (4) inflow boundary condition that should be able to account for turbulence [Murakami, 1998].
To have confidence in CFD predictions, validation against wind-tunnel and field-scale experimental data is a necessary step. There is a strong synergy between trustworthy small-scale trials [Meroney, 2016] and CFD that can be transposed from these canonical configurations to considerably more complex field conditions.

I.2.3 Focus on RANS and LES approaches for urban dispersion

The two widely-used CFD approaches in urban flow simulations are RANS and LES approaches.While turbulence is fully modelled in the RANS approach, a significant fraction of the turbulence spectrum is explicitly resolved in the LES approach, which improves accuracy at the expense of increased computational cost (Fig. I.7). In this section, we describe the governing equations and briefly discusses the methods identified from the literature in the context of urban airflow modelling and dispersion.

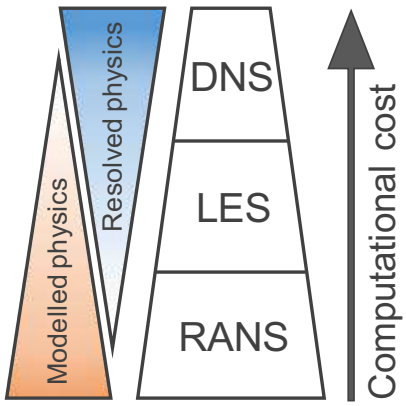

Figure I.7: Hierarchical representation by computational cost of most commonly-used CFD approaches: DNS, direct numerical simulations; LES, large-eddy simulations; RANS, Reynolds-averaged Navier-Stokes [Xiao and Cinnella, 2019].

I.2.3.a Governing equations

Reynolds averaging. For atmospheric dispersion, the RANS equations result from the application of Reynolds averaging to the flow and scalar quantities $\boldsymbol{u}$, p and $\mathbf{K}$ (Eq. I.8) that are involved in the Navier-Stokes equations and in the scalar transport equation. These quantities are formally decomposed into a mean part ($\overline{\cdot}$ operator) and a fluctuating part ($'$ superscript):

$$\boldsymbol{u} = \overline{\boldsymbol{u}} + \boldsymbol{u}' , \qquad \overline{p} + p' , \qquad \overline{\mathbf{K}} + \mathbf{K}' . \tag{I.8}$$

When the Reynolds averaging operation is applied to the original set of conservation equations (Eq. I.6), this results in the RANS equations for Navier-Stokes variables and scalar transport:

$$\begin{aligned} \frac{\partial \overline{u}_i}{\partial x_i} &= 0, \\ \frac{\partial \overline{u}_i\,\overline{u}_j}{\partial x_j} &= -\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}(2\,\nu\,\overline{s}_{i\,j}) - \frac{\partial}{\partial x_j}\left(\overline{u_i' u_j'}\right), \\ \frac{\partial \overline{\mathbf{K}}\,\overline{u_j}}{\partial x_j} &= \frac{\partial}{\partial x_j}\left(D\,\frac{\partial \overline{\mathbf{K}}}{\partial x_j}\right) - \frac{\partial}{\partial x_j}\left(\overline{\mathbf{K}'u_j'}\right), \end{aligned} \tag{I.9}$$

where $\overline{s}_{ij}$ is the mean strain-rate tensor expressed as:

$$\overline{s}_{ij} = \frac{1}{2}\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right). \tag{I.10}$$

The system of RANS equations cannot be directly solved because it has unclosed terms arising from the nonlinear advection terms: two additional second-order unknowns appear in the equations, namely the Reynolds stresses $\overline{u_i' u_j'}$ and the turbulent scalar flux $\overline{\mathbf{K}' u_j'}$. The modelling challenge lies in the construction of appropriate formulations to estimate these terms $\overline{u_i' u_j'}$ and $\overline{\mathbf{K}' u_j'}$. Such formulations are known as turbulence models.

Closure models for the Reynolds stresses. Closure models for the Reynolds stresses $\overline{u_i' u_j'}$ are derived using physical assumptions. They are usually classified as either first-order or second-order turbulence models. The first-order closure for Reynolds stresses is achieved using the Boussinesq (or eddy-viscosity) assumption, which is based on the similarity with momentum transfer through molecular motion in gases, which is characterised by molecular viscosity. Under this assumption, the Reynolds stresses are expressed as a function of the mean flow gradient and the turbulent eddy-viscosity ν_t:

$$-\overline{u_i' u_j'} = 2\,\nu\,\overline{s}_{i\,j} - \frac{2}{3}\,k_{tke}\,\delta_{i\,j}, \quad k = \frac{1}{2}\,\overline{u_i' u_j'}, \tag{I.11}$$

where k_{tke} is the turbulent kinetic energy. Turbulence models relying on the Boussinesq assumption are referred to as eddy-viscosity models (EVM). An appropriate estimation of turbulent viscosity is crucial for ensuring performance but adequacy is strongly case-dependent. Many formulations have been developed in the literature, among whom:

- the one-equation Spalart-Allmaras model [Spalart and Allmaras, 1994] used in building simulations, turbomachinery and aerospace applications, solving for a transport equation for kinematic eddy-viscosity and performing well for boundary-layers with adverse pressure gradients;

- the standard k-ϵ model [Jones and Launder, 1972] commonly used to simulate mean flow characteristics for turbulent flow conditions: it is a two-equation model on the turbulent kinetic energy k and the rate of dissipation ϵ;

- the standard two-equation k-ω model [Wilcox, 2008] on turbulence kinetic energy k and

specific rate of dissipation ω.

Popular first-order models such as k-ϵ and k-ω models have significant shortcomings in complex engineering flows due to the use of the eddy-viscosity assumption. Eddy-viscosity based closures cannot replicate the anisotropic behaviour of turbulent flows. Such models, for instance, perform poorly in flows with substantial anisotropy, considerable streamline curvature, rotational effects, flow separation, or recirculation flow. Reynolds stress equation models provide substantially higher accuracy in such cases.

The Reynolds stress equation model (RSM), often known as the second-order/moment closure model, is the most comprehensive method to classical turbulence modelling. It does not rely on the Boussinesq assumption; instead, transport equations for the Reynolds stresses are derived from the orginal set of equations. Still, this reports the problem of closure to third-order terms. Despite its finer description of the Reynolds stress tensor, the RSM model did not demonstrate consistent superior performance compared to EVM for urban wind engineering (e.g. Ferziger 1990; Murakami 1997, 1998; Nielsen et al. 2007).

Closure models for the turbulent scalar flux. For dispersion applications, closure models for the turbulent scalar flux $\overline{\mathbf{K}' u_j'}$ can be derived by analogy to the Reynolds stresses from the standard gradient diffusion hypothesis. The turbulent flux is then expressed as a function of the mean concentration gradient using a turbulent diffusivity D_t:

$$\overline{\boldsymbol{u}'\mathbf{K}'} = -D_t \frac{\partial \overline{\mathbf{K}}}{\partial x_i}, \tag{I.12}$$

where D_t varies spatially. It is generally related to the turbulent viscosity ν_t through a turbulent Schmidt number Sc_t as follows:

$$D_t = \frac{\nu_t}{Sc_t}. \tag{I.13}$$

Sc_t is generally assumed to be constant for a given type of flow. Numerous investigations highlighted the strong sensitivity to turbulent Schmidt number values for dispersion applications (e.g. Tominaga and Stathopoulos 2007, 2009, 2010, 2013; Gousseau et al. 2011; Gromke and Blocken 2015; Blocken et al. 2016b; Toja-Silva et al. 2017; Li et al. 2018; Kang et al. 2018). It is common practice to provide constant values for Sc_t in RANS simulations, albeit doing so can reduce significantly their modelling capability.

Higher order closure models and anisotropic models are also feasible for the turbulent scalar flux. However, such models are rarely used in CFD for building simulation in practice [Blocken, 2018]. A more detailed view of non-isotropic models is provided in Chapter V (Sect. V.1.4).

LES filtering. Unlike RANS, LES explicitly solves for the larger scales of turbulence. A distinction is made between the large eddies in the flow, mainly driven by the domain geometry, and the smaller eddies, which are expected to exhibit a more universal behaviour described by a subgrid-scale turbulence model. In this context, the Reynolds averaging operator is replaced by a filtering operation, which is also formally applied to the original set of governing equations (Eq. I.6).

Mathematically, a filter consists of a spatial convolution applied to the flow quantities to remove the eddies of size lower than a characteristic filter size Δ. For a given quantity ϕ, the corresponding filtered quantity can be expressed as

$$\widetilde{\phi}(\boldsymbol{x}) = \int_{\Omega} \phi(\boldsymbol{x}')\, G(\boldsymbol{x}, \boldsymbol{x}', \Delta)\, d\boldsymbol{x}', \tag{I.14}$$

where Ω is the fluid domain, and G is the spatial filter. The filter size Δ determines the scale of the resolved eddies. Similarly to the Reynolds decomposition, the original quantities can then be expressed as a resolved contribution ($\widetilde{\cdot}$) below the cut-off scale and a subgrid contribution ($'$ superscript):

$$u_i = \widetilde{u}_i + u_i', \qquad p = \widetilde{p} + p', \qquad \mathbf{K} = \widetilde{\mathbf{K}} + \mathbf{K}'. \tag{I.15}$$

The resulting filtered Navier-Stokes equations are expressed as:

$$\begin{aligned} \frac{\partial \widetilde{u}_i}{\partial x_i} &= 0, \\ \frac{\partial \widetilde{u}_i}{\partial t} + \frac{\partial \widetilde{u}_i \widetilde{u}_j}{\partial x_j} &= -\frac{1}{\rho}\frac{\partial \widetilde{p}}{\partial x_i} + \frac{\partial}{\partial x_j}(2\,\nu\,\widetilde{s}_{i\,j}) - \frac{\partial \tau_{i\,j}}{\partial x_j}, \\ \frac{\partial \widetilde{\mathbf{K}}}{\partial t} + \frac{\partial \widetilde{\mathbf{K}}\widetilde{u_j}}{\partial x_j} &= \frac{\partial}{\partial x_j}\left(D\frac{\partial \widetilde{\mathbf{K}}}{\partial x_j}\right) + \frac{\partial q_{i\,j}}{\partial x_j}, \end{aligned} \tag{I.16}$$

where $\widetilde{s}_{ij}, \tau_{ij}, q_{ij}$ characterise the strain rate tensor, the subgrid-scale Reynolds stresses and the subgrid-scale scalar flux defined as follows:

$$\widetilde{s}_{i\,j} = \frac{1}{2}\left(\frac{\partial \widetilde{u}_i}{\partial x_j} + \frac{\partial \widetilde{u}_j}{\partial x_i}\right), \qquad \tau_{i\,j} = \widetilde{u_i u_j} - \widetilde{u}_i \widetilde{u}_j, \qquad q_{i\,j} = \widetilde{\mathbf{K}}\widetilde{u}_j - \widetilde{\mathbf{K}u_j}. \tag{I.17}$$

To close the system of equations, similarly to the RANS approach, a suitable expression for the second-order terms must be determined. In LES, the model used to provide the closure is known as a subgrid-scale (SGS) model since it deals with the eddies of smaller size than the mesh cells [Pope, 2000]. The most widely used hypothesis in SGS models is the Boussinesq assumption:

$$\tau_{ij} - \frac{1}{3}\tau_{kk}\,\delta_{ij} = -2\,\nu_t\,\widetilde{s}_{i\,j}, \tag{I.18}$$

where ν_t is the SGS turbulent viscosity. Various SGS models are available for the estimation of ν_t. The Smagorinsky-Lilly SGS model [Smagorinsky, 1963] was the first to be developed. Assuming that the energy production is balanced by small-scale energy dissipation, it approximates ν_t as:

$$\nu_t = (C_s\,\Delta)^2 \sqrt{2\,\widetilde{s}_{ij}\,\widetilde{s}_{ij}}, \tag{I.19}$$

where Δ corresponds to the filter width related to the characteristic grid cell size, and $C_s \in [0.1, 0.2]$ is the Smagorinsky constant. Later on, many other models were developed such as the Germano approach providing a dynamic estimation of C_s [Germano et al., 1991; Lilly, 1992].

I.2.3.b LES and RANS in the context of urban dispersion

When choosing between RANS and LES approaceches, there are two main factors that come into play: the simulation cost and the accuracy. The computational cost of LES is higher than for RANS simulation. The LES equations are derived from the governing equations in a similar manner to that of the RANS equations. However, because of the use of spatial domain filtering and the resolution of a part of the turbulent spectrum, the flow must be simulated on much finer meshes, leading to a significant computational cost increase.

The low cost of the RANS approach is the main reason why, in outdoor building simulation applications, RANS was first used rather than LES. The ability to perform more simulations at a reduced cost allows for greater coverage of uncertainty on flow conditions (e.g. urban topography, atmospheric conditions, tracer emission source) [Vervecken et al., 2013; Margheri and Sagaut, 2016; García-Sánchez et al., 2017]. The plurality of RANS studies now offers robust guidelines for a wide range of configurations [Blocken, 2018]. In the context of flow around isolated obstacles, studies cover sensitivity to the grid resolution (e.g. Murakami and Mochida 1989; Murakami et al. 1990a,b; Baskaran and Stathopoulos 1992), the boundary conditions (e.g. Murakami and Mochida 1989; Paterson and Apelt 1990; Baetke et al. 1990; Stathopoulos and Baskaran 1990; Baskaran and Stathopoulos 1992), and the turbulence model (e.g. Baskaran and Stathopoulos 1989; Murakami et al. 1992; Murakami 1993; Mochida et al. 2002). In the context of dispersion, the increasing availability of computational power allowed to study multi-obstacle urban-like configurations (e.g. Murakami 1997; Stathopoulos and Baskaran 1996; Hanna et al. 2006; Philips et al. 2013; Blocken et al. 2016b; García-Sánchez et al. 2017).

If it were only a matter of computational cost, RANS would be a more appropriate method for dispersion. However, RANS does not always succeed in accurately simulating urban flows. Urban wind studies at pedestrian level have shown that the RANS approach can properly model the mean wind speed in areas of high wind speed. However, their effectiveness can be compromised in areas of low wind speed (Yoshie et al. 2007; Blocken and Carmeliet 2008; Blocken et al. 2008a, 2011, 2016a). More generally, the RANS approach performs poorly in strong anisotropic cases, when the flow features vortex shedding in the wake of bluff obstacles or counter gradient diffusion effects, for instance in separation and recirculation regions, which are particularly prevalent in urban topography with sharp-edged obstacles, leading to poor estimation of the turbulent kinetic energy (Murakami 1993; Yoshie et al. 2007; Blocken et al. 2008a).

LES have recently emerged as a benchmark solution for accurately representing the highly unsteady and complex flow topologies typically found in the wake of buildings in urban canopies [Philips et al., 2013; Vervecken et al., 2015a; Grylls et al., 2019]. In particular, the LES approach demonstrates improved prediction accuracy over the RANS approach for highly unsteady flows with strong anisotropy, typically found in the wake of urban-like obstacles [Tominaga and Stathopoulos, 2010; Blocken, 2018]. In addition to improved accuracy, the LES approach provides access to high-order quantities of interest (e.g. turbulent scalar flux, pressure-scalar fluctuation correlation), which serve to identify deficits in lower fidelity models (e.g. RANS) and guide models improvements. For instance, Tominaga and Stathopoulos [2012] studied the ability of LES to recover the turbulent scalar flux for near-field dispersion around buildings, and

compared results with conventional RANS models. With the emergence of machine learning, there is also a growing interest in building data-driven RANS models, where conventional RANS models can be informed from higher fidelity LES data Duraisamy et al. [2019b].

Dispersion modelling approaches differ in the fundamental modelling approach to turbulent dispersion and their assumptions on *i*) the flow boundary conditions, and *ii*) the representation of turbulence in terms of scales (e.g. Reynolds average, LES filter) and of closure terms (e.g. RANS scalar flux closure model, LES SGS model). LES and RANS stand as two versatile, accurate and feasible approaches to study microscale dispersion in an uncertainty context. In this PhD thesis, we explore how to combine LES and RANS approaches with machine learning to improve prediction performance. The next section summarises the state-of-the-art in statistical learning methods for CFD.

I.3 Machine learning for computational fluid dynamics

Over the past decades, fluid mechanics data have been collected through field and wind tunnel experiments, as well as numerical simulations to improve the understanding of physics for a variety of engineering and environmental problems [Rogallo and Moin, 1984; Goldstein, 2017; Dauxois et al., 2021]. As long as the volume of data remained limited, major improvements were due to expert knowledge, elementary statistical analysis and intuitive design. Recent technological improvements have lowered the costs involved in data acquisition, storage and transfer. Consequently, the volume of data in all areas has dramatically expanded, opening up exciting opportunities for applying advanced learning approaches to the field of fluid mechanics [Brunton et al., 2019].

Machine learning has emerged with the goal of learning relevant domain knowledge from large volume of data using appropriate algorithms. It has gained popularity in the last decade due to the strong growth of data in many scientific fields, and fluid mechanics is no exception [Pollard et al., 2017]. For instance, in the field of urban flow mechanics, machine learning has a role to play to effectively and accurately quantify uncertainties in urban flow and pollutant concentration simulations, which is essential to mitigate adverse health effects from pollution [Dauxois et al., 2021].

To properly identify the opportunities arising from machine learning for fluid mechanics, it is crucial to understand (1) which are the specific needs and challenges for fluid mechanics issues, and (2) which data are available. These aspects are discussed in Sect. I.3.1. The focus is then made on two approaches for improving CFD using machine learning: designing reduced-order models on the one hand (Sect. I.3.2), and informing turbulence models on the other hand (Sect. I.3.3).

I.3.1 Challenges and opportunities

Fluid mechanics is often related with critical fields of application (e.g. energy transition, transport, health, safety and military engineering technology), where interpretability, explicability

and generalisation capacity of the modelling approach are of primary importance to guarantee accuracy and reliability.

Fluid flows exhibit complex multiscale phenomena including eddies of varying size and energy. The resulting phenomena may be very local, with highly nonlinear responses ranging with strong scale disparity. For example, urban flows are governed by large-scale weather patterns down to Kolmogorov microscale turbulence. Machine learning approaches suitable for fluid mechanics problems should therefore be able to reproduce this multiscale nature of the flows.

The Navier-Stokes equations accurately represent flow dynamics but this comes with a high level of complexity in the implementation and resolution of these equations [Bai et al., 2017]. In fact, flow complexity often requires accurate computational methods to avoid distorting the underlying physics with numerical artefacts. Fine computational meshes required to resolve the smallest scales leads to exceedingly high dimension and computational cost [Choi and Moin, 2012]. This leads to unfavorable trade-off between accuracy and tractability, especially if the objective of the modelling system is to perform real-time risk assessment. Furthermore, applications are usually subject to uncertainties and require to go beyond the classical deterministic modelling framework for CFD. For instance, uncertainties in atmospheric microscale dispersion modelling come from large-scale atmospheric conditions that have their own intrinsic variability, making the atmospheric boundary conditions for the microscale domain uncertain [García-Sánchez et al., 2014]. Furthermore, in an accidental case, the event characteristics (e.g. position, flow and composition of the emission source) are partially known, while they are of primary importance to map the near-source peak concentrations. There is therefore a need to develop a novel probabilistic modelling approach that is suitable for fluid mechanics problem and that can describe the variety of plausible scenarios [Dauxois et al., 2021]. While fine meshes enable high-dimensional solutions in space and time discretisation accurately captures the temporal dimension, sampling the uncertain space is computationally expensive as it requires performing multiple CFD simulations and as each CFD simulation alone is already costly.

In this uncertainty quantification context, using very high-fidelity modelling approaches (DNS, LES) comes at the cost of a small number of simulations to represent the investigated phenomena. In the opposite, using more affordable but less accurate modelling approaches (e.g. RANS) allow to increase the number of simulations and thereby the number of scenarios to explore. In this context, machine learning techniques provide new opportunities [Brunton et al., 2016; Duraisamy et al., 2019b; Vinuesa and Brunton, 2022]. We can mention three of them: (1) speed up DNS or LES simulations to allow more simulations to be carried out; (2) synthesise the information from a small number of DNS or LES simulations to substitute their future use by inexpensive models such as reduced-order models, and (3) allow the use of lower fidelity models such as RANS but improve their performance by exploiting high fidelity data (Fig. I.8).

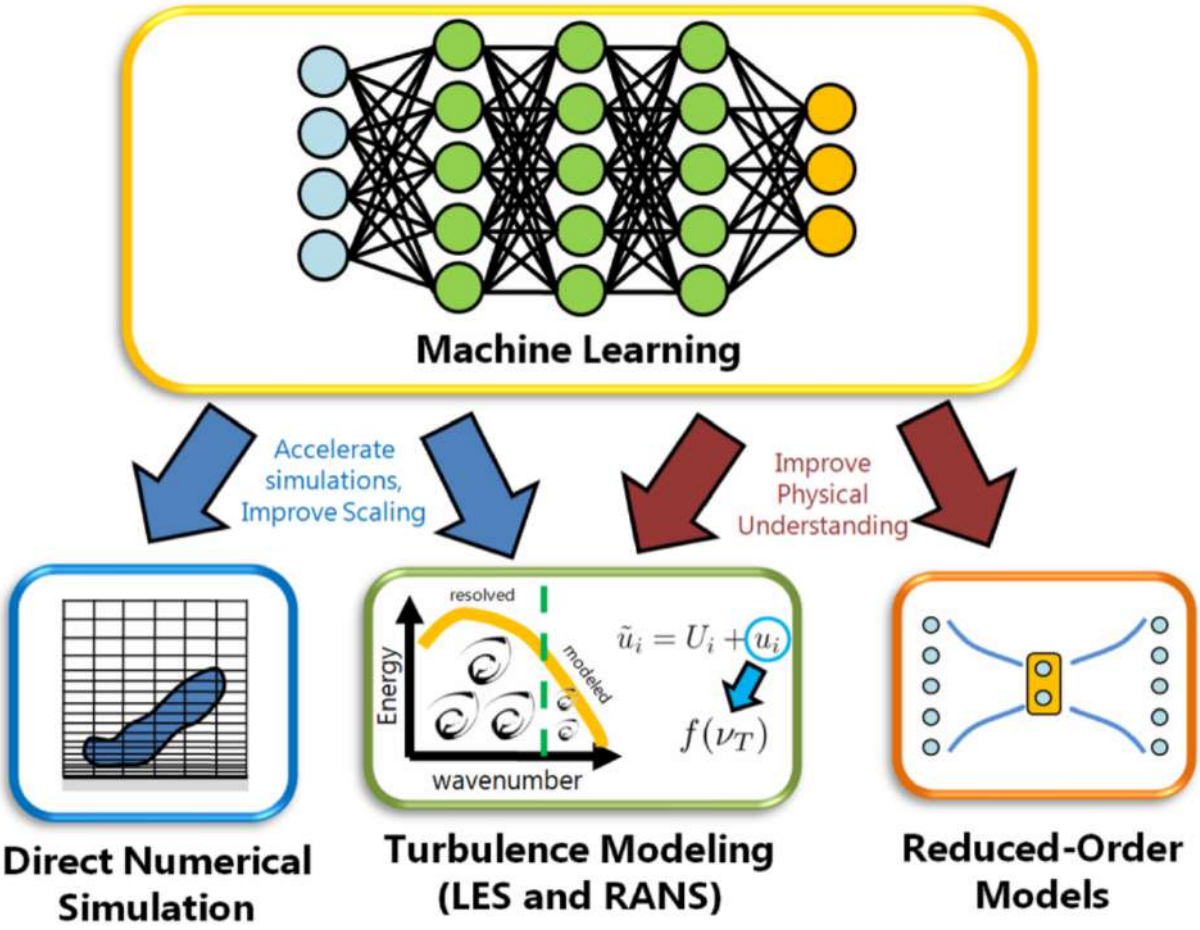

Figure I.8: Overview of topics where machine learning enhances CFD models [Vinuesa and Brunton, 2022]; ranging from computationally-intensive DNS to LES and RANS, whose performance relies on multiple parameters and assumptions. The information gathered (through simulation or observation) can be aggregated into cheaper models known as reduced-order models.

This thesis mainly focuses on the second approach "synthetise high-fidelity data", with the key idea of exploring new ways to synthesise LES information into a data-driven model for direct prediction of quantities of interest. The third approach "improve lower fidelity models" is also addressed by informing RANS modelling approach for tracer dispersion with high fidelity information from LES. Recent machine learning algorithms allow to extract relevant information and to emulate fluid flows in an optimised reduced-order modelling procedure from a relatively small amount of data. They provide opportunities to directly benefit from the high-fidelity information of LES in a context of uncertainty quantification (Sect. I.3.2). They also provide opportunities to improve RANS predictions whose performance highly depends on closure models, which can be improved or substituted by machine learning (Sect. I.3.3). Yet, the literature on the combination of high-fidelity CFD models with data-driven techniques is on the rise but still in its early stages. While a number of research have already been conducted in the context of atmospheric dispersion (e.g. Vervecken et al. 2013, 2015b; García-Sánchez et al. 2014, 2017; Margheri and Sagaut 2016; Lange et al. 2021; Mendil et al. 2022), much work remains to be done to develop efficient data-driven approaches and to identify the most relevant models from the variety of machine and deep learning models available.

I.3.2 Reduced-order modelling for computational fluid mechanics

Data-driven techniques such as machine learning and deep learning may be used to design reduced-order models for fluid dynamics. Reduced-order models rely on the assumption that the flow complexity is essentially carried by a small number of dominant structures. As a result, reduced-order models only characterise the evolution of the dominant structures in order to produce fast predictions, which may be used instead of direct CFD simulations. Reduced-order model efficiency comes at the expense of generalisation capacity. As reduced-order models are tailored to specific flow setups, they provide a huge speedup but on limited scope.

I.3.2.a Challenges for reduced-order modelling

Expensive high-fidelity simulations such as LES may be required for a large number of environmental/industrial applications for their ability to accurately resolve the flow in complex environment and produce accurate flow statistics (e.g. urban environment, Philips et al. 2013; García-Sanchez et al. 2018). Despite these advantages, their computational cost raises challenges in an operational context, particularly for real-time risk assessment. In this context, reduced-order modelling may be a way to provide fast surrogate models for CFD solvers, enabling optimisation and control tasks that involve many model iterations or quick response feedback.

Reduced-order models for fluid mechanics must meet the challenges of interpretability, accuracy, and robustness. Ultimately, they would embed knowledge of the physics to ensure generalisation capacity to various multi-request flow configurations, initial and boundary conditions, etc. [Loiseau and Brunton, 2018; Wang et al., 2020; Guan et al., 2021; Frezat et al., 2021]. To tackle these objectives, reduced-order models rely on the assumption that even complex flows can be described by a few dominant flow patterns. The resulting sparsity produces coherent structures that uncover underlying physics laws [Taira et al., 2017, 2020]. For instance, common reduced-order modelling strategies constrain the flow system on a learned low-dimensional space of projection. The reduced-order models simulate the evolution of the flow system projected components, which are eventually transformed back to the initial high-dimensional space.

As a result, the task is twofold: reduced-order models must (i) provide an optimal reduced-basis coordinate space of representation, and (ii) characterise the (possible strongly nonlinear) evolution of the reduced-basis projected components with respect to uncertain parameters and/or time [Brunton et al., 2019]. Both correspond to machine learning tasks since they aim at establishing links between inputs and outputs of a system based on data, yet they are of different type (Fig. I.9). The task (ii) is a regression problem that can be qualified as supervised learning. Supervised learning involves labelled data, where labels corresponds to the outputs of the learning algorithm (obtained for instance from expert evaluation), with the objective to learn general rules that map inputs to outputs for prediction, optimisation, etc. In contrast, the task (i) belongs to the category of unsupervised learning and more specifically dimensionality reduction. It aims at identifying data underlying structure and at extracting features when no label is given. Dataset snapshots generally carry many features, and the learning task is then to learn useful data patterns for data compression, correlation analysis, noise reduction, etc. Since

CFD solvers are expensive, appropriate algorithms are required to efficiently solve tasks (i) and (ii).

It is worth mentioning that for supervised and unsupervised learning tasks, the dataset is fixed. This is no more the case for reinforcement learning where the learning algorithm does not have access to labelled data but interacts with the environment to gradually learn correct information using reward and punishment for each action done by the algorithm. This type of semi-supervised learning algorithms will not be further discussed in this thesis; more information can be found in Mnih et al. [2013], and Sutton and Barto [2018].

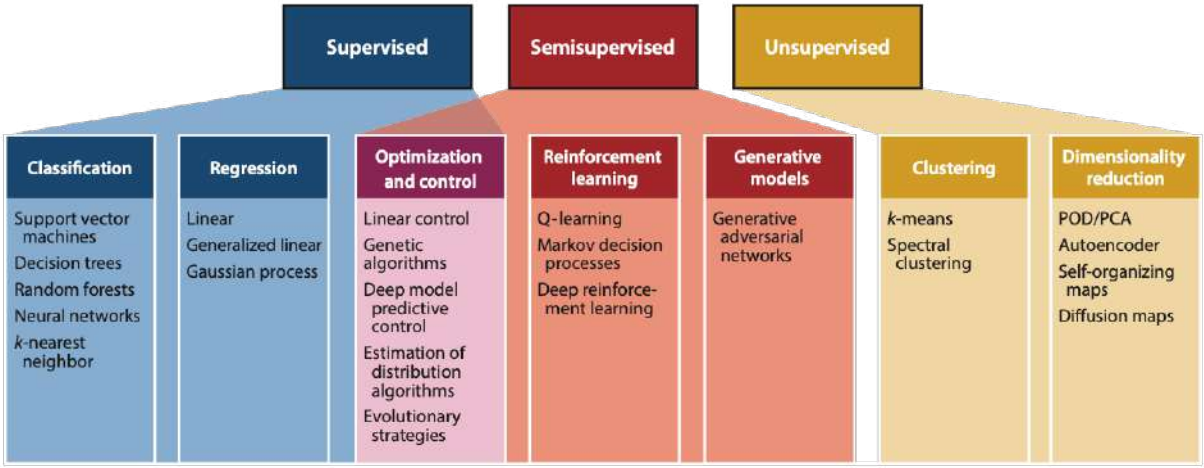

Figure I.9: Categories of learning algorithms depending on the target task, the nature and amount of available data (PCA stands for principal component analysis, and POD for proper orthogonal decomposition) [Brunton et al., 2019].

I.3.2.b Reduced-basis representation

To tackle the first task (optimal reduced-basis representation i), it is necessary to build a low-dimensional space of representation (the latent space) as well as transformation functions (the encoder and decoder, respectively). These functions represent the mapping between the high-dimensional space and the reduced dimensional space, i.e. that compresses (encodes) and decompresses (decodes) information (Fig. I.10a). This constrains the amount of candidate algorithms. Two families of methods suitable for this task are presented here.

Proper orthogonal decomposition. Proper orthogonal decomposition (POD) [Sirovich, 1987; Berkooz et al., 1993] stands as a core dimensionality-reduction data-driven modelling technique for CFD. POD provides a linear subspace of orthogonal modes to approximate field data in a non-intrusive manner, resulting in simpler straightforward yet efficient and interpretable reduced-order models. This technique has therefore been used in a wide variety of problems, for instance to develop a low-dimensional parametrisation for nonlinear Poisson equation and cavity viscous flows [Hesthaven and Ubbiali, 2018], for pedestrian turbulent wind flow and toxic gas dispersion in a full-scale city area [Margheri and Sagaut, 2016; Xiao et al., 2019], or for a flow around an airfoil [Swischuk et al., 2019].

POD can be extended to handle time evolution with dynamic mode decomposition (DMD)

[Deng et al., 2021; Schmid, 2022]. Robust algorithms of POD can find some applications for fluids with highly corrupted data. For instance, Scherl et al. [2020] applied robust principal component analysis (PCA) to a modal decomposition of a turbulent channel flow. For non-negative quantities of interest, the non-negative matrix factorisation (NMF or NNMF) algorithm [Wang and Zhang, 2012] is an interesting alternative to POD. Gleichauf et al. [2020] showed its potential to better distinguish flow regimes from thermographic images .

However, linear methods such as POD may not be very effective for systems that evolve on strongly nonlinear manifolds. In particular, Murata et al. [2020] reported that well-designed methods may be necessary to handle complex turbulent flows. Indeed, many spatial modes may be required to represent the fine structures of turbulence. Several nonlinear algorithms can be suggested. For instance, PCA has been extended to nonlinear manifold representation with kernel PCA [Schölkopf et al., 1998]. The Gaussian process latent variable model (GPLVM) [Lawrence, 2003; Titsias and Lawrence, 2010] performs dimensionality reduction from local correlation structure and uncertainty by merging latent variable model framework with Gaussian processes. However, these models do not provide an encoder formulation as seen in Table I.3, which summarises the capability of different approaches. It may partly explain the lack of research momentum in their direction relatively to neural networks.

Table I.3: Overview of dimensionality reduction algorithms (adapted from Lawrence, 2005), where the encoder corresponds to the mapping from the input high-dimensional space onto the latent space, and where the decoder performs the inverse mapping from the latent space onto the input high-dimensional space.

	Encoder	Decoder	Nonlinear
POD	✓	✓	
Robust PCA	✓	✓	
NMF	✓	✓	
Kernel PCA		✓	✓
GPLVM		✓	✓
Autoencoder	✓	✓	✓

Neural-network autoencoders. Neural networks offer interesting prospects for dimensionality reduction. They can be organised into a bottleneck architecture (Fig. I.10a), thus forming nonlinear autoencoders for developing both latent space representation and nonlinear encoder/decoder transformations [Le Cun and Fogelman-Soulié, 1987; Bourlard and Kamp, 1988; Hinton and Zemel, 1993]. However, basic neural network architectures failed to capture the patterns in pixel data. This is why LeCun et al. [1989] introduced convolutional autoencoders composed of convolutional layers to deal with grid-like architecture data such as images.

Autoencoders can naturally handle nonlinearities and are therefore thought as nonlinear extension of POD [Milano and Koumoutsakos, 2002; Goodfellow, 2010]. POD solutions can indeed be achieved by a linear implementation of autoencoders minimising the squared-error loss function [Baldi and Hornik, 1989]. In the fluid mechanics community, autoencoders have shown promising performance results. Milano and Koumoutsakos [2002] implemented a nonlinear au-

toencoder from DNS to emulate the near-wall field in a turbulent flow channel. Murata et al. [2020] applied a convolutional autoencoder to a laminar cylinder wake and its transient process, and demonstrated improved performance compared to POD. Similar results were obtained for turbulent flows [Eivazi et al., 2022].

Even though nonlinear autoencoders may be more efficient than POD, they usually lack interpretability. Interpreting the physical meaning of the latent vector in autoencoders is exceedingly challenging. The autoencoder modes cannot be arranged through an energy criterion like POD modes. Saegusa et al. [2004] developed the concept of hierarchical autoencoder to extract the autoencoder modes in their contribution order. Later, Fukami et al. [2020] extended this approach to mode-family hierarchical autoencoders and applied it to a two-dimensional cylinder wake and its transient process for both laminar and turbulent flows. Unfortunately, these architectural enhancements do not succeed in making autoencoders as explainable as POD, thus resulting in a trade-off between model interpretability and performance.

I.3.2.c Reduced-basis projected component evolution

POD and autoencoders are powerful approaches to project high-dimensional field data onto a latent space. However, they do not model the flow field response with respect to time or when varying the input parameters. When building a reduced-order model, some type of interpolation techniques is therefore required to model (or metamodel) how the latent variables (or the reduced components) vary with respect to time or to uncertain input parameters. Basic interpolation techniques may fail to capture major patterns of the flow response [Brunton et al., 2019]. Machine learning algorithms offer a fairly generic framework to solve this metamodelling problem.

In the context of reduced-order modelling, regression models learn from the CFD database the mapping between uncertain input parameters and latent variables over a wide range of variation for the input parameters [García-Sánchez et al., 2014; Margheri and Sagaut, 2016; García-Sánchez et al., 2017; Lamberti and Gorlé, 2021]. For instance, García-Sánchez et al. [2014, 2017] studied how parametric atmospheric uncertainties (inlet wind magnitude and direction, terrain roughness length) propagate on flow and pollutant dispersion quantities in downtown Oklahoma City based on ensemble RANS simulations combined with polynomial chaos expansion. This pioneering work demonstrated the feasibility and interest of emulating RANS simulations for microscale pollutant dispersion applications. However, García-Sánchez et al. [2014, 2017] did not provide an analysis of the polynomial chaos expansion metamodel error, whereas it is a necessary step to develop reliable and robust reduced-order models adapted to a variety of atmospheric boundary-layer flows and dispersion conditions.

In the literature, a wide variety of methods have been applied for predictive regression learning on numerical data such as neural networks [Hesthaven and Ubbiali, 2018; Swischuk et al., 2019; Ma et al., 2021; Lucor et al., 2022], multilinear models on nonlinear feature space (e.g. SINDy – Brunton et al. [2016]; polynomial chaos expansion – Rochoux et al. [2014]; García-Sánchez et al. [2014, 2017]; El Garroussi et al. [2022]), decision trees [Xiang et al., 2021] and Bayesian frameworks such as Gaussian process regression (GPR) models [Margheri and Sagaut, 2016; Guo and Hesthaven, 2018; Xiao et al., 2019]. The search for adapted machine learning

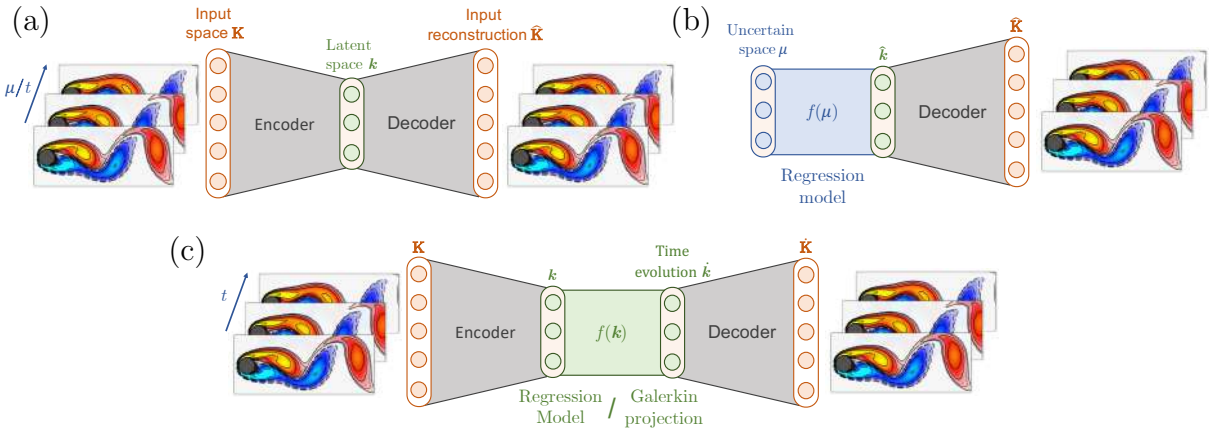

Figure I.10: Schematic of reduced-order models adapted from Vinuesa and Brunton [2022], which represent the evolution of high-resolution flow fields $\mathbf{K}$ with respect to time t and/or uncertain input parameters $\boldsymbol{\mu}$. (a) Autoencoder structure for dimensionality reduction with an encoder to map the high-dimensional data $\mathbf{K}$ onto the low-dimensional latent space $\boldsymbol{k}$ and a decoder to have an approximate reconstruction $\widehat{\mathbf{K}}$ of the high-dimensional data. (b) Regression model structure to metamodel the latent variables $\widehat{\boldsymbol{k}}$ and recover high-dimensional features $\widehat{\mathbf{K}}$ using the decoder. (c) Regression model structure in the specific context of time evolution, where the time-evolution of the latent variables $\dot{\boldsymbol{k}}$ can be handled by classic Galerkin-projection model or machine-learning regression.

tools coincides with the requirement for a careful and documented study of the processed data, the reduced-order model performance, its robustness, and its explicability. In this direction, Swischuk et al. [2019] examined the effectiveness of several regression modelling approaches for predicting high-dimensional flow outputs. They noted the need for interpretation of reduced-order model predictions, but also the choice of the reduced-order model approach depending on available data. To the best of our knowledge, such analysis is missing in the context of microscale pollutant dispersion modelling.

It is worth noting for the specific case of flow time evolution (Fig. I.10c), Galerkin projection [Barone et al., 2009; Hijazi et al., 2020] may replace machine learning regression models. The Galerkin projection benefits from a closer connection to the governing equations, and can allow for a good integration of physical constraints [Loiseau and Brunton, 2018]. However, these advantages require a numerical implementation of the governing equations, sometimes even intrusive modifications of the numerical solver that are easy to implement in legacy codes [Vinuesa and Brunton, 2022]. We do not investigate this approach in this work since we focus on response to input parameters rather than time evolution.

Reduced-order modelling for CFD can be described as a two-step approach: (1) dimension reduction to reduce spatial flow fields to a small number of highly representative components; and (2) regression modelling (or metamodelling) to represent flow response to variation in uncertain inputs parameters. Both tasks rely on a dataset of CFD simulations that provide high-fidelity data as training examples. Machine-learning models stand as popular candidates to perform these tasks. When successful, the reduced-order models efficiently emulate fluid flows at a fraction of the CFD solver computational cost. Yet, this efficiency comes a loss of generality. Reduced-order models are designed to perform well for specific flow configurations (included in the training database), allowing for tremendous speed-up but limited generalisation capacity. Consequently, it is challenging to obtain guarantees of reduced-order model's performance and robustness, which are of first importance for risk assessment problems.

I.3.3 Data-driven turbulence closure models

DNS and well-resolved LES of the Navier–Stokes equations is impractical for many real-world applications due to the high computational cost required to resolve the wide range of turbulence length and time scales, together with flow subtleties arising from complex geometry. For this purpose, industrial and environmental CFD modelling problems mostly rely on RANS approaches (e.g. Milliez and Carissimo, 2007, for simulating pollutant dispersion in an idealized urban area) due to their relatively low computational requirement [Tracey et al., 2013].

Contrary to LES, no turbulent scales are directly simulated in the RANS approach. However, closing the system of equations requires modelling the nonlinear Reynolds stress term introduced by the RANS time-averaging procedure (Sect. I.2.3.a). Many turbulence models have been developed [Hanjalić and Launder, 1972; Spalart and Allmaras, 1994; Wilcox, 2008]. However, their accuracy for turbulent flows can be limited. Shortcomings of eddy-viscosity models for capturing the Reynolds stress anisotropy has been demonstrated in many engineering turbulent flows (e.g. flows with body force effects arising from curvature, system rotation, impingement, separation [Lumley, 1979; Launder, 1990; Speziale, 1991]). Even nonlinear stress-strain relationships in turbulence modelling [Craft et al., 1996; Wallin and Johansson, 2000; Pope, 2000] have not been widely adopted since they do not provide consistent performance, and might result in decreased convergence and stability [Gatski and Speziale, 1993; Belhoucine et al., 2004]. Consequently, there is a need to improve RANS approaches with better designed turbulence models.

Recent investigations have been carried out in using machine learning methods to design new data-driven model components for RANS models, and in particular to develop new Reynolds stress closures from high-fidelity simulation data. For instance, Tracey et al. [2013] used kernel regression to model the Reynolds stress anisotropy tensor from high-fidelity DNS data for a channel flow. Tracey et al. [2015] implemented a three-layer multilayer perceptron (i.e. one of the most simple neural network structures) to emulate boundary-layer flows. Parish and Duraisamy [2016] demonstrated the ability of Gaussian process regression to provide corrections of a standard k-ω closure model using DNS in a turbulent channel flow. Ling et al. [2017] implemented random forest (ensemble of decision trees) regression to emulate Reynolds stress anisotropy from LES data in a low Mach, incompressible, jet-in-crossflow configuration. The

significant improvement over baseline RANS eddy-viscosity models is of particular interest for complex flow problems. Further investigations on machine learning applications to improve turbulence modelling emphasised how imposing physical constraints and incorporating uncertainty quantification alongside machine-learning-based models could improve accuracy and robustness [Mishra and Iaccarino, 2017; Duraisamy et al., 2019a; Rezaeiravesh et al., 2021]. Ling et al. [2016a] demonstrated the ability of random forests and neural networks to predict the Reynolds stress anisotropy with rotationally invariant input features. Ling et al. [2016b] demonstrated the ability of simple-designed multilayer perceptrons to enforce Galilean invariance into the predicted anisotropy tensor.

In parallel with this work on the RANS approach, investigations have also been conducted to improve LES subgrid-scale models using machine learning. For instance, Maulik et al. [2019] and Beck et al. [2019] demonstrated the ability of multilayer perceptrons and convolutional neural networks to predict subgrid-scale model terms using DNS high-fidelity data for a canonical problem of decaying homogeneous isotropic turbulence. More complex flows can be addressed with convolutional neural networks such as buoyancy-driven flows [Ajuria Illarramendi et al., 2022] and premixed turbulent flames [Lapeyre et al., 2019].

In recent years, machine learning approaches have shown their potential to improve turbulence modelling by providing a powerful statistical framework to integrate detailed information on a variety of flow configurations from high-fidelity numerical data. In microscale pollutant dispersion problems, due to its limited computational cost compared to LES, the RANS approach remains the most widely used CFD approach, even if the interactions between the flow and the buildings may induce complex flow structures that are difficult to capture by RANS. For risk assessment problems, it is therefore of high interest to improve the RANS approach by integrating information from a LES database that can be generated offline.

I.4 Aim of the thesis

I.4.1 Scope of the thesis

Better predicting wind patterns and pollutant concentration in the urban canopy is essential for city safety, resilience and sustainability, in particular for mitigating health impacts from air pollution due to chronic emissions and/or accidental emissions (e.g. Lubrizol industrial accident in 2019 in Rouen, France). Microscale pollutant dispersion phenomena in urban areas are complex due to (1) local flow structures driven by the urban geometry, and (2) uncertainties associated with the variability of large-scale meteorological conditions and the limited information on pollutant emission characteristics in the event of an accident.

Simplified modelling approaches such as Gaussian plume models are unable to capture complex dispersion patterns of microscale urban flows. More sophisticated approaches based on the CFD modelling paradigm can provide insights into the wind-building interactions but their computational cost is significant and they do not account for uncertainties limiting their individual prediction capability. As emphasized by Dauxois et al. (2021), there is a need to design

novel probabilistic modelling strategies for detailed urban flow simulations, which can efficiently generate and handle an ensemble of scenarios representative of the uncertainties at play.

In this thesis, we investigate machine learning approaches to harness the high-fidelity capabilities of CFD (LES and RANS) in a context of uncertainties on the urban flow simulations. Ultimately, the aim is to benefit from accurate CFD outputs without the drawback of high computational cost. Machine learning tools offer an interesting solution by taking advantage of the high-fidelity CFD data. They have the potential to learn how to synthesise the most relevant flow patterns in a multi-query context, in order to deliver new predictions for a wide range of parameter variation with a very low computational cost. The challenge is to limit the loss of information to design robust reduced-order models that could, in the long term, be used for operational risk assessment studies.

The research group supervised by Catherine Gorlé at Stanford University (e.g. García-Sánchez et al. 2014, 2017; Sousa and Gorlé 2019) introduced the idea of metamodelling wind and tracer concentration fields for RANS modelling approaches applied to real-scale urban flows (Joint Urban 2003 Experiment in Oklahoma City). The proposed metamodel is based on a polynomial chaos expansion and is directly applied to each grid-point of the field quantities of interest (time-averaged wind velocity and tracer concentration fields). There is no dimensionality reduction approaches to compress them, unlike in the work proposed by Margheri and Sagaut [2016] for instance. We consider that dimensionality reduction is an important component of a reduced-order model to account for the spatial correlations inherent to the coherent structures of urban flows and to have quantitative arguments to define the optimal latent space. Unfortunately, no machine learning algorithm is universally embraced; the learning community frequently argues that there is no universal model and that the algorithm to be used depends on the task at hand. To overcome this issue in the context of microscale pollutant dispersion modelling, in this work, we carry out a detailed comparison of different metamodelling approaches based on machine learning.

This PhD thesis at the interface between environmental CFD, machine learning and uncertainty quantification was funded by CERFACS, a private research laboratory in Toulouse, France. The work was done through a collaboration between CECI (climate-environment topics) and CFD (fluid mechanics-combustion topics) teams at CERFACS, and the LISN laboratory (interdisciplinary numerical sciences) at Université Paris-Saclay.

I.4.2 Key objectives of the thesis

The core idea of this thesis is to design, evaluate and provide a detailed comparison of reduced-order modelling approaches, which include a dimensionality reduction component and a regression algorithm component to emulate fields of interest. We consider a simplified case of urban microscale atmospheric dispersion with a simple-designed but representative isolated obstacle to make this detailed analysis without too much restriction on the size of the learning database due to computational cost constraints. This analysis provides guidelines for future field-scale applications such as the MUST experiment. The goal is to predict plume flow statistics (e.g. the mean tracer concentration around the obstacle) to better represent and understand the spatial

variability of the plume processes for a wide range of inflow wind and emission source conditions. These uncertain conditions are represented through a set of uncertain scalar parameters, and reduced-order models aim at propagating these uncertainties to the field statistics of interest. The main difficulty in setting up these reduced-order models is to handle the high-dimension of the field statistics and their possible nonlinear response to changes in the input parameters.

In this thesis, two main objectives are addressed. The first objective is to directly emulate the LES fields of interest using non-intrusive reduced-order models. The second issue is to design an indirect intrusive approach to construct an hybrid approach combining RANS tracer transport equation with LES airflow data using machine learning.

First main goal: Emulating the LES field statistics of interest using machine learning to design a data-driven reduced-order model.

We aim at designing and evaluating a reduced-order model suitable for LES, which learns the mapping between input uncertainties (associated with atmospheric conditions and source location) and the output concentration statistics. This is a challenging problem due to the limited database that is available for training, validation and test. This is especially true in a LES setting that requires more computational resources than the RANS approach and that limits the size of the training database. For these reasons, in this thesis, we explore several learning algorithms to reduce the dimension (e.g. using POD or neural-network autoencoder) and to conveniently map inputs onto outputs through a regression model (e.g. using polynomial chaos expansion, Gaussian process regression, gradient tree boosting). A detailed analysis of each reduced-order model component performance is carried out to evaluate the accuracy and robustness of the different approaches. The interactions between dimension reduction and regression models are also explored to find ways to better pose the problem and gain efficiency.

Second main goal: Informing the RANS scalar transport equation from LES airflow data to design a hybrid reduced-order model.

Even though non-intrusive reduced-order models are easy to implement and can provide tracer concentration predictions at a low computational cost for a wide range of atmospheric and source conditions, there is no guarantee of accuracy, robustness and consistency with physics. They do well on average, yet they may perform poorly in some very unique scenarios that were not well represented in the training database. One of the main shortcomings is the lack of constraints with the major physical laws and assumptions (e.g. mass conservation). In particular, uncertainties related to tracer source location may lead to a sharp response in parametric space, which challenges the reduced-order models and requires quite a significant number of LES snapshots to obtain accurate predictions. To overcome this issue, we explore an alternative approach embedding physical constraints through the use of hybrid strategy using LES-trained learning algorithms and a RANS transport equation to solve the plume statistics. The key idea of the proposed approach is to decouple the atmospheric parameter uncertainties from the source parameter uncertainties. First, the LES approach allows to efficiently sample the atmospheric parameter uncertainties and to build a reduced-order model for quantities related to turbulence

that are relevant in the RANS framework. Similarly to the first objective, the LES information is synthesised in a reduced-order model before being fed to the RANS model. Then, the LES-informed RANS model is then used to simulate plume statistics for a wide range of source locations, which is of interest for risk assessment perspectives. Finally, the potential of multi-fidelity combining a small number of high-fidelity LES snapshots with a large number of lower fidelity LES-informed RANS model predictions is explored to gain in robustness. Multi-fidelity could be a way to move towards full-scale prediction capability of the reduced-order model, while remaining within the limits of acceptable computational cost as emphasised by Dauxois et al. [2021].

The structure of the manuscript is as follows. Chapter II gives a theoretical overview of reduced-basis and metamodelling methods implemented in this thesis. Chapter III introduces the CFD case study, the parametric uncertainties as well as the strategy proposed to design the reduced-order models and assess their performance. As a first emulation approach, Chapter IV discusses the results obtained on the non-intrusive emulation of the LES field statistics of interest. Chapter V discusses the results obtained with the LES-informed RANS approach and a multi-fidelity strategy. A final chapter summarises the main conclusions and perspectives of this PhD thesis work.

Chapter II

Reduced-order modelling approach using machine learning

A direct numerical approximation of a full-order physical model is not affordable in the multi-query context of parameterised LES, where different scenarios associated with multiple sets of input parameters must be considered to quantify the range of all possible scenarios. This chapter introduces the reduced-order modelling approach we adopt in this work, and its different components required to emulate the LES solution statistics that are of very high dimension and that can have a nonlinear relationship with the uncertain input parameters. This is done in a standard statistical learning framework such as the one introduced by Guo and Hesthaven [2018], where the learning stage corresponds to the construction and evaluation of the reduced-order model, including a dimensionality reduction component and a regression model component. This training can be done offline. Later, multi-query evaluations of the reduced-order model can be done online and provide a large ensemble of predictions at a testing stage. We focus next on the learning stage.

Contents

II.1 Principle of a reduced-basis approach

The purpose of reduced-basis approaches is to provide statistically efficient approximate solutions of the full-order model, decreasing the computational burden while minimising the loss of accuracy. In this work, we aim at learning the mapping between the space of the uncertain input parameters $\boldsymbol{\mu} \in \mathbb{R}^d$ and the LES model response $\mathbf{K}_{\text{les}} \in \mathbb{R}^{N_h}$. It is worth noting that we do not consider the time dimension and that we only focus on the time-averaged LES statistics in this work, meaning that N_h represents the LES mesh dimension. The reduced-order model builds an efficient representation of a dataset of high-dimensional full-order LES snapshots; that is, a reduced-basis space of representation described by reduced-basis functions (or modes):

$$\mathcal{V}_{\text{rb}} = \text{Span}(\{\boldsymbol{\psi}_l\}_{l=1,\dots,L}) \subset \mathbb{R}^{N_h}. \tag{II.1}$$

The space $\mathcal{V}_{\text{rb}}$ is assumed to be of low dimension compared to the number of grid elements (i.e. $L \ll N_h$). Reduced-basis solutions, denoted by $\mathbf{K}_{\text{rb}}$, consist of the projected $\mathbf{K}_{\text{les}}$ fields on $\mathcal{V}_{\text{rb}}$. If the projection stands as a linear operator, $\mathbf{K}_{\text{rb}}$ can be expressed as:

$$\begin{aligned} \mathbf{G}_{\text{rb}} \colon \mathcal{P} &\longrightarrow \mathcal{V}_{\text{rb}} \\ \boldsymbol{\mu} &\longmapsto \mathbf{K}_{\text{rb}} = \sum_{l=1}^{L} k_l(\boldsymbol{\mu})\, \boldsymbol{\psi}_l(\boldsymbol{x})\,, \end{aligned} \tag{II.2}$$

where $\{\boldsymbol{\psi}_l\}_{l=1,\dots,L}$ correspond to the modes and $k_l(\boldsymbol{\mu}) = \langle \mathbf{K}_{\text{les}}, \psi_l(\boldsymbol{x}) \rangle \in \mathbb{R}$ is the lth reduced coefficient (or component).

This approach provides a procedure to split the parametric dependency in $\boldsymbol{\mu}$ from the spatial dimension $\boldsymbol{x}$ since local dissimilarity is now carried by the modes. In the literature, various methods have been used to extract the components $\boldsymbol{k} = (k_l)_l$ and modes $\boldsymbol{\psi} = (\psi_l)_l$ starting from the well-known POD approach to the most recent neural-network autoencoder architectures. It is worth noting that Eq. (II.2) is actually valid for a linear decomposition but does not rigorously extend to nonlinear methods such as nonlinear autoencoder. In the nonlinear context, the

decomposition can be expressed in a more general manner as the following composition:

$$\mathbf{K}_{\text{rb}} = f_d \circ f_e(\boldsymbol{\mu}), \tag{II.3}$$

where the encoding step (or compression) stands as $\boldsymbol{k}(\boldsymbol{\mu}) = f_e(\boldsymbol{\mu})$, and the decoding step (or decompression) stands as $\mathbf{K}_{\text{rb}} = f_d(\boldsymbol{k})$. For complex nonlinear functions f_d and f_e, it is not always possible to recover analytical formulations of the underlying modes.

The reduced-basis approach transforms LES data, i.e. projects the LES snapshots in a reduced space (or latent space) that is spanned by a set of parameter-independent functions $\{\boldsymbol{\psi}_l\}_{l=1,\cdots,L}$. They thereby return discrete reduced-basis coefficients from the original output fields (also referred to as the latent variables in the deep-learning community). Once the reduced basis is identified, regression metamodels (among whom polynomial chaos, Gaussian processes and decision trees) can be trained to map the uncertain parameters $\boldsymbol{\mu}$ onto the reduced coefficients $\{k_l\}_{l=1,\cdots,L}$. The resulting reduced-order model (Eq. II.2) can then be used inline to estimate new quantities of interest $\mathbf{K}_{\text{rb}}$ at new parameter values $\boldsymbol{\mu}^*$ (i.e. at parameter values that are not included in the LES training database).

The issues in building the reduced-order model for LES fields addressed in this work are three-fold: *i)* the quantities of interest simulated using LES are of very large dimension (N_h is on the order of 10^5 in this work); *ii)* the number of LES snapshots is limited due to the computational cost of a single LES (i.e. $N \ll N_h$); and *iii)* the mapping between the quantities of interest $\mathbf{K}$ and the input parameters $\boldsymbol{\mu}$ may be subject to nonlinearity. Additional difficulties associated with LES such as flow unsteadiness or the possibly large dimension of the input uncertainty space are beyond the scope of this work. In the following, Sect. II.2 introduces the main approaches implemented in this work for dimension reduction, while Sect. II.3 presents the metamodelling approaches to solve the regression problems before Sect. II.4 provides an overview of the proposed reduced-order model architecture.

II.2 Some dimensionality reduction methods

II.2.1 Proper orthogonal decomposition

II.2.1.a Snapshot dataset

In practice, POD [Sirovich, 1987; Berkooz et al., 1993] computes an estimate of the optimal solution $\mathcal{V}^*_{\text{rb}}$ from a collection of LES snapshots gathered in the snapshot matrix:

$$\mathbf{S} = \left[\mathbf{K}^{(1)}_{\text{les}} \mid \ldots \mid \mathbf{K}^{(N)}_{\text{les}}\right] \in \mathbb{R}^{N_h \times N}, \tag{II.4}$$

where $\mathbf{K}^{(n)}_{\text{les}}$ represents the nth LES snapshot, i.e. a vector made of N_h grid elements. POD seeks the optimal reduced basis of rank L that stands as the optimal orthogonal projection manifold

with respect to the Frobenius norm:

$$\mathcal{V}_{\mathrm{rb}}^{*} = \underset{\substack{\mathbf{S}_{\mathrm{rb}} = p(\mathbf{S};\mathcal{V}_{\mathrm{rb}}),\\ \mathrm{rank}(\mathcal{V}_{\mathrm{rb}}) \leq L}}{\arg\min} \|\mathbf{S} - \mathbf{S}_{\mathrm{rb}}\|_{\mathrm{F}}, \tag{II.5}$$

with $p(\,\cdot\,;\mathcal{V}_{\mathrm{rb}})$ the projection from $\mathbb{R}^{N_h}$ to $\mathcal{V}_{\mathrm{rb}}$.

The idea behind POD is to find an orthonormal basis maximising the variance of the projected field ensemble. Therefore, POD is usually implemented on the centred snapshot matrix. Let $\mathcal{T}(\,\cdot\,)$ be the linear operator applying an affine transformation (centering and normalisation) to the snapshot data:

$$\begin{aligned} \mathcal{T}\colon \mathbb{R}^{N_h} &\longrightarrow \mathbb{R}^{N_h} \\ \mathbf{K}_{\mathrm{les}} &\longmapsto \frac{1}{\sqrt{N-1}}(\mathbf{K}_{\mathrm{les}} - \hat{\mathbb{E}}(\mathbf{K}_{\mathrm{les}})), \end{aligned} \tag{II.6}$$

where $\hat{\mathbb{E}}(\mathbf{K}_{\mathrm{les}}) = \left[\hat{\mathbb{E}}(\mathbf{K}_{\mathrm{les},1}),\ \dots\ ,\ \hat{\mathbb{E}}(\mathbf{K}_{\mathrm{les},N_h})\right] \in \mathbb{R}^{N_h}$ is the empirical mean of the quantity of interest computed over the N snapshots for each grid element, i.e. for the ith grid element:

$$\hat{\mathbb{E}}(\mathbf{K}_{\mathrm{les},i}) = \frac{1}{N}\sum_{n=1}^{N} K_i^{(n)}. \tag{II.7}$$

In the following, $\mathcal{T}(\mathbf{S})$ refers to the matrix of transformed snapshots $\left(\mathcal{T}(\mathbf{K}_{\mathrm{les}}^{(\mathrm{n})})\right)_n$.

II.2.1.b Eigendecomposition and interpretation

POD may now solve the diagonalisation of the covariance matrix:

$$\begin{aligned} &\mathrm{Cov}(\mathbf{K}_{\mathrm{les}}, \mathbf{K}_{\mathrm{les}}) = \mathcal{T}(\mathbf{S})\,\mathcal{T}(\mathbf{S})^T = \Psi\,\Sigma^2\,\Psi^T, \\ &\text{with } \begin{cases} \mathrm{Cov}(\mathbf{K}_{\mathrm{les}}, \mathbf{K}_{\mathrm{les}}) \in \mathbb{R}^{N_h \times N_h} \\ \Psi = [\boldsymbol{\psi}_1 \mid \dots \mid \boldsymbol{\psi}_{N_h}] \in \mathbb{R}^{N_h \times N_h} \text{ an orthonormal matrix} \\ \Sigma = \mathrm{diag}(\sigma_1, \dots, \sigma_{N_h}) \in \mathbb{R}_{+}^{N_h \times N_h},\ \sigma_1 \geq \dots \geq \sigma_{N_h} > 0. \end{cases} \end{aligned} \tag{II.8}$$

Orthonormal vectors of Ψ, called the reduced-basis modes, carry some fraction of the ensemble variance quantified by the related eigenvalues in Σ denoted by $\{\sigma_l\}_{l=1,\cdots,N_h}$ and satisfying $\mathcal{T}(\mathbf{S})\,\mathcal{T}(\mathbf{S})^T\,\boldsymbol{\psi}_l = \sigma_l\,\boldsymbol{\psi}_l$.

Since the number of mesh elements N_h is very large, it becomes advantageous to keep the L first modes (among the N_h modes) that preserve the maximum variance of the original ensemble. The resulting truncated matrices are denoted by $\widetilde{\Psi} = [\boldsymbol{\psi}_1 \mid \dots \mid \boldsymbol{\psi}_L] \in \mathbb{R}^{N_h \times L}$ and $\widetilde{\Sigma} = \mathrm{diag}(\sigma_1, \dots, \sigma_L) \in \mathbb{R}^{L \times L}$. It is worth noting that the number of modes L to retain to obtain an accurate reduced-order model is problem-dependent, and that this is an open question for the problem of microscale pollutant dispersion that is addressed in this work.

POD acts as a change of basis and the meaning of each mode can be interpreted by a correlation analysis between the modes and the original features. In practical terms, the correlation between the lth POD mode $\boldsymbol{\psi}_l$ and a given snapshot in the initial grid space $\mathbf{K}_{\mathrm{les}}$ (varying

between -1 and 1 by definition) may be stated as the following matrix:

$$\mathrm{Corr}(\mathbf{K}_{\mathrm{les}}, \boldsymbol{\psi}_l) = \left[\sqrt{\frac{\sigma_l}{\hat{\mathbb{V}}(\mathbf{K}_{\mathrm{les},i})}} \psi_{l,i} \right]_{i,j}, \tag{II.9}$$

where the indices i and j correspond to a given element of the matrix $\mathrm{Corr}(\mathbf{K}_{\mathrm{les}}, \boldsymbol{\psi}_l) \in \mathbb{R}^{N_h \times N_h}$, where $\psi_{l,j}$ corresponds to the jth element of $\boldsymbol{\psi}_l \in \mathbb{R}^{N_h}$, the lth function in $\widetilde{\Psi}$, and where $\hat{\mathbb{V}}(\mathbf{K}_{\mathrm{les},i})$ represents the variance unbiased estimation over the snapshots for the ith grid element. This variance is estimated as:

$$\hat{\mathbb{V}}(\mathbf{K}_{\mathrm{les},i}) = \frac{1}{N-1} \sum_{n=1}^{N} \left(K_{\mathrm{les},i}^{(n)} - \hat{\mathbb{E}}(\mathbf{K}_{\mathrm{les},i}) \right)^2. \tag{II.10}$$

where the mean is defined in Eq. (II.7).

II.2.1.c Performance metrics

As the POD performs a projection subspace maximising the ensemble variance of the original data, an intuitive measure of its performance is the cumulative variance carried by the first L modes, i.e. the variance of the projected samples on the L-dimensional POD subspace to the total variance of the original data. The explained variance ratio is expressed as:

$$Q^2_{\mathrm{e.v.}} = \sum_{i=1}^{l} \sigma_i \Big/ \sum_{i=1}^{\min(N_h,N)} \sigma_i, \tag{II.11}$$

where σ_i denotes the ith eigenvalue of the data covariance matrix.

II.2.1.d Reduced-coefficient dataset

The quantity of interest vector field can be projected upon the POD space. The projection can be expressed as:

$$\begin{aligned} \mathcal{F} \colon \mathbb{R}^{N_h} &\longrightarrow \mathbb{R}^L \\ \mathbf{K}_{\mathrm{les}} &\longmapsto \boldsymbol{k} = \widetilde{\Sigma}^{-1/2}\, \widetilde{\Psi}^T\, \mathcal{T}(\mathbf{K}_{\mathrm{les}}), \end{aligned} \tag{II.12}$$

where $\widetilde{\Sigma}^{-1/2} = \mathrm{diag}(1/\sqrt{\sigma_1}, \ldots, 1/\sqrt{\sigma_{N_h}}) \in \mathbb{R}^{L \times L}$ and $\widetilde{\Psi} \in \mathbb{R}^{N_h \times L}$ are the matrices restricted to the L first modes, and where $\boldsymbol{k} = [k_1, \cdots, k_L] \in \mathbb{R}^L$ is the vector of POD reduced coefficients (Eq. II.2).

In complement, whitening (or sphering) can be implemented from POD (e.g. Kessy et al., 2018). It consists in centring and standardising the reduced coefficients obtained in Eq. (II.12). The elements in $\boldsymbol{k}$ have therefore the following statistical properties:

$$\begin{cases} \mathbb{E}[k_l] = 0 \quad \forall\, l = 1, \ldots, L\,, \\ \mathbb{E}[k_l\, k_m] = \begin{cases} 1 \text{ if } l = m, \\ 0 \text{ otherwise.} \end{cases} \qquad \forall\, l, m = 1, \ldots, L. \end{cases} \tag{II.13}$$

The transformation in Eqs. (II.12)–(II.13) is applied to the original LES dataset (expressed in the grid space) to create a new dataset of POD reduced coefficients $\boldsymbol{k}$ for varying inputs $\boldsymbol{\mu}$. In Chapters IV–V, the statistical properties given to the POD reduced coefficients will be used as constraints for designing consistent mappings $f_l : \boldsymbol{\mu} \mapsto k_l$ such that:

$$\hat{\mathbb{E}}[f_l] = \hat{\mathbb{E}}[k_l] = 0\,, \qquad \hat{\mathbb{V}}(f_l) = \hat{\mathbb{V}}(k_l) = 1\,, \qquad \forall l. \tag{II.14}$$

II.2.1.e Inverse reconstruction

The compressed information obtained through Eqs. (II.12)–(II.13) can be used to design efficient mappings of $\boldsymbol{\mu} \mapsto \boldsymbol{k}$ since the dimension of $\boldsymbol{k}$ is small compared to the dimension of $\mathbf{K}_{\text{les}}$ (Sect. II.3). However, in practice, there is a need to map back the reduced coefficients onto the physical space to have access to the LES fields of interest (for instance, to validate the POD approach by comparing reconstructed fields and LES fields of reference).

The linearity in the operator $\mathcal{T}$ in Eq. (II.12) makes it simple to express the inverse reconstruction operator from which the original LES field of interest may be recovered from the POD reduced coefficients:

$$\mathbf{K}_{\text{rb}}(\boldsymbol{\mu}) = \mathcal{T}^{-1}\left(\sum_{l=1}^{L} \sqrt{\sigma_l}\, k_l\, \boldsymbol{\psi}_l\right). \tag{II.15}$$

POD stands as a linear combination of the snapshots (Eq. II.2), which can be a limitation when the response of LES fields features strong nonlinearities. If this is the case, neural-network autoencoders stand as an alternative and can be seen as a nonlinear extension to POD.

II.2.2 Autoencoder neural networks

Autoencoder neural networks can in theory deal with nonlinearities in the LES dataset to provide highly compressed reduced coefficients $\boldsymbol{k}$. In this work, we are particularly interested in convolutional autoencoders (neural networks composed by convolutional layers) since they have great potential to compress information in problems of high dimension and subject to nonlinearities.

II.2.2.a Introduction to neural networks

In this section, we briefly recall the basic principles of neural networks to define what is a neural network and highlight the advantages of convolutional autoencoders.

Neuron definition. A (linear) neuron can be described as a mathematical model, based on an analogy with biological neurons. It stands as a linear model with respect to some inputs $\boldsymbol{u}$:

$$s = f(\boldsymbol{u}) = \lambda_0 + \sum_{j=1}^{b} \lambda_j\, u_j. \tag{II.16}$$

The neuron output s is given from the linear combination of the inputs $\boldsymbol{u} \in \mathbb{R}^b$, with $\boldsymbol{\lambda} = \{\lambda_j\}_{j=1,\cdots,b}$ the associated weights. Considered individually, a neuron can be seen as a basic machine learning model, whose weights must be trained to predict some output $\boldsymbol{s}$. Its individual

predictive capacity is weak as it is a linear model. Combining neurons in the form of a neural network becomes interesting to approximate complex mapping.

Neural-network definition. A neural network is the graph of interconnected neurons. It is specified by the graph architecture (number of layers, number of neurons per layer/width, connections), the type of neurons, the learning task (e.g. supervised/unsupervised), etc.

Figure II.1 shows a simple example of a multilayer perceptron made of three layers. The inputs of the learning problem are collected in the input layer and then sequentially transformed as they run through the network hidden (intermediate) layer and the output layer. Hidden layer neuron outputs are referred to as internal states and are used as inputs of the next layer neurons. This network is qualified as feed-forward since information linearly propagate through the layers from the input to the output, and since connections between neurons only occur between successive layers. It is also qualified as dense since all neuron outputs in each layer are connected to the next layer neuron inputs.

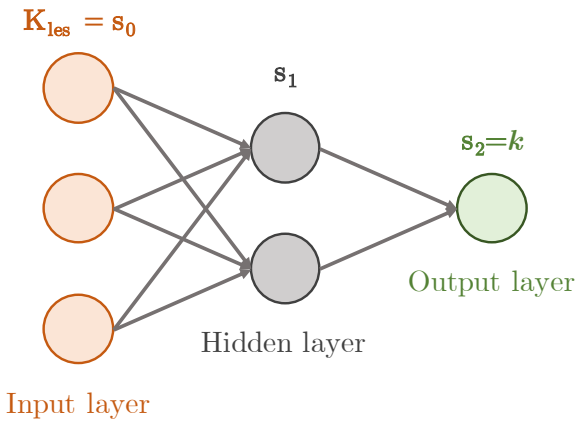

Figure II.1: A simple feed-forward multilayer perceptron made of three layers, each layer i being described by its internal state s_i.

Mathematically, in line with Eq. (II.16), network outputs can be expressed as:

$$k = f_{\text{out}}(\boldsymbol{s}_1) = \lambda_{\text{out},0} + \sum_{j=1}^{2} \lambda_{\text{out},j}\, s_{1,j}, \tag{II.17}$$

where $\boldsymbol{s}_1 = (s_{1,1}, s_{1,2})$ is the output vector from the hidden layer, and $\boldsymbol{\lambda}_{\text{out}} = (\lambda_{\text{out},0}, \lambda_{\text{out},1}, \lambda_{\text{out},2})$ includes the weights of the output neuron. The full neural network can then be expressed as the function composition of all successive layers $\boldsymbol{k} = \mathcal{F}_e(\mathbf{K}_{\text{les}}) = f_{\text{out}} \circ f_{\text{hid}}(\mathbf{K}_{\text{les}})$.

Multilayer perceptron autoencoder. The simple structure of the multilayer perceptron presented in Fig. II.1 can be used for unsupervised learning. For instance, the operator $\mathcal{F}_e$ can be trained to map high-dimensional LES snapshots $\mathbf{K}_{\text{les}}$ to lower-dimensional reduced coefficients $\boldsymbol{k}$ when the output layer width is small. This network may be expanded with additional layers to return the original high-dimensional data $\mathbf{K}_{\text{les}}$ from the reduced coefficients $\boldsymbol{k}$ through the operator $\mathcal{F}_d$ (Fig. II.2). The vector of reduced coefficients $\boldsymbol{k}$ defines the low-dimensional latent space.

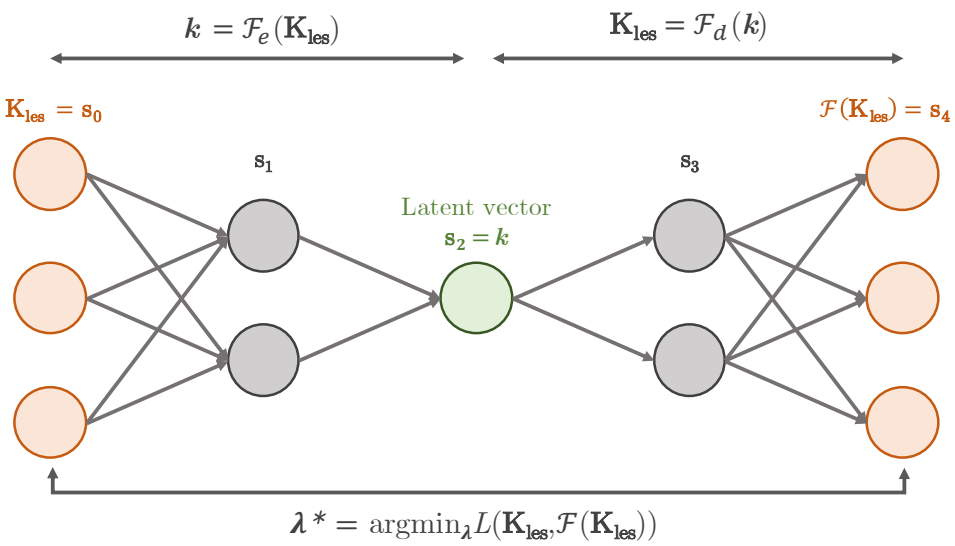

Figure II.2: Example of a multilayer perceptron autoencoder neural network.

The multilayer perceptron autoencoder has a bottleneck architecture. The first encoder subnetwork $\mathcal{F}_e$ maps the input field $\mathbf{K}_{\text{les}}$ onto the latent space. Then, the second decoder subnetwork expands the reduced coefficients $\boldsymbol{k}$ back to the original high-dimensional space. The loss function $\mathcal{L}$ (e.g. the mean-squared error) is the metric used to estimate the reconstruction error of the autoencoder. Minimising the loss function encourages the network $\mathcal{F}$ to output the same LES snapshots that were given as inputs such that $\mathbf{K}_{\text{les}} \approx \mathcal{F}(\mathbf{K}_{\text{les}}; \boldsymbol{\lambda})$ while restraining the number of reduced coefficients $\boldsymbol{k}$. $\boldsymbol{\lambda}$ represent the autoencoder parameters (e.g. the neuron weights) to be calibrated. Contrary to POD, in such a neural network framework, all encoder, decoder and latent space are learnt simultaneously during the training stage.

It is known that multilayer perceptron autoencoders may fail when the input data are large. They are in fact designed for vectors with dense connectivity, meaning that each output unit interacts with each input unit through matrix multiplication. This leads to a very high number of connections and weights in the network, which may make model optimisation intractable.

II.2.2.b Convolutional neural networks

Neural networks are of high interest for image processing because their architecture is relatively flexible and can be adapted to only capture the correlations between neighbouring pixels (in an image, the closer the pixels are, the more likely they are to be correlated). In particular, convolutional layers are of high interest to focus on the most relevant connections.

Convolutional layer. A convolutional neural network (CNN) operates directly on matrices or even tensors. The core convolutional layer involves the convolution between an input tensor and a multidimensional kernel. Mathematically, convolution (also known as cross-correlation, see Goodfellow et al. 2016) can be expressed as a linear combination between the inputs and some kernel weights:

$$s_i = \lambda_0 + \sum_{j=1}^{n_k} \lambda_j \times u_{i-1+j}, \tag{II.18}$$

where s_i is the ith pixel of the output vector, u_i is the ith pixel of the input vector, and $(\lambda_j)_{j=0,\dots,n_k}$ are the kernel weights. Figure II.3 presents a simple example of convolution with $n_k = 3$.

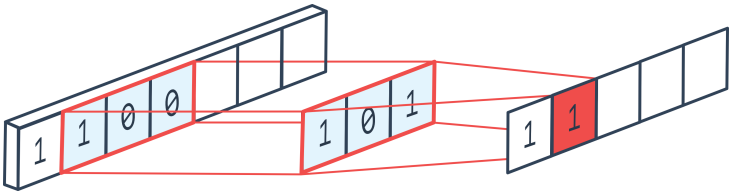

Figure II.3: Example of convolution of a 1-D input array with a kernel of size $n_k = 3$. The output in red is obtained by multiplying each kernel weight by the corresponding input pixel and by returning the sum of the products [pel, 2022].

Convolutional layers actually make use of three concepts that might improve image processing: (*i*) sparse interactions, (*ii*) parameter sharing, and (*iii*) equivariant representations. Equation (II.18) shows that the linear combination involves sparse connectivity. Stated differently, the discrete convolution may be thought of as a sparse matrix multiplication: the output features only depend on a few input features, and the same kernel is employed throughout the convolution (parameter sharing), resulting in only four weights in this case. Long-range correlations are neglected due to the convolution properties. Kernels are therefore suitable to learn local redundant patterns. It does not matter where the pattern appears in the image; convolution will return the same output each time it occurs. This means that convolutional layers make the network equivariant to translation. These properties of convolutional layers reduce the number of parameters in the neural network. This reduces the memory and computing requirements of the autoencoder, and improves its statistical efficiency compared to multilayer perceptron [Goodfellow et al., 2016].

Convolutional autoencoder. So far we have discussed the case of a convolutional layer with a single kernel. This is rather restrictive, given that one kernel learns one unique redundant pattern and that an image generally embeds several patterns. It is therefore of primary importance to have several kernels in the CNN to well characterise the information in the fields of interest. For instance, let us consider the LES snapshots $\mathbf{K}_{\text{les}}$ have been reshaped to a regular grid and are now matrix elements of $\mathbb{R}^{I,J}$. Each snapshot is associated with several quantities (also named filters or channels) such as the tracer concentration and the velocity field, implying that each snapshot is now a 3-D tensor evolving in $\mathbb{R}^{C,I,J}$. The resulting convolutional layer applied to a multi-channel tensor can still be expressed as a linear combination of the inputs with kernels of size $H \times H$:

$$s_{c\,i\,j} = \lambda_0 + \sum_{k=1}^{C(\boldsymbol{u})} \sum_{m=1}^{H} \sum_{n=1}^{H} \lambda_{c\,k\,m\,n}\, u_{k\,i+m-\mathrm{F}\,j+n-\mathrm{F}}\,, \tag{II.19}$$

where $s_{c\,i\,j}$ is the output filter $c \in C(\boldsymbol{s})$ at pixel location (i, j). The index shift $\mathrm{F} = \lfloor H/2 \rfloor$ characterises the image padding, ensuring that convolution does not extend beyond the image's range. Figure II.4 illustrates the tensor convolution operation.

By combining convolutional layers and a multilayer perceptron, a convolutional autoencoder can be designed to handle high-dimensional data as in Fig. II.5. Convolutional layers are added

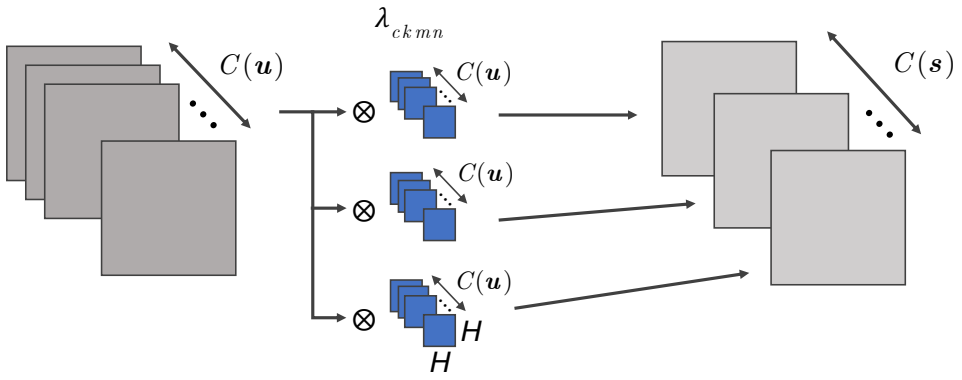

Figure II.4: Schematic representation of a multi-channel convolution [Fukami et al., 2020]. Kernels are represented in blue squares. Image tensors are represented in grey colours.

on both sides of a central multilayer perceptron to drastically reduce the number of connections in the network even if high-dimensional multi-filter images are given as inputs. First, the encoder deals with spatial correlations using convolutional layers. Dimension reduction can be performed using downsampling operations such as stride or pooling [Boureau et al., 2010; Jia et al., 2012; Goodfellow et al., 2016]. A multilayer perceptron is inserted to deal with the compressed tensors and reduce even further the latent space. Then, the reduced components $\mathbf{k}$ are expanded back to the shape of the original input tensor. Upsampling can easily be performed by replicating tensor pixels between convolutional layers [Goodfellow et al., 2016].

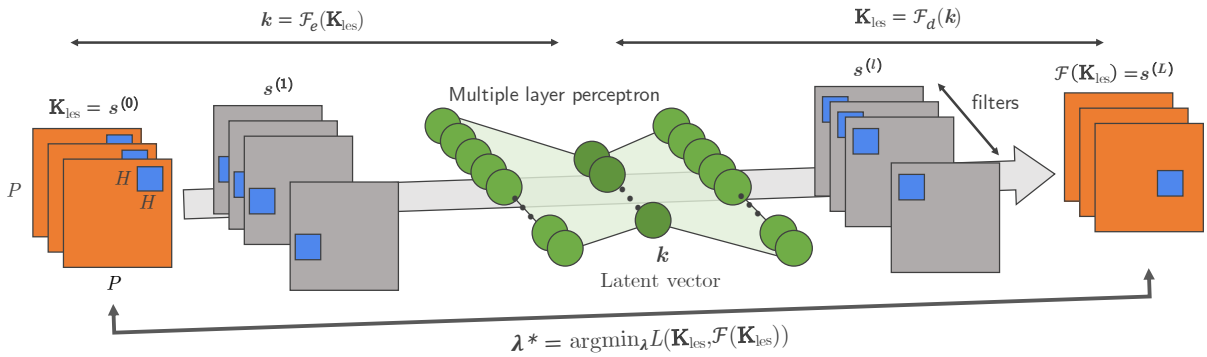

Figure II.5: Schematic view of a convolutional autoencoder involving a central multilayer perceptron and convolutional layers in the encoder and decoder parts [Fukami et al., 2020].

Despite its complex architecture, this convolutional autoencoder is still linear in the input data and it is therefore currently unable to return richer information than POD. To extend the capability of autoencoders to nonlinear problems, the choice of the activation functions is essential.

Nonlinear activation functions. Nonlinear activation functions can be introduced to wrap neuron outputs:

$$s = g\left(\lambda_0 + \sum_{j=1}^{b} \lambda_j \, u_j\right). \tag{II.20}$$

The choice for the function g depends on the task at hand. There are several options among whom rectified linear unit (ReLU), sigmoid logistic function, hyperbolic tangent (Tanh) and the

softplus function (Table II.1). In the following, we discuss the advantages and drawbacks of each activation function, and how they can be used to design a proper autoencoder.

Table II.1: Examples of nonlinear activation functions.

Name	Equation	Range
Rectified Linear Unit (ReLU)	$x \mapsto x_+$	$[0, +\infty[$
Sigmoid logistic function	$x \mapsto \frac{1}{1+e^{-x}}$	$]0, 1[$
Hyperbolic tangent (Tanh)	$x \mapsto \frac{2}{1+e^{-2x}} - 1$	$]-1, 1[$
Softplus	$x \mapsto \log(1 + e^x)$	$[0, +\infty[$

Adding nonlinear activation functions to the network should be done with caution since it affects the gradient descent procedure required to calibrate the network weights during the training stage. The choice of the activation functions is closely linked with the answer of the two following conditions: (i) Does the output range fit the problem? (ii) Is learning the network weights more difficult?

– The first issue is quite easy to solve as the discussion on the output range mainly concerns the network output layer. For instance, the ReLU and softplus functions may be appropriate for modelling tracer concentration as they provide positive outputs. The sigmoid function returns normalised outputs in the range $[0, 1]$ and may be applied to probability distributions.

– The choice of the activation function for the hidden layers is more critical for optimisation purposes, making the second issue more tricky to solve. One must ensure the convergence of neural weights towards a sufficient optimum. Gradient backpropagation is a simple and computationally-efficient method. However, it may be slow to converge, converge to a non-satisfying solution, or convergence may not occur at all, particularly when dealing with large multilayer networks and non-convex high-dimensional response surfaces. Several issues may arise such as node saturation or vanishing gradient. A naive choice of activation functions can reduce backpropagation performance and increase convergence time [LeCun et al., 2012]. Properly configuring the network activation functions and following good practices for (a) weight initialisation, (b) data normalisation, and (c) step decay scheme may improve the gradient descent procedure.

Historically, the sigmoid function was the most commonly used activation function for simplicity and explicability: it is monotonically increasing and differentiable, with bounded positive output values in the range of $[0, 1]$, which prevents optimisation divergence. However, the Tanh function is often preferred to the sigmoid function as it may lead to faster convergence [LeCun et al., 2012]. This phenomenon can be attributed to the symmetry property of Tanh and the stronger magnitude of its gradient (Fig. II.6). LeCun et al. [1989] also suggested that input data should be centered and standardised, and that the weights should be initialised around 0 to benefit from better descent directions and larger steps, and thereby avoid gradient saturation.

Both sigmoid and Tanh functions suffer from the vanishing gradient problem, which may hamper gradient descent in the case of very deep networks [Hochreiter, 1998; Glorot and Bengio,

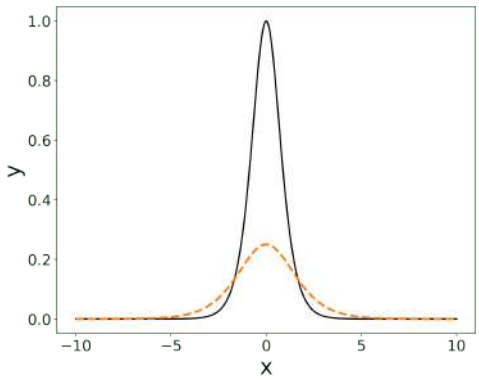

Figure II.6: Gradient of the sigmoid logistic (dotted line) and Tanh (solid line) functions.

2010]. This means that the error decreases after it is backpropagated through each hidden layer, resulting in very slow updates of the weights. The ReLU function (and extensions such as the leaky ReLU) can fix this issue. ReLU can also enhance network sparsity by removing unnecessary nodes, which facilitates its training [Glorot et al., 2011].

It is worth noting that weight initialisation is not straightforward for the ReLU function. Weights are usually initialised randomly. Glorot and Bengio [2010] highlighted the benefits of normalising initial weights with respect to the layer widths so that the output variance remains the same for each layer in the network. They also suggested this could be applied to other activation functions to reduce vanishing gradient effects. This is line with the work by Kumar [2017], which suggested to initialise weight distributions from the activation function and the number of neurons in the layers.

II.2.2.c Training large nonlinear neural networks

For large deep neural network architectures, the number of parameters to optimise can become substantial, and the good practices stated so far (data normalisation, choice of activation functions, weight initialisation) are not always sufficient to ensure good gradient descent convergence. In addition to the node saturation and vanishing gradient issues, the loss function can be made highly irregular by the nonlinear network structure. Some techniques can help regularising high-dimensional loss functions. For instance, network sparsity can be enhanced by adding ReLU activation functions or dropout layers [Srivastava et al., 2014]. A penalty term can also be added to the loss function [Tibshirani, 1996; Kavukcuoglu et al., 2010; Rifai et al., 2011].

Adaptive learning rate schemes and stochastic mini-batch extensions of gradient descent can also speedup the optimisation procedure and prevent it from adhering to an unfavourable local optimum [Heskes and Kappen, 1993; Orr, 1995; Bottou et al., 1998]. In particular, adaptive learning rate schemes will reduce the learning rate when the gradient direction oscillates, and increase it when the gradient direction remains steady across epochs [Sompolinsky, 1995; Sutton, 1992]. For instance, it seems intuitively reasonable to make the learning rate large at the beginning of the procedure to achieve faster convergence, and decrease it to ensure finer tuning and converge to a local solution. When gradient directions are very chaotic due to significant noise, a momentum term can be added to attenuate the gradient direction oscillations and

to smooth the descent procedure [Polyak, 1964; Tseng, 1998]. A more recent version of the momentum approach referred to as the Nesterov accelerated gradient is due to Nesterov [1983] and Sutskever et al. [2013]. Sophisticated adaptive algorithms such as resilient backpropagation (RProp) [Riedmiller and Braun, 1993] involving the sign of the gradient, root mean square propagation (RMSProp) as a mini-batch version of the RProp [Tieleman et al., 2012], adaptive gradient (Adagrad) involving different learning rates for the weights [Duchi et al., 2011] or adaptive moment estimation (Adam) as an update of RMSProp [Kingma et al., 2020] are now widely used by the deep-learning community for training neural networks.

POD and autoencoders can compress high-dimensional data to a smaller latent space of reduced-basis coefficients. The latent space representation is learned along with the encoder and decoder models, which map the high-dimensional data to the compressed coefficients and vice versa. The convolutional autoencoder allows for efficient processing of high-dimensional images and can handle nonlinearities by introducing activation functions. It is therefore an attractive extension to POD. However, POD ensures statistical properties to the reduced coefficients, and guarantees a hierarchical decomposition of the information. Each approach has its own advantages and drawbacks in terms of performance and explicability. Both POD and autoencoders are tested in this work to evaluate their capacity to provide a good latent space representation for microscale pollutant dispersion problems.

II.3 Regression models

Rather than constructing a cumbersome metamodelling procedure involving the high-dimensional LES fields $\mathbf{K}_{\text{les}}$, we prefer to train metamodels to predict the reduced coefficients $\boldsymbol{k} = [k_1, \cdots, k_L]$ introduced in the previous sections from the input parameters $\boldsymbol{\mu}$. Various learning algorithms are available to solve this regression task. For instance, polynomial chaos expansion [García-Sánchez et al., 2014, 2017; El Garroussi et al., 2022], Gaussian process regression (GPR)/kriging models [Margheri and Sagaut, 2016; Guo and Hesthaven, 2018; Xiao et al., 2019], and decision trees [Xiang et al., 2021] have shown promising results in recent CFD literature. This section discusses their mathematical formulation and the hyperparameters to be tuned for the more specific task of predicting reduced basis coefficients.

II.3.1 Overview of the metamodelling task

II.3.1.a Metamodel formulation

Dimension reduction approaches rely on ensemble variance decomposition. Each latent variable carries specific energetic structures of data, and the characteristics of the latent variables can be very different from each other. For instance, POD decomposes ensemble variance into hierarchical information carried by the L reduced coefficients: the first modes carry the large energetic structures contained in the data, whereas the higher modes focus more and more on local effects. Moreover, parametric variability induces strong changes in the modal decomposition distribution. For instance, a change in the tracer emission position drastically affects the

amplitude and/or the signs of the reduced coefficients. As a result, the associated response surfaces $\boldsymbol{k} \equiv \boldsymbol{k}(\boldsymbol{\mu})$ tend to reflect very different characteristic scales of the tracer concentration patterns.

It is essential to design metamodels capable of capturing the specific nature of each latent variable in order to accurately and robustly describe the possible variation of the reduced coefficients. For this reason, in this work, we design L independent metamodels, i.e. we learn the relation between each reduced coefficient k_l and the input parameters $\boldsymbol{\mu} = (\mu_1, \dots, \mu_d)$:

$$k_l = f_l(\boldsymbol{\mu}) + \epsilon_l, \quad \forall l = 1, \cdots, L, \tag{II.21}$$

where $\epsilon_l \sim \mathcal{N}(0, s_l^2)$ is an additive noise term with variance s_l^2 to account for noise in the training database. Each metamodel f_l is learnt by minimising the error with respect to a criterion $\mathcal{L}$ made on predictions of k_l. In order to obtain the best available approximation to the true response k_l, one can measure the loss $\mathcal{L}(k_l, f_l(\boldsymbol{\mu}; \boldsymbol{\lambda}_l))$ between the true reduced coefficient k_l associated with a given input $\boldsymbol{\mu}$ and the metamodel prediction $f_l(\boldsymbol{\mu}; \boldsymbol{\lambda}_l)$ ($\boldsymbol{\lambda}_l$ represent the learning model parameters that must be calibrated). The loss expected value is given by the following risk functional:

$$R(\boldsymbol{\lambda}_l) = \int \mathcal{L}(k_l, f(\boldsymbol{\mu}; \boldsymbol{\lambda}_l))\, dQ(\boldsymbol{\mu}, k_l), \tag{II.22}$$

where the different elements are defined as random variables to characterise their probability of occurrence, implying that $Q(\boldsymbol{\mu}, k_l)$ represents the joint probability distribution of the inputs $\boldsymbol{\mu}$ and the reduced coefficient k_l. In practice, the joint probability distribution Q is usually unknown and generally requires modelling assumptions. Finally, the objective is to find the function f minimising Eq. (II.22) over one or several classes (or families) of models f for a given probability distribution on the observations.

In this work, we are particularly interested in parametric metamodels built around combinations of inputs $\boldsymbol{\mu}$. This parametric nature refers to the requirement to estimate model parameters from the training data. For instance, a polynomial regression model assumes a polynomial relationship between the input features $\boldsymbol{\mu}$ and the output k_l. As an example, we could take the total electricity demand of the city of Toulouse $k \in \mathbb{R}$ as the sum of the polynomial features between the two major individual demands $(\mu_1, \mu_2) \in \mathbb{R}^2$. Mathematically, it is expressed as:

$$k = \lambda_0 + \lambda_1\, \mu_1 + \lambda_2\, \mu_2 + \lambda_3\, \mu_1^2 + \lambda_4\, \mu_2^2 + \lambda_5\, \mu_1 \mu_2 + \epsilon, \tag{II.23}$$

where combinations of inputs are taken with a total polynomial order $P = 2$. Usually, the model parameters are unknown a priori; the training stage has the objective to learn the parameter optimal values from data observations but not all model parameters can be optimised simultaneously. These non-optimisable parameters – the ones that must be determined outside the learning algorithm – are called hyperparameters. In the polynomial model example (Eq. II.23), $\{\lambda_0, \cdots, \lambda_5\}$ are model parameters; while the total polynomial order P is a hyperparameter ($P = 2$ in the example). In practice, the user will generally test several values of P to determine an appropriate model choice.

Each learning algorithm differs in how it handles data and expresses the map between inputs

$\boldsymbol{\mu}$ and the output k_l. A broad range of methods are provided in statistical learning [Hastie et al., 2009], with the linear Gaussian model being the most basic, oldest, and well-known statistical model. As soon as the map to be modelled between inputs and outputs is not linear or the volume of data is large, more advanced methods (e.g. polynomial chaos expansion, Gaussian processes, decision trees) may be required. In practice, there is no single best metamodelling strategy and the choice of the metamodelling approach depends on the problem specificities.

II.3.1.b Metamodel resolution

Dataset management. The dataset available for solving the lth metamodelling problem can be formulated as an ensemble of input/output pairs of N elements, i.e.

$$\mathcal{D}_l = (\mathcal{U}, \mathcal{K}_l) = \{(\boldsymbol{\mu}^{(n)}, k_l^{(n)}),\ 1 \leq n \leq N\}, \tag{II.24}$$

where the nth pair $(\boldsymbol{\mu}^{(n)}, k^{(n)})$ represents a snapshot in the reduced-coefficient database.

The main problem in metamodelling is to develop an unbiased assessment of the risk (Eq. II.22) based on limited LES data. The measure of model's error on the training data (the training error) may lead to an underestimation of the actual model's error. In practice, to avoid overfitting, we are interested in how well the learning model will perform on new independent data through the estimation of the generalisation error. For this purpose, the snapshots in the database $\mathcal{D}_l$ are generally split into two subsets: *i*) the training dataset for model learning and minimising the risk during the optimisation procedure, and *ii*) the test dataset for final model performance evaluation. In the following, we make the distinction between the training sample $(\mathcal{U}, \mathcal{K}_l)$ and the test sample $(\mathcal{U}^*, \mathcal{K}_l^*)$.

Error estimation. The objective in the metamodelling process is to find the statistical model f_l (i.e. to estimate the optimal parameters $\boldsymbol{\lambda}_l$) that minimise the risk R [Vapnik, 1999]. In practice, an empirical risk estimate can be obtained by evaluating the loss function across an ensemble of inputs in $\mathcal{D}_l$:

$$\widehat{R}(f_l, \mathcal{D}_l) = \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}\left(k_l^{(n)}, f_l(\boldsymbol{\mu}^{(n)})\right). \tag{II.25}$$

where the loss function $\mathcal{L}(k_l, f_l(\boldsymbol{\mu}))$ measures a distance between the reference output value k_l and its estimate $f_l(\boldsymbol{\mu})$. In the context of regression, losses are generally defined in L^p $(p \geq 1)$. The choice of a loss function introduces a bias to favour a specific desired behaviour of the learning model. For multidimensional data, the loss function measures the total distance between two points as the aggregation of the distance in each dimension. The choice of the loss is then inextricably linked to the choice of how to weight the distance in each dimension. For instance, in a regression task, a small value of p further penalises small prediction errors, whereas a large value of p further penalises large errors. This means that a small value of p aims primarily at having a small systematic error in the statistical model predictions, making possible to have wrong outlier predictions. In the opposite, when p is large (with the extreme case of the infinite norm measuring

the maximum error over the snapshots), the objective is to primarily reduce the model's error on the few most challenging snapshots with largest error. In low dimension, the Euclidean norm (associated with the L^2-space) is commonly used in machine learning frameworks. It provides a reasonable compromise between systematic errors and outliers, and its differentiability makes it appropriate for training learning algorithms.

II.3.1.c Metamodel generalisation capacity and selection

Standard statistical methods (e.g. polynomial model) generally fail in handling high-dimensional data with a small training dataset. Alternatives combining more advanced models (e.g. gradient boosting, neural network) may be developed but they may suffer from a lack of interpretability, while it may be an important criterion for the target problem. From a methodological point of view, it is essential to have a well-defined framework and protocol to compare metamodelling methods in order to select the most appropriate one for a given problem. In this process, there are two subquestions: (*i*) finding the best model structure (the best hyperparameters) for a given family (e.g. choosing the most suitable total polynomial order P for a polynomial regression model), and (*ii*) finding the best model family (or class) for a given problem (e.g. polynomial model, Gaussian processes, decision trees).

The training error does not reflect model's generalisation capacity on new independent snapshots. For instance, it is biased by model complexity. One can always find the polynomial of degree P that passes through the P snapshots of the database. The more sophisticated a model, the more flexible it is, allowing it to adjust to the data and generate a low adjustment error. However, such a model generalisation may fail during the prediction step when applied to new independent data. As a result, the test error (the error estimated on the unbiased test data) offers a suitable metric for evaluating the generalisation capacity of the statistical model and for answering questions (*i*) and (*ii*) stated above. The gap between the training error and the test error gives insights into model capacity and complexity. Overfitting occurs when the test error is larger than the training error, indicating that the model's complexity is too high and the patterns learned from the training snapshots do not generalise to new snapshots. Conversely, a training error that is larger than the test error is often a sign of underfitting, meaning that the model has difficulty representing the variability in the data and its complexity may be increased. For example, a linear model capacity may be expanded by including more explanatory variables in its formulation or increasing the total polynomial order in Eq. (II.23). Changing the learning model structure may leverage its balance towards data fitting (Fig. II.7). The trade-off between model complexity and goodness-of-fit is often referred to as bias-variance trade-off.

In the following, an overview of each model family is given and the key hyperparameters of each model family are highlighted. Section II.3.2 introduces how polynomial chaos expansion improves the flexibility of the linear model by integrating polynomial features of the inputs. Then, Sect. II.3.3 explains why the Gaussian process formalism is particularly well-suited to emulate a wide range of length-scales across the latent variables. And finally, Sect. II.3.4 presents how decision trees can be implemented to solve a regression problem and how the major problems of variance instability may be solved by using a boosting procedure. These different metamodelling

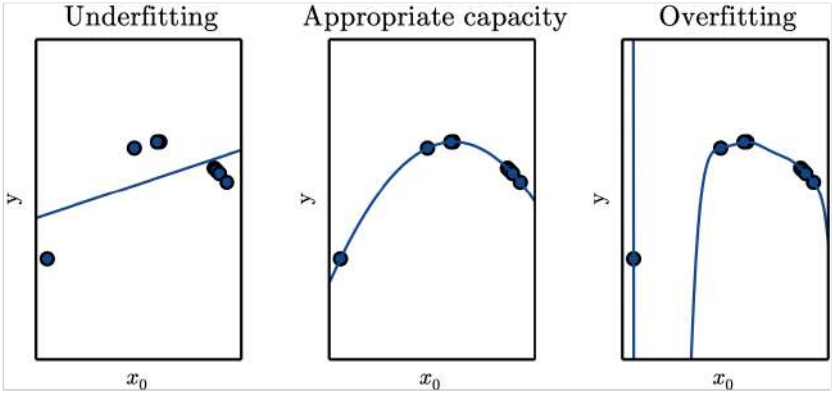

Figure II.7: Illustration of underfitting and overfitting issues to for arbitrary data observations (dots) sampled from a quadratic function [Goodfellow et al., 2016]. Left panel: A linear model suffering from underfitting cannot capture the data curvature. Middle panel: A well-suited quadratic function generalises well to new data observations. Right panel: A high-level polynomial model suffering from overfitting interpolates well the data at the training points but highly oscillates between them.

approaches offer a large flexibility of emulation but require some solid knowledge of the many hyperparameters they handle.

II.3.2 Polynomial chaos expansion

We introduce polynomial chaos expansion as an extension of the linear regression model. The linear model may be enriched with polynomial combinations of the input parameters, which can be rearranged to produce an orthogonal polynomial basis adapted to the input probability distribution.

II.3.2.a Formulation

General formulation. Polynomial chaos expansion uses a linear combination of inputs on a polynomial expansion basis to map the uncertain parameters $\boldsymbol{\mu} \in \mathbb{R}^d$ to the lth reduced coefficient k_l:

$$\begin{aligned} k_l(\boldsymbol{\mu}) &= \lambda_0 H_0 + \sum_{\alpha_1=1}^{d} \lambda_{\alpha_1} H_1(\mu_{\alpha_1}) + \sum_{\alpha_1=1}^{\infty} \sum_{\alpha_2=1}^{\alpha_1} \lambda_{\alpha_1\alpha_2} H_2(\mu_{\alpha_1}, \mu_{\alpha_2}) + \dots \\ \implies k_l(\boldsymbol{\mu}) &= \sum_{\boldsymbol{\alpha}} \lambda_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}(\boldsymbol{\mu}), \end{aligned} \tag{II.26}$$

where $H_i(\mu_{\alpha_1}, \dots, \mu_{\alpha_d})$ denote basis polynomials, $\boldsymbol{\lambda}_{\boldsymbol{\alpha}} = (\lambda_0, \lambda_{\alpha_1}, \lambda_{\alpha_1\alpha_2}, \cdots)$ are the model weights, and $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_d)$ is a multi-index that identifies the components of the multivariate polynomials $\Psi_{\boldsymbol{\alpha}}$. Note that the second line in Eq. (II.26) is more convenient and that the polynomials $\Psi_{\boldsymbol{\alpha}}$ are re-ordered and match the functions H_i. For instance, in the original polynomial chaos formulation (i.e. the homogeneous chaos introduced by Wiener [1938]), the terms H_i were chosen as the Hermite polynomials:

$$H_i(\mu_{\alpha_1}, \dots, \mu_{\alpha_d}) = (-1)^d \exp\left(1/2\, \boldsymbol{\mu}^T \boldsymbol{\mu}\right) \frac{\partial^d}{\partial \mu_{\alpha_1} \dots \mu_{\alpha_n}} \exp\left(-1/2\, \boldsymbol{\mu}^T \boldsymbol{\mu}\right). \tag{II.27}$$

Table II.2: Optimal choice of polynomial basis based on the input probability distribution [Xiu and Karniadakis, 2003].

Random inputs	Wiener-Askey chaos	Support
Gaussian	Hermite-chaos	$\mathbb{R}$
Gamma	Laguerre-chaos	$\mathbb{R}_+$
Beta	Jacobi-chaos	$[a, b]$
Uniform	Legendre-chaos	$[a, b]$

For example, one-dimensional Hermite polynomials read:

$$H_0 = 1, \quad H_1(\mu) = \mu, \quad H_2(\mu, \mu) = \mu^2 - 1, \quad H_3(\mu, \mu, \mu) = \mu^3 - 3\mu, \quad \dots \tag{II.28}$$

Polynomial basis. The polynomial chaos functions form a orthogonal basis in the L_2-space with respect to the joint probability density distribution of the input parameters $\boldsymbol{\mu}$:

$$\langle \Psi_{\boldsymbol{\alpha}}(\boldsymbol{\mu}), \Psi_{\beta}(\boldsymbol{\mu}) \rangle = \int_Z \Psi_{\boldsymbol{\alpha}}(\boldsymbol{\mu}) \Psi_{\beta}(\boldsymbol{\mu}) \, dp(\boldsymbol{\mu}) = \delta_{\boldsymbol{\alpha}\boldsymbol{\beta}}, \ \forall \boldsymbol{\alpha}, \boldsymbol{\beta}, \tag{II.29}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product in the Hilbert space of the random variables, $Z \in \mathbb{R}^d$ is the space in which $\boldsymbol{\mu}$ evolves, $p(\boldsymbol{\mu})$ is the probability density distribution of the inputs, and $\delta_{\boldsymbol{\alpha}\boldsymbol{\beta}}$ is the Kronecker delta-function.

Historically, the Hermite-chaos expansion has been effective in the context of Gaussian and non-Gaussian input distributions [Ghanem, 1999]. The popularity of the Hermite polynomial expansion has been somewhat motivated by Cameron-Martin's theorem, which states the L^2-convergence of polynomial expansions for any functional of L^2 with bounded support [Cameron and Martin, 1947]. However, for non-Gaussian distributions, the convergence rate is not guaranteed and may deteriorate significantly. To overcome this issue, polynomial chaos expansions were combined with the Askey scheme's sets of orthogonal polynomials to extend the procedure to more general random inputs, implying that the choice of the polynomial basis is determined in practice by the distribution $p(\boldsymbol{\mu})$. Table II.1 gives the correspondence between several Askey schemes and the associated random distributions. Each type of polynomial in the Askey scheme forms a complete orthogonal basis that can speed-up the convergence for non-Gaussian distributions [Ogura, 1972; Xiu and Karniadakis, 2002, 2003]. For example, the Legendre polynomials form the optimal basis when the input parameters follows a uniform distribution.

In practice, the multivariate orthogonal basis can be made orthonormal and is often built using the tensor product of one-dimensional polynomial functions:

$$\Psi_{\boldsymbol{\alpha}} = \phi_{\alpha_1} \times \phi_{\alpha_2} \times \cdots \times \phi_{\alpha_d} \tag{II.30}$$

where ϕ_{α_i} represents the one-dimensional polynomial function associated with the input α_i.

Polynomial chaos expansion still stands as a linear model with respect the polynomial basis functions. The polynomial functions alleviate the limited flexibility and expressiveness of linear models, while keeping their easiness of implementation and interpretation.

II.3.2.b Basis truncation

Once the probability distribution is chosen, $\{\boldsymbol{\lambda}_{\boldsymbol{\alpha}}\}$ are the unknowns to calibrate during the training stage to build the polynomial chaos metamodel. This calibration can be done through a Galerkin pseudo-spectral projection (by using the orthonormality property of the polynomial basis – Eq. II.29) or by solving a regression problem [Berveiller et al., 2006]. With this last approach, the polynomial chaos expansion coefficients are obtained by solving a least-square minimization problem. A gradient descent approach can be used for this purpose. One important question lies in the choice of the coefficients to be estimated.

Standard truncation rule. Equation (II.26) describes an infinite sum, which is not easily solvable numerically. In practice, a truncated version of the polynomial chaos expansion is used. Building a polynomial chaos expansion therefore implies to adopt a truncation rule to determine which polynomial features to keep in the representation, i.e.

$$k_l = f_l(\boldsymbol{\mu}) = \sum_{\boldsymbol{\alpha} \in \Theta} \lambda_{\boldsymbol{\alpha}} \, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{\mu}), \tag{II.31}$$

where Θ is a finite set of orthonormal polynomial functions. A first simple choice comes from selecting a maximum degree of truncation in $\Psi_{\boldsymbol{\alpha}}(\boldsymbol{\mu})$, meaning that all polynomials involving the d inputs of maximum degree less or equal to P are retained. Hence, $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_d) \in \{0, 1, \cdots, P\}^d$. Stated differently, the set of selected multi-indices for the multi-variate polynomials Θ is defined as:

$$\Theta \equiv \Theta(d, P) = \{\boldsymbol{\alpha} \in \mathbb{N}^d : |\boldsymbol{\alpha}| \leq P\} \subset \mathbb{N}^d, \tag{II.32}$$

where $|\boldsymbol{\alpha}| = ||\boldsymbol{\alpha}||_1 = \alpha_1 + \cdots + \alpha_d$ is the multi-index total order. In this case, the number of explanatory variables in the polynomial model can be very large and rapidly explode when the number of input parameters increases:

$$\text{Card}\,(\Theta) = \binom{d+P}{P} = \frac{(d+P)!}{d!\ P!}. \tag{II.33}$$

We refer to this basis as the full basis for a given total polynomial order P.

Promoting sparsity. In most cases, not all terms in the polynomial basis are relevant and the most important terms tend to be the main effects and the low-order explanatory variable interactions. This implies that the high-order interaction terms between the inputs can usually be removed from the polynomial basis without any effect on the model predictions. In practice, there are two main techniques to build a sparse polynomial basis: applying a truncation scheme to the full basis, or adding a penalty term.

Truncation schemes excluding high-order interaction terms have been proposed in the literature. For instance, Blatman [2009] suggested the hyperbolic truncation scheme involving the

q-semi-norm:

$$\Theta \equiv \Theta(d, P, q) = \left\{ \boldsymbol{\alpha} \in \mathbb{N}^d : ||\boldsymbol{\alpha}||_q \leq P \right\}, \quad ||\boldsymbol{\alpha}||_q \equiv \left(\sum_{i=1}^{d} (\alpha_i)^q \right)^{1/q}, \tag{II.34}$$

where $q \in [0, 1]$. The adoption of such a semi-norm penalizes high-rank indices and high-order interactions. The lower the value of q, the higher the penalty in the determination of Θ. Note that $q = 1$ corresponds to the simple truncation scheme expressed in Eq. (II.32). An alternative to the hyperbolic truncation scheme is the cleaning strategy, which builds an optimal sparse polynomial chaos expansion containing at most P significant basis functions. Starting from the full basis, the terms that have a low magnitude coefficient are discarded from the basis, i.e. when

$$|\lambda_{\boldsymbol{\alpha}}| \leq v \times \max_{\alpha \in \Theta} |\lambda_{\boldsymbol{\alpha}}|, \tag{II.35}$$

where v is the significance factor. The polynomial chaos is iteratively enriched with significant terms until either P terms are retained or if the given maximum index for $\boldsymbol{\alpha}$ has been reached.

Sparsity can also be promoted by adding a penalty term to the least-square minimisation problem, for instance through a L^1 penalty factor with least absolute shrinkage and selection operator (LASSO) [Tibshirani, 1996] or least-angle regression (LAR) [Blatman and Sudret, 2011].

In the polynomial chaos framework, the model's parameters are the coefficients $\{\boldsymbol{\lambda}_{\boldsymbol{\alpha}}\}_{\boldsymbol{\alpha} \in \Theta}$, while the main hyperparameters are the total polynomial order P and the truncation scheme parameters such as the hyperbolic coefficient q when using hyperbolic truncation scheme or the significance factor v and the maximum number of terms when using the cleaning strategy.

II.3.3 Gaussian process regression model

In complement to polynomial chaos expansion, we aim to construct a Gaussian process regression model. As stated by Rasmussen and Williams [2006], a Gaussian random process is a random stochastic process indexed over the parameter space for which any finite collection of functions has a joint Gaussian distribution. It is then fully described by its mean and correlation structure (or kernel) that are conditioned by the training dataset $(\mathcal{U}, \mathcal{K}_l)$. This implies that a Gaussian process regression model can make predictions of the quantities of interest incorporating prior knowledge and can provide uncertainty estimates over the predictions. In the present context of reduced-order modelling, the Gaussian process regression model aims at predicting the reduced coefficients over the variation interval of the uncertain parameters $\boldsymbol{\mu}$.

II.3.3.a Formulation

Reduced-coefficient prediction. The Gaussian process regression framework assumes that the mapping f_l is a Gaussian stochastic process such that:

$$k_l = f_l(\boldsymbol{\mu}) \sim \mathcal{GP}\left(m_l(\boldsymbol{\mu}), r_l(\boldsymbol{\mu}, \boldsymbol{\mu}^*)\right) \quad \forall\, (\boldsymbol{\mu}, \boldsymbol{\mu}^*) \in \mathcal{P} \times \mathcal{P}, \tag{II.36}$$

where $m_l(\boldsymbol{\mu}) = \mathbb{E}[f_l(\boldsymbol{\mu})]$ is the mean function of the Gaussian process, and $r_l(\boldsymbol{\mu}, \boldsymbol{\mu}^*) = \mathbb{E}[(f_l(\boldsymbol{\mu}) - m_l(\boldsymbol{\mu}))\,(f_l(\boldsymbol{\mu}^*) - m_l(\boldsymbol{\mu}^*))]$ is its associated covariance function.

Gaussian process regression starts with a prior distribution over the mean and covariance function. For instance, the prior mean can be assumed constant and equal to the empirical average of k_l estimated on the training samples $\mathcal{K}_l$. The prior covariance is specified by choosing a kernel (Sect. II.3.3.b). Note that in the framework of POD, some properties on the reduced basis coefficients are satisfied (Sect. II.2.1.d). When whitening is applied, the prior mean of the Gaussian process regression model can be set to 0, and its prior variance can be set to 1. The learning stage consists in updating the mean and covariance by integrating the information from the reduced-coefficient dataset.

The joint distribution between the training dataset $(\mathcal{U}, \mathcal{K}_l)$ and some new test evaluations $(\mathcal{U}^*, \mathcal{K}_l^*)$ is expressed with respect to the kernel as:

$$\begin{bmatrix} \mathcal{K}_l \\ \mathcal{K}_l^* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} r_l(\mathcal{U},\mathcal{U}) + s_l^2\, I & r_l(\mathcal{U},\mathcal{U}^*) \\ r_l(\mathcal{U}^*,\mathcal{U}) & r_l(\mathcal{U}^*,\mathcal{U}^*) \end{bmatrix}\right), \tag{II.37}$$

where s_l is the noise variance (Eq. II.21), and I stands for the identity matrix. We can derive the inference formula for the test reduced coefficients from the following conditional distribution:

$$\mathcal{K}_l^* \mid \mathcal{U}, \mathcal{K}_l, \mathcal{U}^* \sim \mathcal{N}\left(m_l^*, \mathrm{cov}(\mathcal{K}_l^*)\right), \tag{II.38}$$

where

$$\begin{cases} m_l^* & = r_l(\mathcal{U}^*,\mathcal{U})\ \left[r_l(\mathcal{U},\mathcal{U}) + s_l^2\, I\right]^{-1}\ \mathcal{K}_l \\ \mathrm{cov}(\mathcal{K}_l^*) & = r_l(\mathcal{U}^*,\mathcal{U}^*) - r_l(\mathcal{U}^*,\mathcal{U})\ \left[r_l(\mathcal{U},\mathcal{U}) + s_l^2\, I\right]^{-1}\ r_l(\mathcal{U},\mathcal{U}^*). \end{cases} \tag{II.39}$$

All terms in Eq. (II.39) are known. The covariance formulation depends on prior variance $r_l(\mathcal{U}^*,\mathcal{U}^*)$ over the test dataset refined by information from the training dataset. Note that in practice, the matrix $[r_l(\mathcal{U},\mathcal{U}) + s_l^2\, I]$ in Eq. (II.39) can be inverted using a computationally-efficient Cholesky decomposition [Rasmussen and Williams, 2006]. The posterior distribution for the lth reduced coefficient k_l can then be directly estimated using Eq. (II.38). The Gaussian process regression estimator is set as the mean posterior, which is a linear combination of kernel distances computed between the test point and all the training data.

II.3.3.b Choice of the kernel

The choice of the kernel $r_l(\boldsymbol{\mu}, \boldsymbol{\mu}^*)$ is at the core of Gaussian process regression as it entails specific assumptions on data covariance in the input space $\mathcal{P}$, i.e. it describes how similar two data points $(\boldsymbol{\mu}, \boldsymbol{\mu}^*)$ behave. The metamodel is mainly built by estimating the kernel hyperparameters (e.g. variance, characteristic length-scales) that provide a good fit to the training dataset. Several kernel functions are available in the literature among whom the radial basis function kernel and the more general Matérn kernels.

Radial basis function kernel. The radial basis function (RBF), also known as the squared exponential kernel, is the most widely used kernel in the framework of Gaussian process regression.

The RBF is a stationary kernel as it is expressed as a function of the ℓ_2-norm Euclidean distance $d(\boldsymbol{\mu}, \boldsymbol{\mu}^*) = \|\boldsymbol{\mu} - \boldsymbol{\mu}^*\|_2$ for $\boldsymbol{\mu}$ and $\boldsymbol{\mu}^*$, two sets of parameters of the input space $\mathcal{P}$:

$$r_{\text{rbf}}(\boldsymbol{\mu}, \boldsymbol{\mu}^*) = \varrho \exp\left(-\frac{d(\boldsymbol{\mu}, \boldsymbol{\mu}^*)^2}{2\,\lambda^2}\right). \tag{II.40}$$

where ϱ is the signal variance parameter, and λ is the length-scale (or stability) hyperparameter. The RBF kernel is infinitely differentiable and has therefore interesting smoothness properties.

Matérn kernel. The Matérn class of covariance functions can be seen as a generalisation of the RBF kernel. In additional to the length-scale λ, it also includes a smoothness hyperparameter. The Matérn kernels are also stationary and can be generally expressed as:

$$r_{\text{Matérn}}(\boldsymbol{\mu}, \boldsymbol{\mu}^*) = \varrho\,\frac{2^{1-\nu}}{\gamma(\nu)}\left(\frac{\sqrt{2\nu}}{\lambda}\,d(\boldsymbol{\mu}, \boldsymbol{\mu}^*)\right)^{\nu} \mathcal{B}_\nu\left(\frac{\sqrt{2\nu}}{\lambda}\,d(\boldsymbol{\mu}, \boldsymbol{\mu}^*)\right), \tag{II.41}$$

where $\gamma(\,\cdot\,)$ is the Gamma function and $\mathcal{B}_\nu(\,\cdot\,)$ is a modified Bessel function, and where ϱ is the signal variance parameter and $\lambda > 0$ is the length-scale as for the RBF kernel in Eq. (II.40), and where $\nu > 0$ is the smoothness hyperparameter. The smaller ν, the less smooth the metamodel is. When $\nu \to \infty$, it becomes equivalent to the RBF kernel.

The stochastic Gaussian process resulting from a Matérn kernel is $\lceil \nu \rceil - 1$ times differentiable in the mean-square sense. The smoothness parameter ν will take the form $\nu = p + 1/2,\ p \in \mathbb{N}$, since it is a common choice in machine learning framework (e.g. Elbeltagi et al., 2021; Mukesh Kumar and Kavitha, 2021).

Anisotropic kernel. The length-scale parameter λ in the RBF or Matérn kernel represents the level of variability in the reduced coefficients as a function of distance in the input space $\mathcal{P}$. In practice, the length-scale can either be a scalar or can vary for each dimension of the input vector $\boldsymbol{\mu}$. The kernel is then qualified as anisotropic. When the input parameters are of very different nature, it is recommended to adopt an anisotropic kernel. Anisotropy may be embedded using a distinct correlation length-scale per dimension:

$$d(\boldsymbol{\mu}^{(m)}, \boldsymbol{\mu}^{(n)}) = \sqrt{(\boldsymbol{\mu}^{(m)} - \boldsymbol{\mu}^{(n)})^T\,\boldsymbol{\Lambda}\,(\boldsymbol{\mu}^{(m)} - \boldsymbol{\mu}^{(n)})}, \tag{II.42}$$

where $\boldsymbol{\mu}^{(m)}$ and $\boldsymbol{\mu}^{(n)}$ are realisations of the input vector $\boldsymbol{\mu}$, and where $\boldsymbol{\Lambda} \in \mathbb{R}^{d\times d}$ corresponds to the length-scale matrix (with d the size of the input vector $\boldsymbol{\mu}$). If the different length-scales are assumed independent, the matrix $\boldsymbol{\Lambda}$ will be of the form:

$$\boldsymbol{\Lambda} = \text{diag}(1/(\lambda_{\mu_1}^2), 1/(\lambda_{\mu_2}^2), \cdots, 1/(\lambda_{\mu_d}^2)), \tag{II.43}$$

where μ_i represents the ith input parameter in $\boldsymbol{\mu}$. This form of the Gaussian process length-scale matrix is referred to as automatic relevance determination (ARD) in the literature.

II.3.3.c Hyperparameter optimisation

Hyperparameter settings have a substantial impact on the Gaussian process regression model prediction performance. An optimisation process is usually used to determine an optimal value for the hyperparameters rather than simply specifying them. In the present context of reduced-order modelling, the reduced coefficients can be very different due to the complex nature of the latent space. It is therefore important to optimise the hyperparameters for each of the L Gaussian process regression models to adapt to the characteristic length-scale of each mode (Sect. II.3.3.a). For the present noisy Gaussian process regression framework with anisotropic kernel, the set of hyperparameters $\boldsymbol{\theta}_l$ includes the correlation length-scales, the noise variance as well as the Gaussian process variance, meaning that $\boldsymbol{\theta}_l = \{s_l^2, \varrho, \lambda_{\mu_1}, \cdots, \lambda_{\mu_d}\} \in \mathbb{R}^{d+2}$.

Empirical Bayesian maximisation is used to determine the optimal set of the hyperparameters maximising their posterior:

$$\begin{aligned} \boldsymbol{\theta}_{l,\text{opt}} &= \arg\max_{\boldsymbol{\theta}_l} \log p(\boldsymbol{\theta}_l \,|\, \mathcal{U}, \mathcal{K}_l) \\ \Rightarrow \boldsymbol{\theta}_{l,\text{opt}} &= \arg\max_{\boldsymbol{\theta}_l} \log p(\mathcal{K}_l \,|\, \mathcal{U}, \boldsymbol{\theta}_l) + \log p(\boldsymbol{\theta}_l), \end{aligned} \tag{II.44}$$

The first term called the marginal log-likelihood is assumed to be Gaussian; and the second term involves the prior distribution over the hyperparameters. In the literature, gradient descent is widely used to find the local optimum of Eq. (II.44) [Rasmussen and Williams, 2006].

Maximum log-likelihood estimation. Without making any further assumption about the noise or the length-scales, one may proceed with a naive optimisation of the log-likelihood, assuming uniform prior distributions over the hyperparameters (i.e. the term $p(\boldsymbol{\theta}_l)$ in Eq. II.44). Unfortunately, gradient descent algorithms perform poorly in this case due to multiple local optima. To overcome this issue, one way is to perform multiple gradient descent iterations starting from different hyperparameter initial conditions. The final solution is then chosen as the one achieving the highest maximal log-likelihood (MLL) score. These multiple gradient descent iterations increase the computational cost of the Gaussian process regression, especially since the optimisation process is repeated L times for each reduced coefficient.

Maximum a posteriori estimation. To ensure convergence to a solution consistent with reduced-basis properties, the optimisation procedure can be informed by providing prior distributions (the term $p(\boldsymbol{\theta}_l)$ in Eq. II.44) and an appropriate starting point for the hyperparameters. These information can be derived from the reduced basis coefficients. This approach is referred to as the maximum a posteriori (MAP) estimation in the following.

The key settings of Gaussian process regression are related to the kernel: the kernel function (e.g. RBF, Matérn), and the kernel hyperparameters – the correlation length-scales $(\lambda_{\mu_1}, \cdots, \lambda_{\mu_d})$, the noise variance s_l^2 and the Gaussian process variance ϱ. These hyperparameters are estimated through an optimisation process (MLL, MAP) that can be costly but that is necessary to find the Gaussian processes that fit the most the observations. Note that the Matérn class of functions includes an additional smoothness parameter ν. In practice, strong smoothness is irrelevant when dealing with experimental data since it might be hard to distinguish high values of smoothness ($\nu \geq 7/2$) from noisy data [Rasmussen and Williams, 2006]. Since the training dataset is assumed to be noisy in this work, it is important to evaluate the sensitivity of Gaussian process regression to the choice of the kernel and in particular of the smoothness parameter ν.

II.3.4 Gradient tree boosting

Gradient boosting is an ensemble learning algorithm that sequentially aggregates metamodels. It can be applied to different metamodelling classes but decision trees are typically good candidates for boosting when their size is reasonable. They are rather fast to train and yield interpretable models, but they may suffer from significant instability (slight variations in data can result in significantly different trees) and typically result in high variance. Boosting mitigates these limitations.

II.3.4.a Principle of boosting

In practice, a boosted model f_l can be expressed as a sum of metamodels belonging to the same class of functions T:

$$k_l = f_l(\boldsymbol{\mu}) = \underbrace{\sum_{m=0}^{M} T_m(\boldsymbol{\mu}; \boldsymbol{\lambda}_m)}_{f_{l,M}(\boldsymbol{\mu})}, \tag{II.45}$$

where T denotes the class of metamodels (e.g. decision trees), M the number of individual metamodels to be trained, and $\boldsymbol{\lambda}_m$ the hyperparameters related to the mth metamodel T_m (e.g. the number of leaves for decision trees).

The idea behind boosting is to iteratively add new metamodels to the existing sum to provide increasingly robust predictions. Boosting stands as a functional gradient descent. In a general framework, boosting aims at minimising the risk function (Eq. II.22). For a given differentiable loss function $\mathcal{L}$, the minimisation problem can be approximated using gradient descent on the empirical loss (Eq. II.25):

$$\begin{cases} f_{l,0} = T_0 \in \mathbb{R} \\ f_{l,M} = f_{l,M-1} - \rho \nabla \widehat{R}(f_{l,M-1}, \mathcal{D}_n) \iff T_M = -\rho \nabla \widehat{R}(f_{l,M-1}, \mathcal{D}_n), \ \forall M > 0, \end{cases} \tag{II.46}$$

where $\rho \in \mathbb{R}$ is the gradient descent step (or shrinkage), and the initial guess $f_{l,0}$ is usually set

as the mean over the training snapshots.

At each iteration, a new metamodel $f_{l,M}$ is introduced as an estimator of $\nabla\mathcal{L}(f_m)$, leading to sequentially dependent learners. Note that when the risk is derived from the squared-error loss, the gradient term $\nabla\mathcal{L}(f_{l,M})$ matches the pseudo-residuals. In this specific case, the new metamodel is fitted to the current least-square pseudo-residuals:

$$f_{l,M} : \boldsymbol{\mu} \mapsto k_l - f_{l,M-1}(\boldsymbol{\mu}), \tag{II.47}$$

with k_l the reference lth reduced coefficient. In a more general framework, any almost everywhere differentiable loss can be implemented for boosting. More robust criterion such as the absolute error or the Huber loss [Friedman, 2001; Huber, 2011] may provide stronger resistance to outliers, while being almost as efficient as squared-errors.

Techniques for improving gradient descent also apply to the boosting procedure.

– It is possible to tune the learning step through a shrinkage term $\rho \in]0, 1]$ that controls the learning rate and prevents overfitting [Friedman, 2001]. However, smaller values of ρ (more shrinkage) increase the computational cost: it slows down the convergence of the gradient boosting procedure as $f_{l,M}$ requires a larger amount M of metamodels. Usually, ρ is set small (less than 0.1) and an optimal value for M is obtained by an early stopping criterion.

– Stochastic gradient boosting, in analogy with stochastic gradient descent (SGD), can produce more accurate boosting models through variance reduction while reducing computational cost by introducing random mini-batch averaging. It uses subsamples of the training batch to estimate empirical risk gradient at each iteration [Friedman, 1999]. Usually, the fraction of data kept in the mini-batch is less than 50% of the training batch size.

– An early stopping criterion may be used to determine how many trees should be retained in the boosting model to prevent overfitting and to maximise generalisation error. Hyperparameters control the early stopping rule such as the proportion of data set aside for validation, the number of iterations without performance improvement or the tolerance threshold (the algorithm is stopped when the performance is not improved by at least the tolerance value).

II.3.4.b Regression trees

We now introduce decision trees – or more precisely classification and regression trees (CART) – as a piecewise constant function [Breiman et al., 2017]. Decision trees partition the input space of $\boldsymbol{\mu}$ into subregions $\{\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_A\}$ associated with constant local responses $\{\gamma_1, \gamma_2, \ldots, \gamma_A\}$. Mathematically, a decision tree model can be written as:

$$T(\boldsymbol{\mu}; \boldsymbol{\lambda}) = \sum_{a=1}^{A} \gamma_a \mathbb{1}(\boldsymbol{\mu} \in \mathcal{R}_a), \tag{II.48}$$

with $\boldsymbol{\lambda} = \{\mathcal{R}_a, \gamma_a\}_{a=1,\ldots,A}$.

Approximate suboptimal solutions of $\boldsymbol{\lambda}$ can be computed from an iterative recursive algorithm. The algorithm can be summarized as follows: at each iteration, the algorithm chooses an optimal input parameter $(\mu_i, i = 1, \dots, d)$ to split the domain along with a splitting value s. For the first iteration, the full domain is split into two subsets $\mathcal{R}_1(i,s) = \{\boldsymbol{\mu} \mid \mu_i \leq s\}$ and $\mathcal{R}_2(i,s) = \{\boldsymbol{\mu} \mid \mu_i > s\}$. Solutions for μ_{i^*} and s^* are obtained from the following minimisation problem:

$$\min_{i,s} \left[\min_{\gamma_1} \sum_{\boldsymbol{\mu} \in \mathcal{R}_1(i,s)} \mathcal{L}(\gamma_1, k_l^{(n)}) + \min_{\gamma_2} \sum_{\boldsymbol{\mu} \in \mathcal{R}_2(i,s)} \mathcal{L}(\gamma_2, k_l^{(n)}) \right], \tag{II.49}$$

where $\mathcal{L}$ is the loss function, γ_1 and γ_2 are constant values associated with the two subspaces $\mathcal{R}_1$ and $\mathcal{R}_2$, and $\{k_l^{(n)}\}_n$ the reduced basis coefficients of the training snapshots. Finding optimal values for s and i is done by scanning through all of the inputs. It remains computationally feasible but scales with the dataset size. Determining the domain partitions γ_1 and γ_2 is usually an easy task. For instance, the optimal solution for the mean-square error is the mean value obtained over the corresponding subspace, i.e. $\gamma_1 = \text{mean}(k_l^{(n)} | \forall \mu_i^{(n)} \in \mathcal{R}_1(i,s))$. Finally, this process is repeated recursively on all of the resulting subregions. At the end of the process, the final subregions correspond to the terminal nodes of the tree and related γ values coincide to the metamodel responses.

One issue in decision trees relates to the size of the trees, or more generally to their statistical complexity. The aforementioned procedure has no stopping criteria, meaning that the procedure maximises the tree complexity as it splits the input space into as many snapshots as there are in the training database. The final subregions (the leaf nodes of the tree) will only contain a unique sample. This will result in both overfitting and increased computational cost. We now discuss how to handle the complexity of trees in the context of gradient boosting.

II.3.4.c Managing tree complexity for boosting

Tree complexity is characterized by the depth D and the number of terminal nodes J. While a large tree overfits data, a small tree might not capture the key patterns present in the data. Boosting partially removes the issue of tree complexity: if a tree does not capture the full complexity of the response surface, the next trees will balance this lack of accuracy and improve the overall response. In practice, tree complexity is usually kept small in the context of boosting for computational reasons.

An initial guess on maximum tree depth can be given from the interaction level complexity of the input variables. For instance, if the input dimension stands as $\boldsymbol{\mu} = \{\mu_1, \dots, \mu_d\} \in \mathbb{R}^d$, no interaction of order larger that d is possible. A tree with a maximum depth $d+1$ handles at most dth-order effects. For instance, three-depth tree will allow second-order interactions at maximum. There is therefore no reason to have trees with a depth much larger than the input dimension $D \leq d+1$. It should be noted, however, that for statistical reasons, which are highly dependent on the cases studied, enabling slightly increased tree depth may be likely to slightly increase the performance of the boosting method. As a result, the rule is merely a hint of the order of magnitude to select and usual tree depth is chosen such that $D \approx d+1$.

Trees naturally tend towards overfitting. Tree complexity must therefore be constrained by setting hyperparameters. One may configure (*i*) a minimum number of samples per leaf node, (*ii*) a maximum depth of the tree, (*iii*) the features to look at for choosing the splitting rule, (*iv*) a minimum threshold for the decrease of the cost function (Eq. II.49) and/or (*v*) a penalisation term. For instance, cost-complexity pruning introduces an additive penalisation term to the tree cost function. For the mean-squared error, the new cost can be expressed as:

$$
\begin{aligned}
C_{\alpha_{\mathrm{CCP}}}(T) &= \sum_{m=1}^{J} N_m \, Q_m(T) + \alpha_{\mathrm{CCP}} \, J, \\
&\text{where} \begin{cases} N_m = \mathrm{Card}\left(\{\boldsymbol{\mu} \in \mathcal{R}_m\}\right), \\ \gamma_m = \frac{1}{N_m} \sum\limits_{\boldsymbol{\mu}^{(n)} \in \mathcal{R}_m} k_l^{(n)}, \\ Q_m(T) = \frac{1}{N_m} \sum\limits_{\boldsymbol{\mu}^{(n)} \in \mathcal{R}_m} (k_l^{(n)} - \gamma_m)^2, \end{cases}
\end{aligned}
\tag{II.50}
$$

where J denotes the number of terminal leaves, and $\alpha_{\mathrm{CCP}} \geq 0$ is the penalisation hyperparameter (larger values of α_{CCP} result in smaller trees). To find the optimal tree, we successively collapse the internal node that produces the smaller increase in $\sum_m N_m Q_m(T)$. The procedure is reproduced sequentially until reaching the single-node (root) tree. This algorithm is referred to as weakest link pruning. Optimal estimation of α_{CCP} may be achieved by cross-validation.

The main strength of gradient boosting is its ability to approximate highly non-linear response surfaces, even with severe discontinuities, using boosted decision trees. Gradient boosting is typical of machine learning algorithms that provide a high degree of flexibility since it is based mostly on a statistical representation of the data but quickly loses physical meaning. The flexibility of boosting is ensured by a large number of parameters that concern both the configuration of the boosting procedure (similar to a gradient descent) and the decision trees. Tuning all these parameters can be tedious and requires a good comprehensive knowledge of statistics fundamentals.

II.4 Learning algorithm synthesis

The reduced-order models we consider in this work include two components: a dimension reduction component (Sect. II.2) and a regression component (Sect. II.3). The construction of the reduced-order models can be done using a training dataset. The resulting reduced-order models are then evaluated using an independent dataset referred to as the test dataset.

Statistical model hyperparameters add degrees of freedom to the reduced-order model construction. The test dataset should not be used to both choose the model hyperparameters and compare the different model families. Otherwise, one may overfit each model candidate by fine-tuning their hyperparameter values based on the test performance. This issue is particularly present if the number of hyperparameters is large as in gradient tree boosting and in neural network approaches (having a large number of hyperparameters gives more flexibility in the model construction). Moreover, using a dataset that is independent of the training dataset to

determine the hyperparameters values is essential to provide unbiased estimations of optimal hyperparameters and ensure generalisation capacity of the reduced-order models. Hence, the full LES dataset is split into three subsets: *i)* a training dataset to learn the dimension reduction model and the mapping between the uncertain inputs $\boldsymbol{\mu}$ and the reduced coefficients $\boldsymbol{k}$ using regression models; *ii)* a calibration dataset to investigate multiple hyperparameter configurations within each family; and *iii)* a test dataset to evaluate the capacity to predict LES quantities of interest for new samples of the uncertain input parameters. A fairly standard trade-off for splitting the database is roughly 60%, 20%, 20% for training, validation and test, respectively.

The algorithm we use to train and validate the reduced-order models can be summarised as the following training/prediction two-step process.

Training stage

1. Build the latent space representation (Eq. II.3), including training the encoder and decoder functions (using POD or autoencoders).

2. Transform data inputs $\boldsymbol{\mu}$ and outputs $\mathbf{K}_{\text{les}}$: the inputs $\boldsymbol{\mu}$ follow a uniform statistical distribution on $[0,1]^d$ (where d is the number of uncertain inputs); and the outputs $\mathbf{K}_{\text{les}}$ are encoded into the latent variables and whitened.

3. Train the L regression models $\{f_l\}_{l=1,\dots,L}$ (e.g. using polynomial chaos expansion, Gaussian process regression and gradient tree boosting) independently: i.e. estimate the weight parameter values $\{\boldsymbol{\theta}_{l,\text{opt}}\}_{l=1,\dots,L}$ (e.g. using ordinary least squares or gradient descent depending on the model) from training data.

4. Choose the best values of the hyperparameters to maximise the regression model's performance from calibration data.

Prediction stage

1. Predict the latent variables $\{k_l(\boldsymbol{\mu}^*)\}_{l=1,\dots,L}$ for LES test samples of input parameters $\boldsymbol{\mu}^*$ (Eqs. II.38-II.39),

2. Perform decoding to recover the fields $\mathbf{K}^*_{\text{rb}}$ from the predicted latent variables for the test samples (Eq. II.15),

3. Compare the emulated fields with the reference LES test samples $\mathbf{K}^*_{\text{les}}$ (Eqs. IV.3-IV.4).

Figure II.8 shows the final architecture of the reduced-order model, and illustrates the three successive key steps: data preprocessing, emulation on the reduced-dimensional latent space, and decoding to the original high-dimensional space. The performance of the reduced-order model may be validated by comparing the emulated fields to the LES reference solutions over the test database. The performance criterion can be chosen according to the objective of the reduced-order model. For example, in a regression context, Chapter IV relies on the Q^2 metric to evaluate the model accuracy.

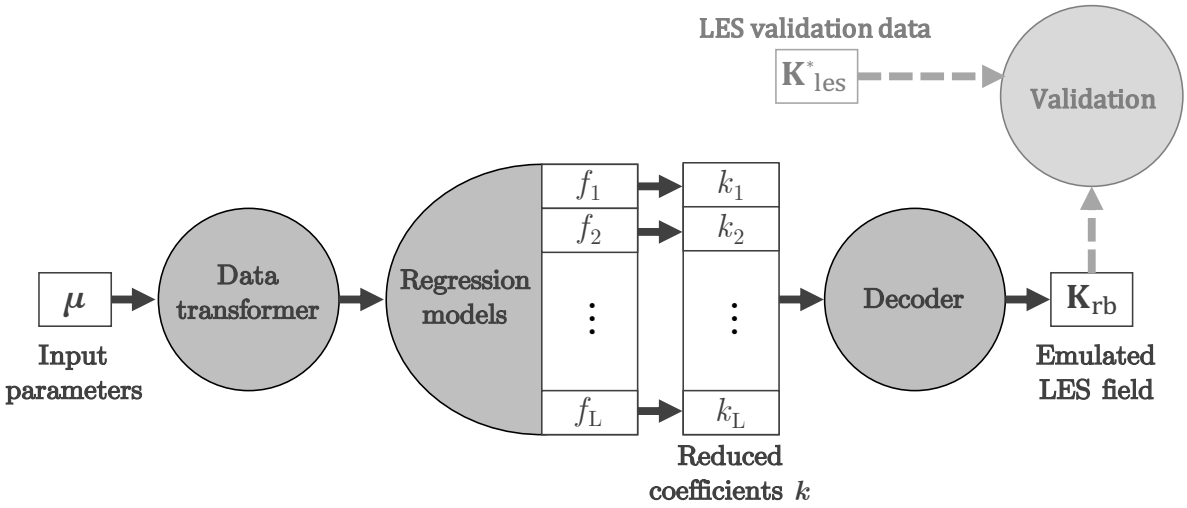

Figure II.8: Schematic of the reduced-order modelling approach consisting in training L independent metamodels to emulate the L reduced coefficients $[k_1, \cdots, k_L]$ with respect to the input parameters $\boldsymbol{\mu}$, and then to reconstruct the LES field of interest by an inverse decoding transformation (the emulated LES field can be compared to the LES test dataset for validation).

Chapter III

Case study of dispersion around a wall-mounted obstacle

The scope of this thesis is to thoroughly investigate different reduced-order modelling approaches informed by LES simulations, which provide a complex description the flow and tracer concentration patterns in urban environment. To achieve this objective, we need to choose a suitable case study that is representative of the microscale urban flow structures but that also provides access to a large database for detailed analysis. We therefore consider a two-dimensional case of dispersion in an atmospheric boundary-layer interacting with an isolated surface-mounted obstacle, for which inlet atmospheric conditions and source parameters are uncertain and induce uncertainties in the flow velocity and tracer concentration field statistics. The obstacle has a simplified square section, which is simple enough to limit computational cost but at the same time the induced flow structures are well studied in the literature by experimental wind tunnel studies [Li and Meroney, 1983a,b] and CFD simulations [Li and Stathopoulos, 1997; Blocken et al., 2008b; Tominaga and Stathopoulos, 2009, 2010; Gousseau et al., 2012; Bazdidi-Tehrani et al., 2013]. This chapter details the case study and the associated numerical setup. The complexity of the flow and tracer response is briefly illustrated, along with the diversity of the response to the uncertain parameters. The generation of the LES ensemble in this parametric setting is then described.

Contents

III.1 Case study description

This section presents the numerical solver and the case study that is simulated using LES in this work.

III.1.1 Numerical solver

The computations are performed with the AVBP[1] DNS and LES code developed by CERFACS [Schönfeld and Rudgyard, 1999; Gicquel et al., 2011]. AVBP solves the compressible Navier-Stokes equations on unstructured grids. Additional advection-diffusion equations are solved for passive scalar dispersion. It is widely used to predict non-reactive and reacting unsteady flows in simple or complex geometries, and is applicable to pollutant formation and atmospheric dispersion [Poubeau et al., 2016; Paoli et al., 2020].

In this work, the numerical discretisation is based on an explicit, centred scheme from the continuous Taylor-Galerkin family called TTG4A, which is third-order in space and fourth-order in time on unstructured grids [Colin and Rudgyard, 2000]. An artificial compressibility approach (also known as Pressure Gradient Scaling/PGS [Ramshaw et al., 1985]) is used as compressibility and acoustics effects are not relevant for atmospheric flows that are very low-Mach flows: with this approach, the solution of the incompressible equations (Eq. I.16) are

[1] AVBP documentation, see http://www.cerfacs.fr/avbp7x/

recovered with this transformation of the compressible formulation of AVBP. It enables to artificially reduces the speed of sound and thereby relaxes the constraint on the time-step due to the Courant-Friedrichs-Lewy (CFL) condition (for explicit time discretisation scheme). This makes the artificial compressibility approach competitive with incompressible solvers.

III.1.2 Case study description

The case study is selected to be representative of a canonical dispersion problem, namely the interaction of an atmospheric boundary flow with a surface-mounted obstacle. In a context of construction and validation of reduced-order models, it is necessary *i*) to dispose of as many LES snapshots as may be needed for a thorough evaluation of the source errors, and *ii*) to be able to quickly implement and evaluate different approaches. This is in contradiction with the use of LES, which involves an important computational and storage burden, with database sizes that are expected to be limited in practical applications (about 10 to 100 snapshots). For the sake of this study, the canonical problem is restricted to a two-dimensional computational domain. It induces artefacts in the numerical models, as turbulence, which is partly resolved in LES, is intrinsically three-dimensional: the two-dimensional structures are perfectly coherent in the missing (spanwise) direction; fundamental mechanisms like vortex stretching are absent in two-dimensional flow simulations [Pope, 2000], while other phenomena like back-scatter become significant [Rivera et al., 2003]. Still, the proposed two-dimensional setup described below involves the main physical processes that are relevant for atmospheric dispersion and the associated challenges in terms of reduced-order modelling: the problem exhibits a wide variety of scales (with different time-scales for the incoming flow, vortex shedding and turbulence around the obstacles), with a strong nonlinear response to input parameters, in particular to tracer source location uncertainty.

As illustrated in Fig. III.1, we consider a case in which a single obstacle interacts with a fully-developed neutral turbulent boundary-layer flow.

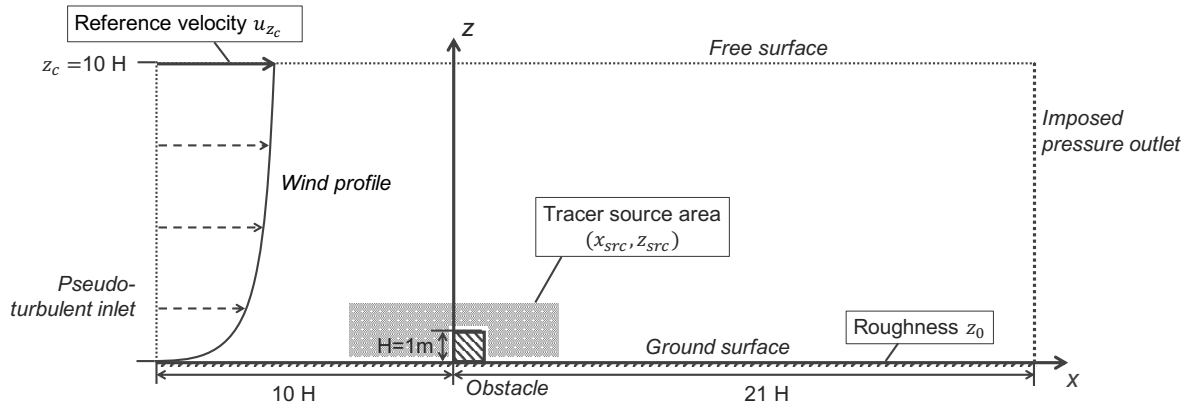

Figure III.1: Sketch of the test case modelling a turbulent boundary-layer flow (coming from the left boundary) interacting with a surface-mounted square obstacle (crosshatched area). Text boxes indicate the uncertain parameters. The grey area indicates the area for the tracer emission source.

The height of the obstacle is $H = 1$ m. The two-dimensional computational domain is 31-m long (x-axis, streamwise direction) by 10-m high (z-axis, vertical direction). The domain height

is ten times the obstacle height following guidelines for urban flow simulations [Franke et al., 2011]. It is discretised with a uniform triangular mesh comprising 240,000 elements with an edge size $\Delta xz = H/10$.

The left boundary of the domain (at $x = -10$ m) corresponds to a turbulent inlet boundary where unsteady wind conditions are imposed to mimic atmospheric boundary-layer turbulence, as detailed in Sect. III.1.3. The right boundary (at $x = 21$ m) and the upper boundary (at $z = 10$ m) correspond to an outlet with imposed pressure condition to mimic the atmosphere. The ground (at $z = 0$ m) is modelled as a rough surface with a law of the wall based on the roughness length z_0 [m]. The obstacle surfaces (the obstacle is centred in $(x, z) = [0.5, 0.5]$ m×m) are modelled with the standard law of the wall.

The passive gas tracer emission source is constant in time and modelled by a Gaussian shape in space with a spread parameter $\sigma_{src} = 0.1$ m around the center of release $(x_{\mathrm{src}}, z_{\mathrm{src}})$:

$$Q_s(x, z; x_{\mathrm{src}}, z_{\mathrm{src}}) = \frac{1}{2\pi \times \sigma_{src}^2} \exp\left\{-\frac{1}{2 \times \sigma_{src}^2}\left((x - x_{\mathrm{src}})^2 + (z - z_{\mathrm{src}})^2\right)\right\}. \tag{III.1}$$

The emission source can be either located upstream, above or downstream of the obstacle as seen from the tracer source area in Fig. III.1.

III.1.3 Inflow boundary condition modelling

Mean inlet wind profile. The inlet wind condition imposes a mean vertical profile $\overline{u}_{inlet}$, based on the Monin-Obukhov similarity theory in neutral conditions:

$$\begin{cases} \overline{u}_{inlet}(z) = \dfrac{u_\tau}{\kappa} \log\left(\dfrac{z + z_0}{z_0}\right), \\ \overline{u}_{inlet}(z = z_c) = u_{z_c}, \end{cases} \tag{III.2}$$

where u_τ [m s^{-1}] is the friction velocity, $\kappa = 0.41$ is the dimensionless von Kármán constant, z_0 [m] is the aerodynamic roughness length, and z [m] is the vertical axis in the domain. In this study, the velocity u_{z_c} at the reference height $z_c = 10$ m and the characteristic surface roughness length z_0 are used as input parameters in order to mimic operational conditions, where velocity measurements are typically obtained at some arbitrary reference height [Brutsaert, 2013; Sousa and Gorlé, 2019]. From u_{z_c} and z_0, Eq. (III.2) can be inverted to obtain the corresponding friction velocity u_τ:

$$u_\tau = \frac{\kappa}{\log\left(1 + \frac{z_c}{z_0}\right)} u_{z_c}. \tag{III.3}$$

In accordance with the inlet profile, the same surface roughness length z_0 is considered assuming a fully developed boundary-layer flow in equilibrium with the rough terrain.

Inlet wind fluctuations. In addition to the mean inlet wind profile from Monin-Obukhov similarity theory, wind fluctuations are superimposed on the mean profile to obtain a turbulent inlet boundary condition that mimics boundary-layer turbulence. The synthetic fluctuations are generated with the Kraichnan method [Kraichnan, 1970], and follow the Passot-Pouquet turbulence

spectrum [Passot and Pouquet, 1987]. The target turbulent kinetic energy k_{tke} at the inlet is estimated as $k_{tke} = u_\tau^2/\sqrt{C_\mu}$ [Richards and Hoxey, 1993].

III.1.4 Flow and tracer dispersion main features

We present here what is identified as the reference LES run of the parametric study in order to provide insights into the unsteady physical processes that are captured by LES. This reference LES run corresponds to the nominal snapshot, i.e. it corresponds to the averaged atmospheric conditions over the ensemble of LES snapshots that we generate in the uncertainty quantification study (that is further discussed in Sect. III.2.2.a). For this reference snapshot, the inlet wind profile is defined by the parameters $u_{z_c} = 5.78\ \mathrm{m\,s^{-1}}$ and $z_0 = 2.79 \times 10^{-2}$ m; the emission source is centred at $(x_{\mathrm{src}}, z_{\mathrm{src}}) = (-1.01\,\mathrm{m}, 0.83\,\mathrm{m})$.

III.1.4.a Output variable normalisation

To analyse the LES simulations, we normalise the output velocity and concentration variables u and K by a reference velocity $u_\tau^{(ref)} = 3.7 \times 10^{-1}\ \mathrm{m\,s^{-1}}$ and a reference length scale (the obstacle height $H = 1\,\mathrm{m}$) as follows:

$$\widetilde{u} = \frac{u}{u_\tau^{(ref)}}, \qquad \widetilde{K} = K\left(\frac{u_\tau^{(ref)}\,\mathrm{H}^2}{Q_s}\right), \tag{III.4}$$

where the normalised velocity and tracer concentration variables are denoted by $\widetilde{u}$ and $\widetilde{K}$, respectively, and where Q_s is the flow rate. The value of $u_\tau^{(ref)}$ represents the mean friction velocity for the flow conditions explored in the parametric study (see Sect. III.2.2.a for further details). In the following, we drop the tilde notation for the sake of simplification.

III.1.4.b Instantaneous flow and tracer dispersion features

Figure III.2 shows several instantaneous normalised tracer concentration fields along with instantaneous vertical profiles of streamwise wind velocities for the nominal LES snapshot.

The tracer concentration patterns observed in Fig. III.2 illustrate the complexity of the dispersion process:

- Upstream of the obstacle, the perturbed logarithmic profile (at $x = -3.5\ H$) highlight the turbulent nature of the flow. As a result of the flow fluctuations, the plume dispersion upstream of the obstacle is bi-modal: the tracer is either trapped in the recirculation zone on the windward face of the obstacle, or advected downstream of the obstacle.

- Downstream of the obstacle, the flow is driven by a combination of the quasi-periodic vortex shedding induced by the flow-obstacle interaction and the background turbulence propagating from the inlet. The velocity profiles at $x = 5\ H$ and $x = 10\ H$ indicate that a reverse flow occurs near the ground, associated with a second recirculation region, transporting the tracer back towards the obstacle.

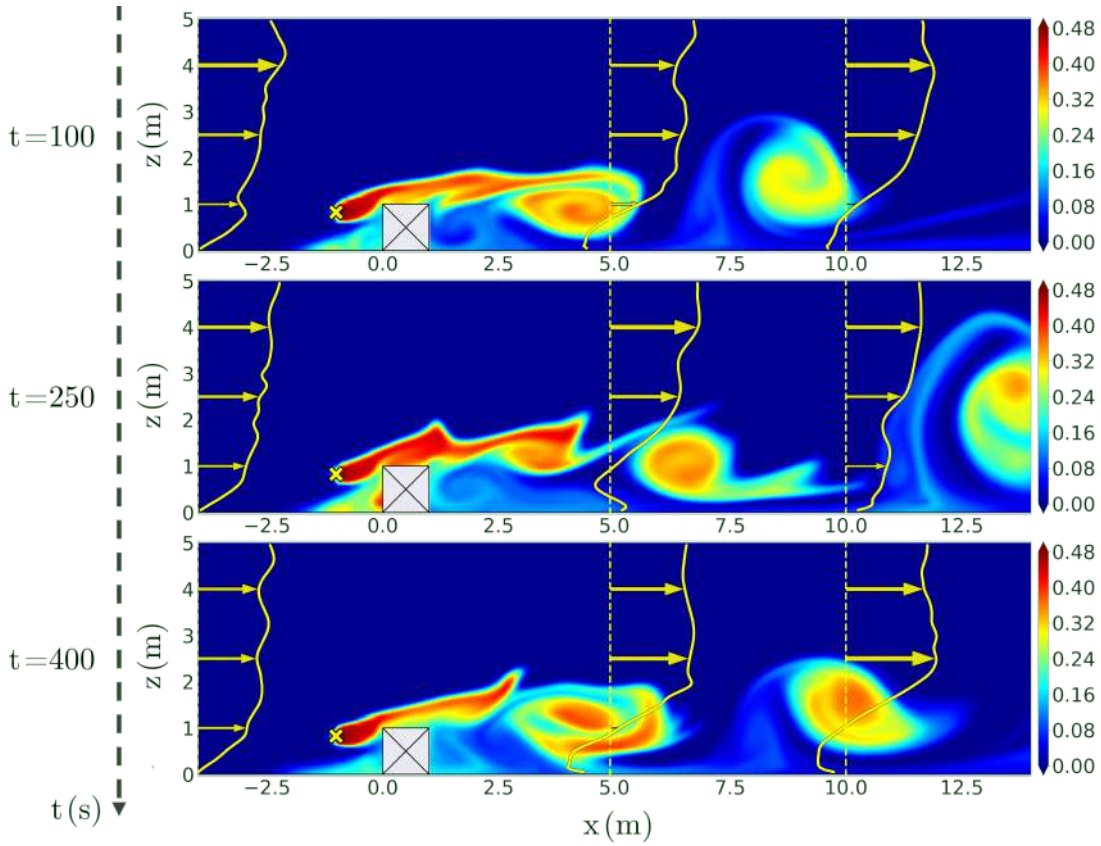

Figure III.2: Instantaneous normalised tracer concentration fields at times $t = 100$, 250 and 400 s from the reference LES with inlet flow parameters $u_{z_c} = 5.78\ \mathrm{m\,s^{-1}}$, $z_0 = 2.79 \times 10^{-2}$ m, and tracer source position coordinates $x_{\mathrm{src}} = -1.01$ m and $z_{\mathrm{src}} = 0.83$ m. Three vertical profiles of instantaneous streamwise velocities at $x = -4$, 5 and 10 m are superimposed on the fields.

These qualitative observations match the typical patterns observed for flow and tracer dispersion in the vicinity a square obstacle. In spite of the two-dimensional representation of turbulence, the numerical results are qualitatively similar to several experiments with bidimensional square section obstacles (infinitely long in the spanwise direction) [Vinçont et al., 2000; Gamel, 2015].

III.1.4.c Flow and tracer concentration field statistics

The time-averaged tracer concentration is the primary output of interest for a dispersion study. Other flow statistics, such as mean airflow statistics or second-order statistics (kinetic energy, momentum and scalar turbulent fluxes), are useful for physical understanding and modelling of plume dispersion. These quantities can be extracted from LES data, as they offer a rich description of the turbulent flow. However, it implies to define a time-averaging window to collect the statistics.

Time-averaging process. Identifying the physical time of simulation necessary to achieve converged LES statistics is one issue when generating a database of parameterised LES snapshots to avoid introducing noise during the reduced-order model training stage.

For each LES snapshot, the initial flow field is specified in adequation with the inflow boundary condition (Sect. III.1.3). A spin-up is then necessary to establish turbulence throughout the computational domain and overcome the initial time transient. Following the spin-up, the LES

model should be run a certain time period to obtain LES field statistics (this is the statistically-stationary phase). This time period depends on the choice of the inflow parameters and therefore on each LES snapshot of the database.

In the present case, the averaging window is based on the characteristic time scale of the vortex shedding. Each LES simulation is run for a total duration corresponding to 40 periods of the vortex shedding, and the first four periods are excluded from the averaging process. For the nominal snapshot, this process leads to 414 s of physical time, which corresponds to about 50 convective times based on the length of the domain and the inlet mean velocity at the height of the obstacle.

First-order statistics. To illustrate the flow complexity around the obstacle, Fig. III.3 shows the line integral convolution (LIC) vector field obtained from the time-averaged airflow for the nominal snapshot. LIC consists of a visualisation technique convolving noise with the vector field $\boldsymbol{u}$ to produce streaking patterns that follow vector field tangents [Cabral and Leedom, 1993]. This figure highlights four specific flow regions associated with (1) clockwise rotation *i*) close to the inlet, *ii*) above the obstacle rooftop, and *iii*) in the wake of the obstacle centred on $(x, z) \approx (5.3\ \text{m}, 1\ \text{m})$; and (2) counter-clockwise rotation in the vicinity of the leeward surface of the obstacle centred on $(x, z) \approx (1.6\ \text{m}, 0.5\ \text{m})$. The topology of the recirculation areas is consistent with the experimental data available in the literature for surface-mounted obstacles with high aspect ratios [Martinuzzi and Tropea, 1993; Vinçont et al., 2000].

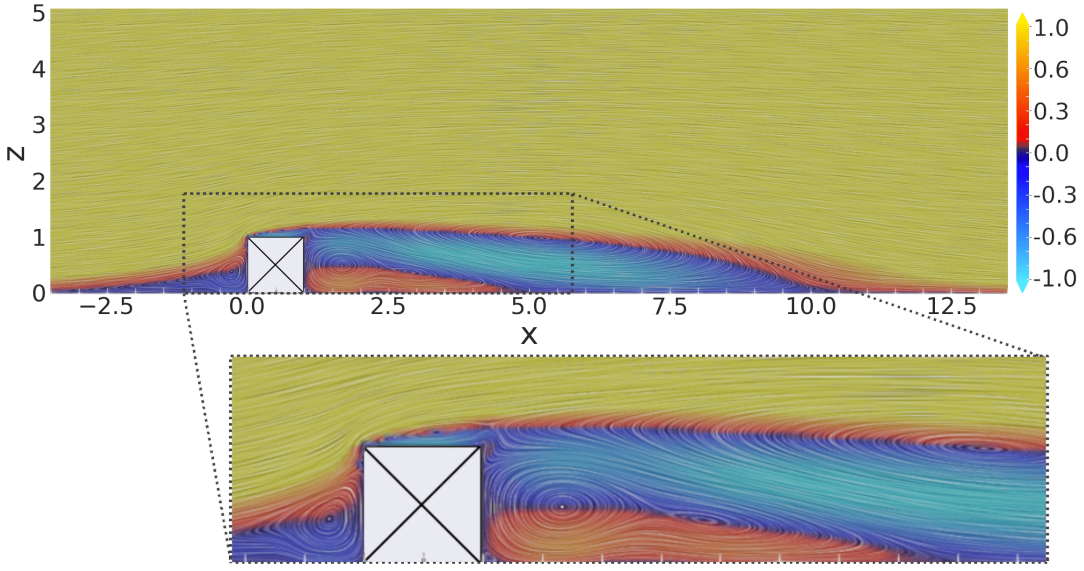

Figure III.3: Line integral convolution (LIC) of the velocity field (white lines) along with horizontal mean velocity ($\text{m}\,\text{s}^{-1}$; background colormap) for the time-averaged reference snapshot corresponding to $U_{z_c} = 5.78\ \text{m}\,\text{s}^{-1}$ and $z_0 = 2.79 \times 10^{-2}$ m.

These complex airflow features are critical for plume dispersion prediction, as the tracer tends to be trapped in recirculation regions. Regarding the tracer field, the features already observed in the instantaneous fields in Fig. III.2 are present in the time-averaged tracer concentration field of Fig. III.4. Close to the windward wall of the obstacle, there is a first tracer accumulation area, with significant tracer concentration close to the ground, which extends to axial locations upstream of the emission sources due to the reverse flow velocity in this region (Fig. III.3).

Above the obstacle top wall, the tracer is deflected and convected downstream, following the deviation of the mean flow induced by the obstacle. In the wake of the obstacle, the plume disperses due to the unsteady motion induced by vortex shedding. A fraction of the tracer plume accumulates in the wake clockwise recirculation zone. Conversely, there is no significant tracer accumulation in the anti clockwise recirculation close to the leeward wall for this case. It is further evidenced in Sect. III.2.2.c that varying the tracer source location leads to different regions of tracer accumulation, highlighting the strong nonlinearity of the concentration field response to the uncertain input parameters.

These downstream features are consistent with what was observed by Philips [2012] (Chapter 3) for an emission source located on a cubical obstacle roof, and by Vinçont et al. [2000] for a release downstream of a high aspect ratio obstacle with a square section.

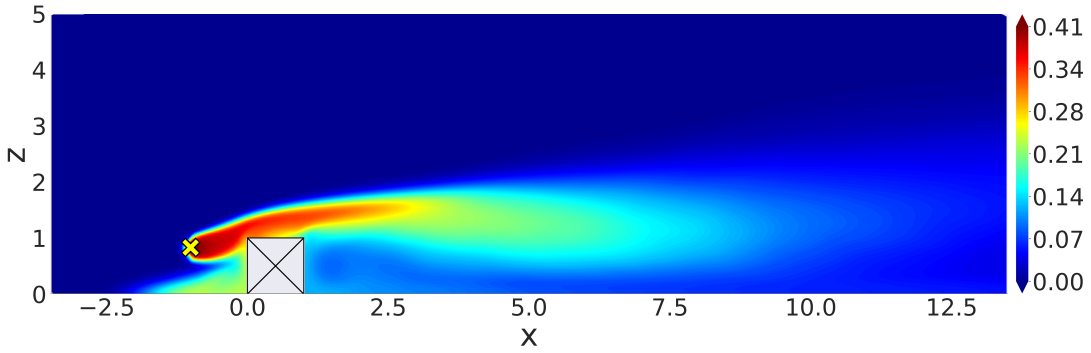

Figure III.4: Time-averaged normalised tracer concentration field for the LES reference snapshot corresponding to $U_{z_c} = 5.78\ \mathrm{m\,s^{-1}}$, $z_0 = 2.79 \times 10^{-2}$ m and $(x_{\mathrm{src}}, z_{\mathrm{src}}) = (-1.01\ \mathrm{m}, 0.83\ \mathrm{m})$.

Second-order flow statistics. LES also provides access to higher-order quantities that are of interest for modelling tracer concentration statistics. For instance, the Reynolds averaging of the momentum and scalar transport equations yields second-order terms associated with correlations between flow features. For the scalar transport equation, the second-order term is the turbulent scalar. Its resolved part obtained with LES is shown in Fig. III.5 for the nominal snapshot. Figure III.5a shows that the streamwise turbulent scalar transport primarily occurs in the streamwise direction above the height of the obstacle, and in the opposite direction in the obstacle wake. As for the vertical turbulent transport in Fig. III.5b, it primarily leads to an upward transport of the scalar.

The two components of the turbulent scalar flux highlight the potential difficulty of the reduced-order models for such quantities, which feature both horizontally elongated patterns and sharp sign changes along critical lines related to flow topology (shear layer, recirculation regions).

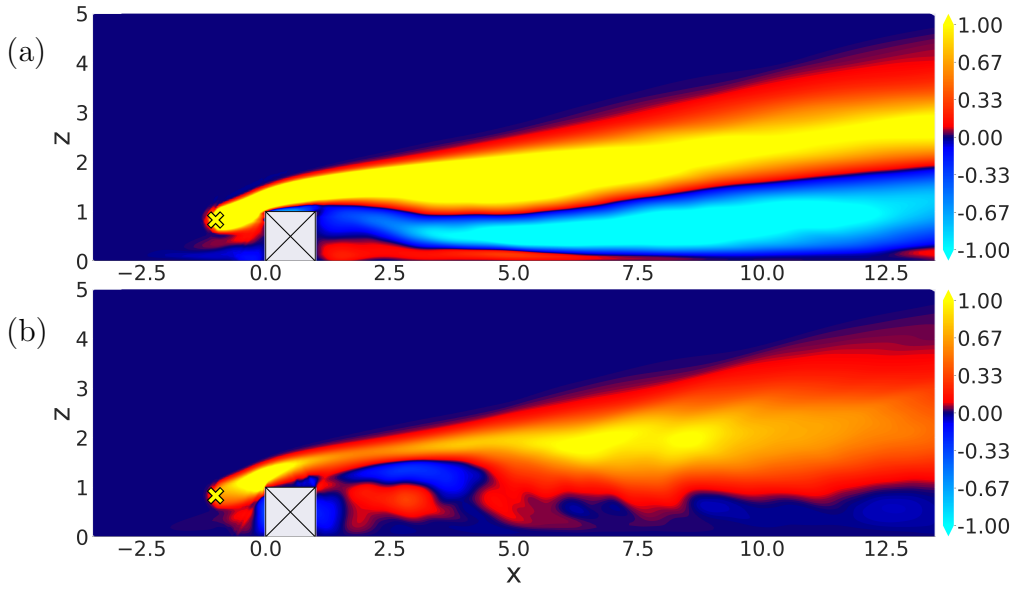

Figure III.5: Resolved turbulent scalar flux components (a) $\overline{u'\mathbf{K}'}$ and (b) $\overline{v'\mathbf{K}'}$, obtained from LES, where $\overline{\cdot}$ denotes the Reynolds time-averaging operator, for the reference snapshot.

To conclude this section, the flow and tracer concentration field statistics are the quantities of interest in this work, implying that we do not consider the time dimension. The objective of this PhD thesis is therefore to design a reduced-order modelling approach that is able to reproduce (or emulate) the spatial variability of these quantities of interest in a range of conditions that are representative of the uncertainties considered. While the reduced-order modelling is primarily focused on the direct prediction of the tracer concentration, as detailed in Chapter IV, the approach is versatile: it is extended to higher order statistics relevant for RANS formalism in Chapter V.

III.2 Uncertainty modelling

In the previous section, we focused on the flow and tracer concentration analysis for the nominal LES simulation snapshot. We now describe how we propagate uncertainties in the LES framework. The starting point is to define the uncertain input parameters that characterise both the inflow conditions and the emission source.

III.2.1 Choice of the uncertain input parameters

In this study, we consider four uncertain parameters: *i)* the reference velocity magnitude u_{z_c} at the reference height $z_c = 10\,H$; *ii)* the aerodynamic roughness length z_0; *iii)* the emission source axial position x_{src}; and *iv)* the emission source height z_{src}. The input vector on uncertainty parameters can be expressed as the four-dimensional vector $\boldsymbol{\mu} = (u_{z_c}, z_0, x_{\text{src}}, z_{\text{src}})^T$. Such parameters have been chosen accordingly with previous uncertainty studies for atmospheric dispersion [García-Sánchez et al., 2014, 2017; Margheri and Sagaut, 2016].

In this uncertainty quantification context, we need to define an interval of variation for each parameter that covers a range of realistic physical conditions. These intervals of variation define the four-dimensional uncertainty space over which we aim to emulate the LES model

response. For this purpose, we need to generate a set (or ensemble) of LES snapshots that sample the uncertainty space. A number of constraints can be prescribed to perform this sampling, including statistical distributions on the parameters to give a probability of occurrence at the different sample points. In this work, uniform distributions are used to remain as generic as possible, except for the roughness parameter z_0 for which the values may vary by several orders of magnitude and the uniform distribution is therefore not appropriate.

III.2.2 Parameter statistical distributions

III.2.2.a Parameterisation of the inlet wind conditions

The uncertain roughness length z_0 and the reference velocity magnitude u_{z_c} impact the mean inlet wind profile, implying that the inlet mean wind profile $u(x_{inlet}, z)$ becomes uncertain.

A wide range of aerodynamic roughness length is considered, ranging from small roughness length corresponding typically to small obstacles between grass (up to 1 cm high), to larger roughness length corresponding to 1-to-2 m high vegetation [Brutsaert, 2013], leading to $z_0 \in [10^{-3}, 10^{-1}]$ m. Its distribution is assumed to be log-uniform with the following probability density function:

$$\log(z_0) \sim \mathcal{U}\left(\log(10^{-3}), \log(10^{-1})\right) , \tag{III.5}$$

corresponding to a mean value $\mathbb{E}[z_0] \approx 0.021$ m and a standard deviation $\sigma(z_0) \approx 0.025$ m (the coefficient of variation, i.e. the ratio of the mean value to the standard deviation, is 1.16). The distribution on z_0 is defined as log-uniform so that the marginal distribution $u(x_{inlet}, z)|u_\tau$ is close to a uniform distribution.

Streamwise velocity magnitude at the reference height u_{z_c} is supposed to follow a uniform distribution so that the marginal distribution $u(x_{inlet}, z)|z_0$ is also uniform:

$$u_{z_c} \sim \mathcal{U}([3, 9])\ \mathrm{m\,s^{-1}} , \tag{III.6}$$

corresponding to a mean value $\mathbb{E}[u_{z_c}] = 6.0\ \mathrm{m\,s^{-1}}$ and a standard deviation $\sigma(u_{z_c}) \approx 1.7\ \mathrm{m\,s^{-1}}$ (the corresponding coefficient of variation is 0.29). The range of variation for u_{z_c} is typical of urban air dispersion studies in the literature [García-Sánchez et al., 2014].

It is of interest to study how the statistics on z_0 and u_{z_c} impact the statistical distribution of u_τ (Eq. III.3). The analytical expression for this distribution is non-trivial because of the inverse of the log-transform of z_0. Still, it can be numerically approximated. Figure III.6 shows the approximate probability density function of u_τ using random Monte-Carlo sampling of Eq. (III.3). This sampling is obtained by perturbing both the roughness length z_0 and the streamwise velocity magnitude u_{z_c} following the statistical distributions defined in Eqs. (III.5)–(III.6). It is worth noting that the reference velocity $u_\tau^{(ref)} = 0.37\ \mathrm{m\,s^{-1}}$ (introduced in Sect. III.1.4) stands as the mean value over the Monte Carlo ensemble (i.e. $\mathbb{E}[u_\tau]$).

The probability distribution on the inlet mean wind profile $\overline{u}(x_{inlet}, z)$ depends on the marginal distributions chosen on the two uncertain parameters z_0 and u_{z_c}. Figure III.7 shows how this distribution varies vertically (i.e. with respect to the vertical axis z). At the reference

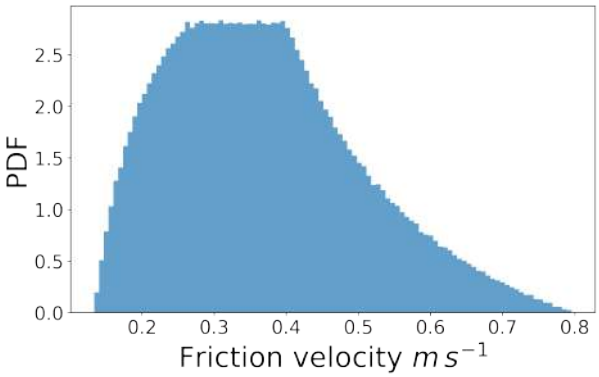

Figure III.6: Approximate probability density function (PDF) of the friction velocity u_τ ($\mathrm{m\,s^{-1}}$) using Monte Carlo random sampling (10^5 samples).

height $z = z_c = 10$ m, it follows a uniform distribution corresponding to the distribution on u_{z_c}. But this is no longer the case when getting closer to the ground surface (i.e. when $z < 10$ m), the probability distribution support decreasing as z decreases. Several inlet mean wind profiles based on varying entries for u_{z_c} and z_0 are also plotted in Fig. III.7 to illustrate the variety of profiles in the LES ensemble.

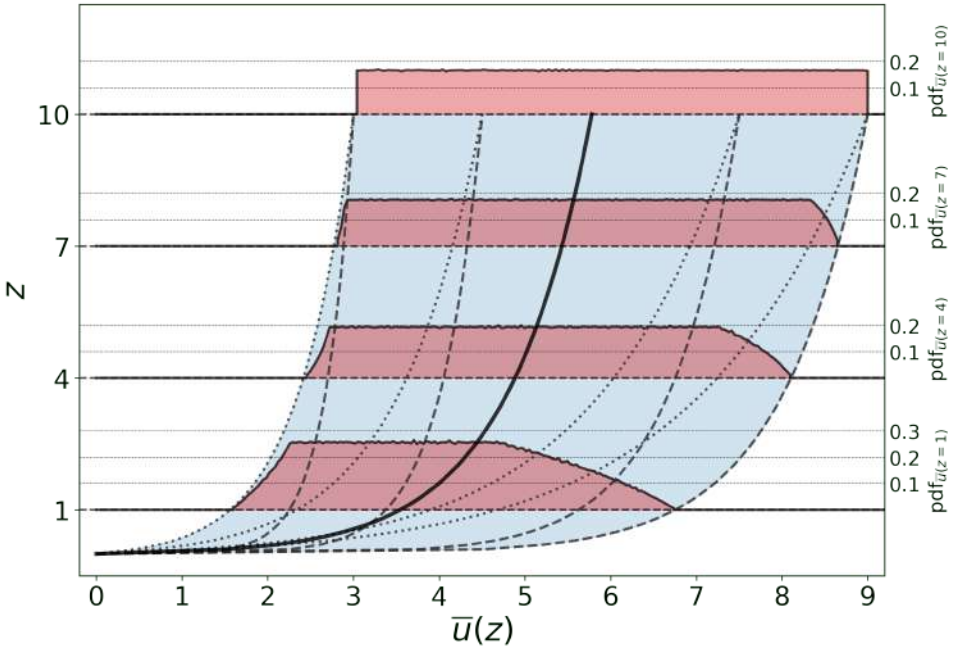

Figure III.7: Probability distribution of the mean streamwise velocity inlet profile $\overline{u}_{inlet}(z)$. The shaded blue area denotes the bounded support of non-zero probability. Streamwise velocity probability distributions are plotted at heights $z = 1$, 4, 7, 10 m. Examples of logarithmic profiles are shown for $z_0 = 10^{-3}$ m (dashed lines) and $z_0 = 10^{-1}$ m (dotted lines). The nominal case (Fig. III.3) profile is also plotted (solid line).

III.2.2.b Parameterisation of the tracer source location

Tracer uncertainty relates to the horizontal and vertical coordinates of the source location $(x_{\mathrm{src}}, z_{\mathrm{src}})$, also referred to as position and height, respectively. In this study, tracer emission can occur upstream, downstream or above the obstacle. The range of variation is chosen so as to cover a broad panel of existing experimental studies [Li and Meroney, 1983a; Mavroidis and

Griffiths, 2000; Mavroidis et al., 2003; Gamel, 2015]. The marginal distributions associated with the source position and height are set uniform as:

$$x_{\text{src}} \sim \mathcal{U}([-3.5, 3.5])\ \text{m}, \quad z_{\text{src}} \sim \mathcal{U}([0.2, 2.0])\ \text{m}. \tag{III.7}$$

The lower limit for the source height is slightly above the ground surface to avoid a truncation of the spatially Gaussian source term. The area $[0, 1.2]\ \text{m} \times [-0.2, 1.2]\ \text{m}$ is removed from the range of variation to avoid having a source inside the obstacle.

III.2.2.c Diversity of flow and tracer field topologies in the ensemble

The wide intervals retained for the parameter distributions result in a large diversity of flow and tracer concentration fields. It is illustrated here through three snapshots picked from the ensemble of LES snapshots, in order to highlight the complexity of the response to be emulated by a reduced-order model.

Figure III.8 shows examples of normalised tracer concentration and streamwise velocity mean fields for: (a) the nominal case, (b) the case of a near-ground source emission located downstream of the obstacle within a recirculation area with moderate wind conditions ($U_{z_c} = 5.79\ \text{m}\,\text{s}^{-1}$, $z_0 = 7.89 \times 10^{-3}$ m), and (c) the case of a high source emission where there is no significant influence of the obstacle on the dispersion and where the wind conditions are stronger ($U_{z_c} = 7.45\ \text{m}\,\text{s}^{-1}$, $z_0 = 1.3 \times 10^{-3}$ m). Significant variations are observed for both the velocity and tracer fields. Regarding the velocity field, a critical feature is the reattachment point, which corresponds to the interaction of zero axial velocity isoline with the ground. Its location varies significantly with the inlet parameters: it ranges from $x = 10\,H$ in Fig. III.8ab, up to $x = 13.5\,H$ in Fig. III.8c. Regarding the tracer dispersion, for the nominal case (Fig. III.8a), the tracer accumulates in two of the three main recirculation regions, as previously discussed. For the second case (Fig. III.8b), the tracer primarily accumulates in the recirculation region near the leeward face of the obstacle, while there is no significant interaction between the plume and the recirculation regions for the third case (Fig. III.8c).

This diversity of flow topologies and tracer concentration distributions evidences the challenges in constructing reduced-order models as changing the input parameters may induce nonlinear changes in the field quantities of interest. This implies that the mapping between the input parameters and the field statistics that we seek to learn with the reduced-order models may be subject to significant nonlinearities.

III.2.3 Large-eddy simulation database

III.2.3.a Synthesis of the input-output mapping problem

The uncertainties are described by uncertain scalar parameters that are considered as inputs to the LES problem and that form the input vector parameter:

$$\boldsymbol{\mu} = (\boldsymbol{\mu}_{atm}, \boldsymbol{\mu}_{tr})^T = (u_{z_c}, z_0, x_{src}, z_{src})^T \in \mathbb{R}^4. \tag{III.8}$$

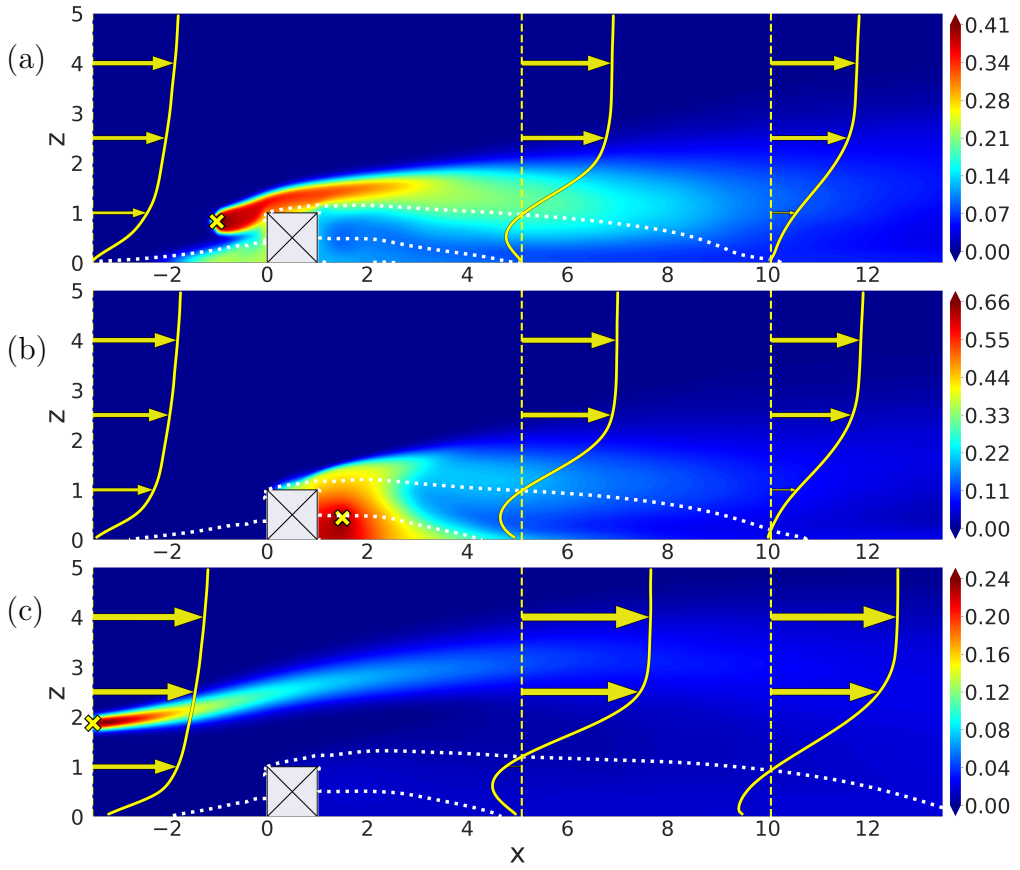

Figure III.8: Time-averaged normalised tracer concentration field with superimposed time-averaged streamwise velocity vertical profiles at abscissa $x = -4$, 5 and 10 m (yellow vertical solid lines) and zero-velocity magnitude contour lines (white dotted lines) for three LES snapshots of the LES test database: (a) the nominal snapshot (associated with Figs. III.2 to III.5) corresponding to $U_{z_c} = 5.78\ \mathrm{m\,s^{-1}}$, $z_0 = 2.79 \times 10^{-2}$ m and $(x_{\mathrm{src}}, z_{\mathrm{src}}) = (-1.01\ \mathrm{m}, 0.83\ \mathrm{m})$; (b) the snapshot with $U_{z_c} = 5.79\ \mathrm{m\,s^{-1}}$, $z_0 = 7.89 \times 10^{-3}$ m and $(x_{\mathrm{src}}, z_{\mathrm{src}}) = (1.5\ \mathrm{m}, 0.44\ \mathrm{m})$; and (c) the snapshot with $U_{z_c} = 7.45\ \mathrm{m\,s^{-1}}$, $z_0 = 1.3 \times 10^{-3}$ m and $(x_{\mathrm{src}}, z_{\mathrm{src}}) = (-3.49\ \mathrm{m}, 1.86\ \mathrm{m})$.

The input parameters impact the simulated flow response and drive the quantities of interest $\mathbf{K}_{\mathrm{les}} = \{K_1, \ldots, K_{N_h}\}^T \in \mathbb{R}^{N_h}$. The following Chapters IV–V investigate several quantities of interest depending on the context. In Chapter IV, the term $\mathbf{K}_{\mathrm{les}}$ corresponds to the time-averaged tracer concentrations predicted at the N_h grid points of the discretised computational domain (the nominal snapshot example is given in Fig. III.8a). In Chapter V, various quantities are targeted, such as the streamwise and vertical components of the flow velocity, the turbulent kinetic energy k_{tke}, as they are re-used in a lower fidelity model for dispersion.

The LES full-order model prediction of the quantities of interest comes a significant computational cost, which prevents real-time outputs and requires extensive computational resources (the average cost of one simulation is about 800 CPU hours). These issues motivate the use of a reduced-order model, which can result in significant speedups to predict the quantities of interest for any new value of the input vector $\boldsymbol{\mu}$. Therefore, the objective of the reduced-order model is to approximate the LES model response based on a collection of N snapshots $\{\mathbf{K}_{\mathrm{les}}^{(1)}, \cdots, \mathbf{K}_{\mathrm{les}}^{(N)}\}$ corresponding to a collection of N input sets $\{\boldsymbol{\mu}^{(1)}, \cdots, \boldsymbol{\mu}^{(N)}\}$ with $N \ll N_h$. One challenge for training and validating the reduced-order model is that the number of snapshots N remains limited in the context of LES (from ten to one hundred for a practical three-dimensional case).

III.2.3.b Sampling strategy

In this work, we build a large dataset of LES snapshots (750 LES in total that can be considered as untractable for a LES application on a real dispersion case, but still remains limited from a statistical learning perspective) in order to train and carefully evaluate the different formulations of the reduced-order model. This ensemble is obtained from sampling the input parameter vector $\boldsymbol{\mu}$ (Eq. III.8) on the probability space $(\Omega, \mathcal{P})$, where Ω represents the four-dimensional uncertainty space (i.e. the set of all possible samples) and $\mathcal{P}$ represents the associated probability measure. The coverage of the space Ω requires samples to be "well" distributed, i.e. to minimise the distance to the closest sample with respect to the related distribution $\mathcal{P}$. This step is of primary importance as the samples are the datasets to train and validate the reduced-order models in a reliable way.

Notion of discrepancy. Mathematically, the uncertainty space coverage by the samples is characterised using the notion of discrepancy. The discrepancy is low if the local distribution of the samples is close to the probability measure $\mathcal{P}$. For the simple case where Ω is the one-dimensional interval $[0, 1]$ and $\mathcal{P}$ is the uniform probability measure, the discrepancy is expressed as:

$$D_N(\{\boldsymbol{\mu}^{(1)}, \dots, \boldsymbol{\mu}^{(N)}\}) = \sup_{0 \leq a < b \leq 1} \left| \frac{\#\{\mu_i \in [a, b[\}}{N} - \lambda([a, b[) \right|, \tag{III.9}$$

where $\{\boldsymbol{\mu}^{(1)}, \dots, \boldsymbol{\mu}^{(N)}\}$ is the N-ensemble of uncertain parameters, $\#$ is the cardinal of the related ensemble, and λ is the Lebesgue measure on $[0, 1]$. The discrepancy corresponds to the largest difference between the theoretical distribution and its related discrete approximation. D_N is the optimal value but is not achievable for continuous probability measures with a finite number of samples. It is therefore of primary importance to use a sampling strategy that minimises discrepancy.

In this work, due to the significant computational cost of a given LES, the sampling strategy should allow to build the LES database in an incremental way. It should be easy to add new samples and update the reduced-order model as the LES database grows. A counter-example is the rectangle strategy to sample the interval $[0, 1]$: for $N = 2$, the optimal set is $\{\mu^{(1)}, \mu^{(2)}\} = \{1/3, 2/3\}$; adding a new point changes the whole dataset as for $N = 3$ the new set becomes $\{\mu^{(1)}, \mu^{(2)}, \mu^{(3)}\} = \{1/4, 1/2, 3/4\}$. This implies that increasing the dataset size requires to recomputing all the samples (which would be impractical in the context of LES). Although randomised Monte Carlo approaches would allow cheaper resampling, they may suffer from discrepancy issues.

Low-discrepancy sequences. Low-discrepancy sequences, also known as quasi Monte Carlo sampling approaches, are designed to have a lower asymptotic discrepancy than in the purely Monte Carlo random approach. In a one-dimensional case, a Van der Corput sequence can be considered as a low-discrepancy sequence. It generates in a deterministic way, a series of numbers over the unit interval that has a low discrepancy in the context of a uniform probability measure [Lemieux, 2009]. The nth sample value of the sequence $\mu^{(n)}$ is computed from the integer

decomposition $(a_k(n))_k$ of n in base b:

$$\mu^{(n)} = \sum_{k=0}^{+\infty} a_k(n)\, b^{-k-1}, \quad n = \sum_{k=0}^{+\infty} a_k(n)\, b^k. \tag{III.10}$$

For instance, the first seven Van der Corput samples for $b = 2$ are $\{1/2,\ 1/4,\ 3/4,\ 1/8,\ 5/8,\ 3/8,\ 7/8\}$.

Halton [1960] generalises the Van des Corput sequences to higher dimensions [Lemieux, 2009]. To do so, each dimension is associated with a prime number. For instance, in a two-dimensional case, the first dimension is related to $b = 2$ and the second dimension to $b = 3$. This simple trick prevents the samples from being located on the hypercube diagonals. Halton sequences allow a simple resampling of the uncertainty space, while guaranteeing a low discrepancy whatever the number of samples N. It is known that its asymptotic discrepancy is of the order of $\mathcal{O}(\frac{(\log(n)^d}{n})$, where d is the space dimension. For comparison, the discrepancy for a uniform sequence is of the order of $\mathcal{O}(n^{-1/d})$ and of $\mathcal{O}(\frac{\sqrt{\ln \ln n}}{\sqrt{n}})$ for a fully random Monte Carlo sampling [Lemieux, 2009]. In practice, Halton sequences are known to perform well when the uncertain dimension d is small. In this work, the dimension is $d = 4$, which can be considered as small. For this reason, we adopt the Halton sequence to sample the input vector parameter $\boldsymbol{\mu}$.

Extension to non-uniform sampling. In its simplest form, for a given dimension d, the Halton sequence provides a uniform sampling of the hypercube $[0, 1]^4$. When the uncertain parameter of interest varies along an interval $[a, b]$, the Halton samples can be transformed using affine transformation. Indeed, a non-uniform sampling may be easily recovered from the Halton sequence using the inverse transformation method when the cumulative distribution function (CDF) is available. In this work, for instance, the roughness parameter z_0 follows a log-uniform distribution in the interval $[10^{-3}, 10^{-1}]$. Let F_{z_0} denote its CDF, and U be a random variable satisfying $U \sim \mathcal{U}([0, 1])$. The random variable Z defined as

$$Z = F_{z_0}^{-1}(U), \text{ with} \qquad F_{z_0}^{-1} : u \mapsto 10^{-1} \left(\frac{10^{-3}}{10^{-1}}\right)^u, \ \forall\ u \in [0, 1], \tag{III.11}$$

follows a log-uniform distribution on $[10^{-3}, 10^{-1}]$. Consequently, the samples for z_0 satisfying the target log-uniform distribution over the interval $[10^{-3}, 10^{-1}]$ can be directly obtained from a Halton sequence on the interval $[0, 1]$ using the $F_{z_0}^{-1}$ transformation.

To conclude this section, the objective of the reduced-order models is to represent how the LES flow and tracer concentration field statistics (in particular the mean tracer concentration field and the mass flux tensor) respond to perturbations in four input parameters, including the reference velocity magnitude u_{z_c} and the aerodynamic roughness length z_0 that impact the inflow streamwise velocity vertical profile and the emission source location (x_{src}, z_{src}). A large ensemble of LES snapshots (made of 750 simulations in total) has been generated based on Halton's low-discrepancy sequence of the four parameters. These parameters are mostly characterised by uniform statistical distributions to remain as generic as possible, except for z_0 that follows a log-normal distribution. This leads to a diversity of flow and tracer concentration conditions in the ensemble that is challenging to emulate.

III.3 Practical implementation

III.3.1 Ensemble large-eddy simulation management

Generating an ensemble of LES snapshots requires to run multiple instances of the AVBP code, each instance corresponding to a set of parameters $\boldsymbol{\mu}$ and being supervised by the Lemmings simulation manager developed by CERFACS. Lemmings[2] is an open-source Python code designed to simplify the submission of multiple dependent jobs on HPC cluster schedulers. It eases the simulation workflow for repetitive and sequenced steps by automating a certain number of tasks such as job chained submissions (e.g. until the required physical simulation time is reached) or preprocessing, model integration and postprocessing tasks. This Lemmings workflow is very useful for this work as it facilitates the generation of the large LES database: all simulation instances share the same structure, as only the uncertain parameters (inflow boundary condition and tracer emission source location) vary from one simulation to another.

In terms of computational cost, the average CPU cost for one LES simulation is 800 CPU hours, which requires a total budget of 600,000 CPU hours to generate all 750 LES snapshots. A three-dimensional computation is expected to be an order magnitude higher in terms of cost: it justifies the restriction to a two-dimensional setup for this study, which focuses primarily on the reduced-order modelling method rather than on a detailed investigation of the physics.

The LES runs have been performed on CERFACS Kraken supercomputer, a total of 72 core processors (Intel Skylake architecture) has been used for each LES run.

III.3.2 Machine- and deep-learning librairies

The emergence of open-source community libraries enables the relatively straightforward implementation of most conventional machine and deep-learning model formulations. Table III.1 summarises the various statistical libraries considered in this PhD thesis, as well as the algorithms, processing units (CPUs/GPUs), and primary features assessed. The GPUs used are Nvidia Volta V100/16GB.

[2]See Lemmings webpage at https://gitlab.com/cerfacs/lemmings

Table III.1: Overview of the machine-learning and deep-learning approaches used in this work.

Category	Tool	Library	Architecture	Hyperparameters
Sampling	Halton	OpenTURNS	CPU	
Dimension reduction	POD	Scikit-Learn	CPU	
	Autoencoder	Keras–Tensorflow	GPU	Activation functions, nb. of layers, layer depth, learning rate
Regression	Gradient boosting	Scikit-Learn	CPU	trees depth, tolerance, learning rate, pruning, nb. of trees
	k-nearest neighbours	Scikit-Learn	CPU	nb. of neighbours, norm, weight function
	Gaussian processes	Scikit-Learn /SMT /GPyTorch	CPU /CPU /GPU	kernel, learning rate
	Polynomial chaos	OpenTURNS	CPU	truncation strategy, cleaning strategy, maximum polynomial degree

The Python package Scikit-Learn provides implementations for both dimension reduction tools (e.g. POD), regression models (e.g. Gaussian processes, linear regression models, decision trees with boosting technique) and many other machine-learning algorithms [Pedregosa et al., 2011]. In particular, data preprocessing and model selection algorithms largely facilitate the deployment of reduced-order modelling pipelines. For Gaussian processes, although Scikit-Learn provides a fairly standard implementation sufficient for preliminary investigation, we also used GPyTorch and Surrogate Modeling Toolbox (SMT) [Gardner et al., 2018; Bouhlel et al., 2019]. Scikit-Learn only supports CPUs, and for large datasets, GPyTorch provides considerable GPU acceleration for model training. GPyTorch also proposes other advanced Gaussian process formulations, e.g. Gaussian process latent variable models (GPLVM) for dimension reduction. The SMT library was used for co-kriging implementation in the multi-fidelity framework proposed in Chapter V.

Among the various models we tested in this work, only the neural networks and the polynomial chaos expansion are not supported by the Scikit-Learn package. We used Keras–Tensorflow for the implementation of neural networks using GPUs [Chollet et al., 2015]. OpenTURNS, as Scikit-Learn, is a fairly exhaustive package. It is used to implement polynomial chaos expansion and to perform Halton's sampling.

Chapter IV

Reduced-order model for mean tracer prediction based on LES data

In this chapter, we aim at building and learning a reduced-order model that can directly emulate LES field statistics. Building such a reduced-order model is a two-step process that includes a dimensionality reduction component and a regression component. Proper orthogonal decomposition (POD) is used as the baseline approach for dimension reduction and is compared to a more advanced deep-learning approach based on convolutional autoencoder. Several families of regression models among whom polynomial chaos expansion, Gaussian process regression and gradient tree boosting are implemented and optimised to identify which is the best strategy for the pollutant dispersion test case studied in this work. The mean tracer concentration field is used as the quantity of interest to assess the potential of each approach to emulate the LES field statistics.

Contents

IV.1 Construction and evaluation strategy of the reduced-order model

The objective is to build a reduced-order model that can leverage LES data in such a manner that patterns may be learned and reproduced on new unseen data. The reduced-order model combines a dimension reduction component with a regression model as detailed in Chapter II to represent how the mean tracer concentration fields $\mathbf{K}_{\text{les}}$ change with respect to the uncertain input parameters $\boldsymbol{\mu}$.

IV.1.1 Dataset acquisition

In this study, we consider different sources of uncertainty in the LES full-order model: *i)* uncertainties associated with the large-scale atmospheric flow conditions $\boldsymbol{\mu}_{atm}$ (affecting the inflow and surface boundary conditions), and *ii)* uncertainties with the tracer emission source characteristics $\boldsymbol{\mu}_{tr}$. These uncertainties are described by uncertain scalar parameters that are considered as inputs to the LES problem (Sect. III.2), and that form the input vector parameter:

$$\boldsymbol{\mu} = (\boldsymbol{\mu}_{atm}, \boldsymbol{\mu}_{tr})^T = (u_{z_c}, z_0, x_{\text{src}}, z_{\text{src}})^T \in \mathbb{R}^4. \tag{IV.1}$$

The input parameters impact the simulated flow response and drive the quantities of interest $\mathbf{K}_{\text{les}} = \{K_1, \ldots, K_{N_h}\}^T \in \mathbb{R}^{N_h}$, namely means of tracer concentrations predicted at the N_h grid elements of the discretised computational domain (the nominal snapshot example is given in Fig. III.4).

We build a dataset of 750 LES snapshots in order to train and carefully test our reduced-order model. Each snapshot of the dataset corresponds to different random sample of the input vector $\boldsymbol{\mu} = (u_{z_c}, z_0, x_{\text{src}}, z_{\text{src}})^T$. The samples are obtained using Halton's low-discrepancy sequence. Moreover, all LES were conducted on the same numerical setting (same grid, convection scheme, LES model, etc.). Since the LES setup is robust, the non-intrusive procedure is well decoupled from the LES model to only handle the parametric variability.

IV.1.2 From training to validation

The algorithm we use to train and validate the reduced-order model can be summarised as the following training/prediction two-step process.

– **Training stage (learning)**

1. Extract the modes $\{\boldsymbol{\psi}_l\}_{l=1,\ldots,L}$ using encoder transformation (POD or convolutional autoencoder) and truncate the reduced basis to the first L modes from the mean tracer concentration fields $\mathbf{K}_{\text{les}}$ in the training dataset,

2. Train the regression model's parameters using the training dataset,
3. Choose the best values of the hyperparameters to maximise the regression model's performance from the validation dataset.

– **Prediction stage (validation)**

1. Compute the reduced coefficients $\{k_l(\boldsymbol{\mu}^*)\}_{l=1,\dots,L}$ for the test sample of input parameters $\boldsymbol{\mu}^*$ using the regression model,
2. Perform decoder transformation (inverse POD or convolutional autoencoder) to recover the predicted mean tracer concentration fields $\mathbf{K}^*_{\text{rb}}$ from the reduced coefficients for the test samples,
3. Compare the emulated fields $\mathbf{K}^*_{\text{rb}}$ with the reference LES test snapshots $\mathbf{K}^*_{\text{les}}$.

The full LES dataset (made of 750 snapshots, as described in Sect. III.2) is split into three subsets following Halton's sequence ordering (see explanations in Sect. II.4): *i)* a training dataset made of $N_{\text{train}} = 450$ snapshots (60% of the full LES dataset) to learn the dimension reduction model and the mapping between the uncertain inputs $\boldsymbol{\mu}$ and the reduced coefficients $\boldsymbol{k}$ using regression models; *ii)* a calibration dataset made of $N_{\text{calib}} = 150$ snapshots (20% of the full LES dataset) to investigate multiple hyperparameter configurations within each family; and *iii)* a test dataset made of $N_{\text{test}} = 150$ snapshots (20% of the full training dataset) to evaluate the capacity to predict LES quantities of interest for new samples of the uncertain input parameters.

IV.1.3 Performance metrics

The choice of the performance metric is essential for the machine learning framework as it introduces bias to favour a specific desired behaviour of the learning model. In low-dimension, mean-squared error (MSE) is a suitable function for assessing the quality of a model as it incorporates both variance and bias contributions [Wackerly et al., 2014]. However, MSE appears unsuitable for ordering comparisons on quantities with different variance magnitudes, which is the case of time-averaged tracer concentration that can vary by several orders of magnitude between two grid points. A simple solution to this problem is to normalise the MSE by the variance estimation. This is called the explained variance criterion, also denoted by Q^2.

Individual model performance evaluation. In this work, we quantify individual model performance using a Q^2 (explained variance) criterion applied to reduced-coefficient prediction. We highlight here the example of Q^2 performance for POD reduced-coefficient prediction using regression models. With this criterion, the prediction error of the lth regression model is weighted by the variance over the reduced coefficients $\boldsymbol{k}_l$ of the lth POD mode:

$$Q^2_l = 1 - \frac{\|\boldsymbol{k}_l - \boldsymbol{m}^*_l\|^2_2}{\|\boldsymbol{k}_l - \hat{\mathbb{E}}[\boldsymbol{k}_l]\|^2_2}\,,\ \forall l = 1, \dots, L, \tag{IV.2}$$

where $\boldsymbol{m}^*_l$ is the lth regression model mean prediction (e.g. Eq. II.39 for Gaussian process regression). When dealing with POD, since we use whitening, reduced coefficients have zero-

mean and unit-variance, meaning that this per-mode Q^2 metric directly reflects the MSE on the predicted reduced coefficients (i.e. $Q_l^2 \approx 1 - \text{MSE}$).

Reduced-order model performance evaluation. To quantify the performance of the reduced-order model in the physical space, we evaluate the Q^2 criterion on each feature (on each grid point) of the domain of interest. The reconstruction/prediction error is weighted by the variance over the LES snapshots for each grid element i:

$$Q_i^2 = 1 - \frac{\|\mathbf{K}_{\text{les},i} - \mathbf{K}_{\text{rb},i}\|_2^2}{\|\mathbf{K}_{\text{les},i} - \widehat{\mathbb{E}}[\mathbf{K}_{\text{les},i}]\|_2^2}\,,\ \forall i = 1, \ldots, N_h. \tag{IV.3}$$

For instance, if $\mathbf{K}_{\text{rb},i}$ stands as the standalone POD reconstruction of LES snapshots, Q_i^2 carries the performance of the encoding/decoding process on the ith grid point. $\mathbf{K}_{\text{rb},i}$ may also be replaced by reduced-order model prediction $\widehat{\mathbf{K}}_{\text{rb},i}$ to quantify prediction performance (including encoding/decoding and regression operations). Thus, the difference in the Q^2 criterion calculated on the standalone decoding/decoding process and on the complete prediction process allows to quantify the error introduced by the standalone regression process. In this work, final assessment of Q_i^2 on the LES calibration dataset helps selecting optimal model's hyperparameters, while assessment on the test dataset evaluates the reduced-order model generalisation capacity.

To help with the analysis, we also derive a global score from the variance weighted local Q^2 criterion as:

$$Q_{\text{global}}^2 = \sum_{i=1}^{N_h} \omega_i\, Q_i^2, \quad \omega_i = \frac{\widehat{\mathbb{V}}(\mathbf{K}_{\text{les},i})}{\sum_{j=1}^{N_h} \widehat{\mathbb{V}}(\mathbf{K}_{\text{les},j})}, \tag{IV.4}$$

where $\widehat{\mathbb{V}}(\mathbf{K}_{\text{les},i})$ corresponds to the variance unbiased estimation over the snapshots defined in Eq. (II.10). It is worth noting that Q_{global}^2 can also be derived on subdomains to study the spatial variability of the reduced-order model error.

The Q^2 metric is homogeneous with the MSE taken over all grid points and normalised by the ensemble variance:

$$Q_{\text{global}}^2 = 1 - \frac{\sum_{k=1}^{N_h} ||\mathbf{K}_{\text{les},j} - \hat{\mathbf{K}}_{\text{les},j}||_2^2}{(N-1)\sum_{j=1}^{N_h} \hat{\mathbb{V}}(\mathbf{K}_{\text{les},j})}. \tag{IV.5}$$

It can also be proven that Q_{global}^2 applied to POD reconstruction is equivalent to the POD cumulative explained variance ratio. Indeed, when $\mathbf{K}_{\text{rb},i}$ matches snapshots reconstructed from the first l modes, Q_{global}^2 can be expressed as $Q_{\text{e.v.}}^2$ (Eq. II.11).

IV.1.4 Domain of interest

Because of the directional nature of plume flow, the dispersion phenomena are localised to a subregion of the domain. To minimise statistical modelling efforts, the area of interest is restricted on the x-axis to the interval $[-3.5, 13.5]$ m, containing 99.9% of the overall ensemble variance (this area encloses the space of uncertainty on the source location position x_{src} and

height z_{src}). This is equivalent to keeping all grid points with a variance greater than 6.4×10^{-7}. Figure IV.1 shows the mean and variance ensemble statistics estimated from the 750 LES snapshots but restricted on the subdomain of interest.

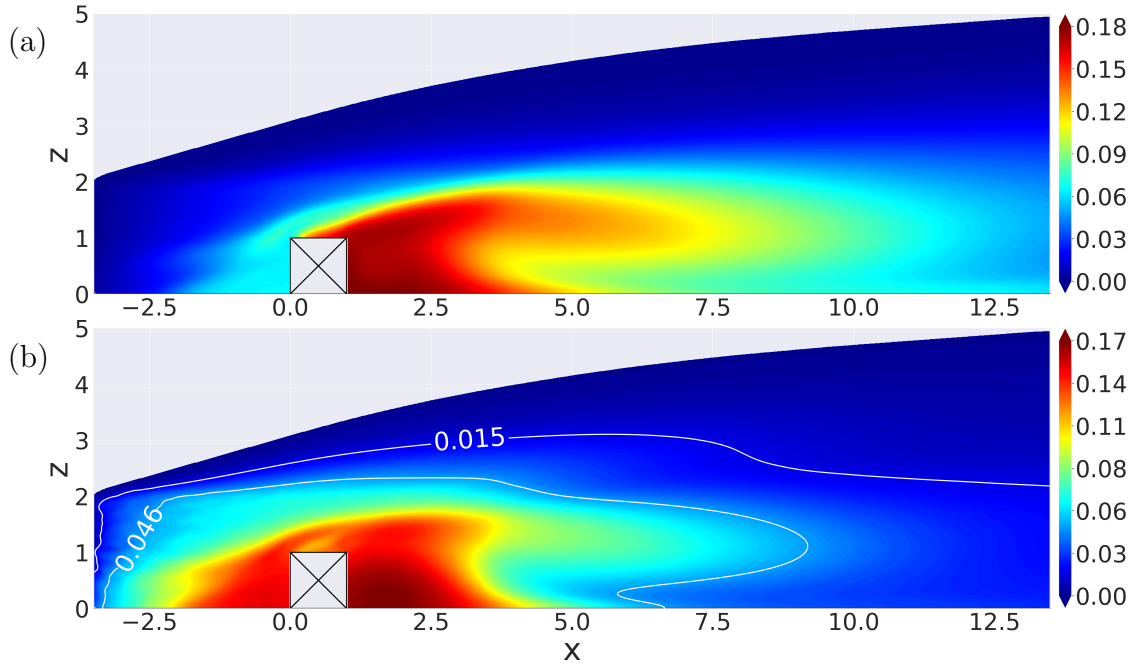

Figure IV.1: Ensemble statistics for the mean tracer concentration fields obtained over the full LES dataset (750 snapshots) on the restricted subdomain of interest: (a) ensemble mean, (b) ensemble standard deviation with $\frac{1}{3}$- and $\frac{2}{3}$-quantiles denoted by white solid lines.

The domain restriction focuses attention on the grid points well represented by the LES database snapshots. Outside of the restricted region, the variance reported by the LES ensemble is dominated by numerical noise. The quality of the representation is essential for building a machine learning framework and quantifying performance. For this reason, it is essential to discard areas where learning data are already known to be inadequate.

From the $\frac{1}{3}$- and $\frac{2}{3}$-quantiles of ensemble variance (Fig. IV.1b), we identify three subdomains of equal number of grid points, sorted by variance quantiles. In the following, the subregion of local variance nodes below the $\frac{1}{3}$-quantile will be noted as T_0, the subregion of medium-range variance between the $\frac{1}{3}$- and $\frac{2}{3}$-quantiles as T_1, and the high-variance nodes as T_2.

IV.2 Performance evaluation of proper orthogonal decomposition

In this section, POD is used as the baseline approach to reduce the dimension of the mean tracer concentration fields obtained with LES. We investigate here the impact of truncating the POD decomposition at the first L modes on the representation of the ensemble variance and on the reconstruction of the LES fields. Recall that the number of POD modes directly determines the number of regression models to train (Eq. II.2).

IV.2.1 Impact of reduced-basis truncation

Statistical evaluation for the training dataset. Several criteria are available in the literature for selecting the number of modes, the vast majority relying on the explained variance metrics

evaluated on training data (e.g. the cumulative explained variance, the Kaiser and elbow rules – Jolliffe, 2002). Figure IV.2 shows the cumulative explained variance (Eq. II.11) of the full training dataset (made of 450 LES snapshots) when decomposed on the POD reduced basis as a function of the L POD modes. The first mode alone contributes to about 50% of the explained variance on the whole computational domain, the first fifteen modes explain more than 95% and the first hundred modes about 99.7%. Alternatively, the Kaiser rule applied to 70% of the mean eigenvalue suggests keeping only 23 modes; this corresponds to 96.3% of the total ensemble variance. The elbow rule estimates the truncation threshold from the sign change in the eigenvalue second-order derivative with respect to the mode index: only the first fives modes shall be kept following this rule, which corresponds to 85.6% of the total ensemble variance. This first comparison shows that these different truncation rules lead to very different number L of modes. A finer analysis of the POD modes is necessary to determine an appropriate truncation level L.

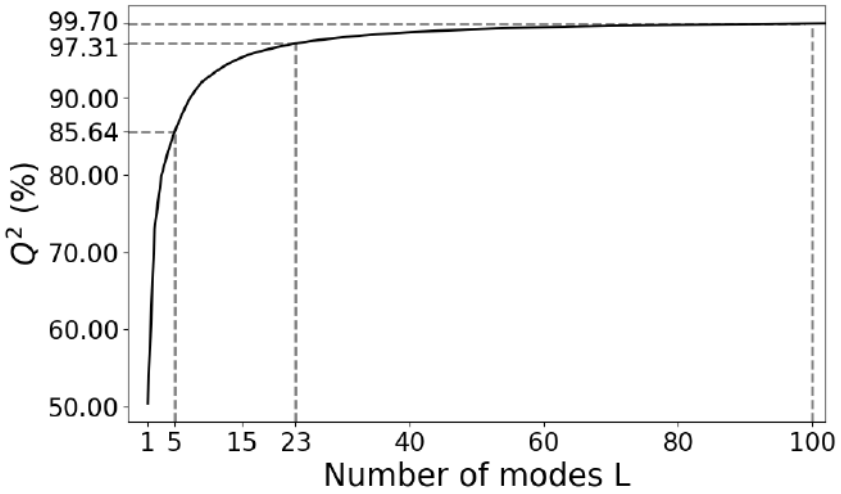

Figure IV.2: Cumulative explained variance (Q^2 in %, see Eq. II.11) for the POD reduced-basis size L varying between 1 and 100 (solid line). The truncation thresholds for the Kaiser rule ($L = 23$) and the elbow rule ($L = 5$) are represented in vertical dashed lines.

Statistical evaluation for the test dataset. The cumulative explained variance computed over the test dataset (using Eq. IV.5) for $L = 100$ modes is globally equal to $Q^2 = 99.3\%$, which is slightly smaller than for the training set (Table IV.1). This implies that POD slightly overfits data. However, not all areas of the domain are equally affected by overfitting. While the explained variance criterion is useful for characterising the overall efficiency of POD reconstruction, it may be worthwhile to investigate local disparities of the explained variance. Figure IV.3 shows the local capability of POD to represent the ensemble variance using a Q^2 metric on each grid point of the computational domain (Eq. IV.3).

– Figure IV.3ab compares the spatial performance of POD reconstruction on the (a) training and (b) test datasets using $L = 100$ modes. We can observe that the reconstruction error is very low over most of the domain but that there is a sharp drop in performance upstream of the obstacle at the domain edge. This drop in performance is stronger for the test dataset than for the training dataset.

This is consistent with POD ensemble variance maximisation behaviour. Far from the ground and from the obstacle, the number of representative snapshots (i.e. plumes flowing in this area) is reduced because of the distribution of emission source position and height. These grid points are characterised by small ensemble variance, which drastically penalises the Q^2 criterion and explains the sharp drop in performance spatially.
This is confirmed by Table IV.1, indicating that over the lowest variance area the average Q^2 is equal to 98.5% for the training dataset, while it is equal to 91.6% for the test dataset. In the contrary, the average Q^2 remains above 98% for the medium and largest variance regions even for the test dataset, with differences of the order of 1% or less between the training and test datasets.

– Figure IV.3cd shows the POD reconstruction error for varying number L of modes. We can observe that decreasing the number of modes gradually affects areas of low variance, then of high variance. The drop in performance remains stronger in low variance area with a 40% decrease of the average Q^2 in the low variance area versus a 10% decrease in the high variance area between the cases $L = 100$ and $L = 5$ (Table IV.1). Spatially, this results in a satisfactory POD reconstruction performance close to the obstacle but a severe degradation far from the ground.

Table IV.1: Explained variance ratio (in %) computed from training and test datasets for different number L of modes. The global Q^2 criterion represents the POD performance on all grid points. Local explained variance information can be obtained for lowest variance region $Q^2_{T_0}$, medium variance region $Q^2_{T_1}$ and largest variance region $Q^2_{T_2}$. Rigorous domain splitting was obtained from the 1/3 and 2/3 variance quantiles.

Dataset	Number L of modes	Q^2_{global}	$Q^2_{T_0}$	$Q^2_{T_1}$	$Q^2_{T_2}$
Train	100	99.7	98.5	99.4	99.7
Test	100	99.3	91.6	98.0	99.5
Test	23	97.0	87.0	93.7	97.3
Test	5	86.1	52.4	79.1	87.0

Snapshot reconstruction example. Figure IV.4 illustrates, through the example of the nominal snapshot from the LES test database, the capacity of reconstructing the mean tracer concentration fields from POD for different truncation levels L. Only accounting for the very first modes as suggested by the elbow rule ($L = 5$) provides a good representation of highly dispersed tracer areas downstream and of the tracer accumulation in the recirculation areas near the obstacle (Fig. IV.4d). However, it does not handle well sharp patterns resulting from tracer advection-dominated dispersion near the emission source. Moreover, the peak emission of the source is underestimated and not well located. The Kaiser rule (Fig. IV.4c) partially reconstructs the wake structures by including more modes ($L = 23$) but cannot correctly recover the plume structure close to the emission source. When including up to $L = 100$ modes in the reduced basis (Fig. IV.4b), improvements in the field reconstruction mainly relate to the intensity and location of the source peak emission and to the localised structures around the source.

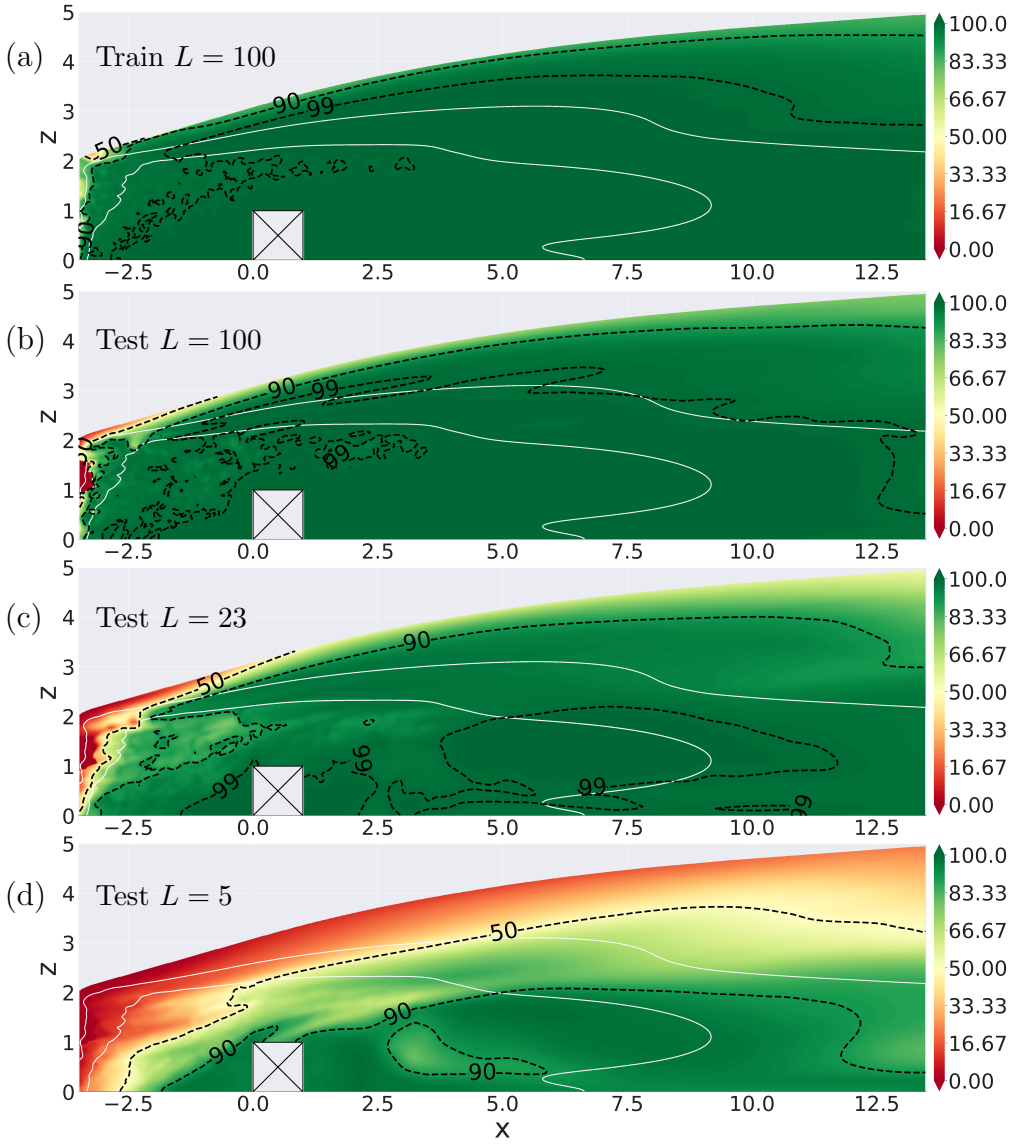

Figure IV.3: POD reconstruction error based on local Q^2 (%) computed at each grid point (Eq. IV.3) for the training and test datasets: (a) training dataset using $L = 100$ modes in the reduced basis; (b) test dataset using $L = 100$ modes; (c) test dataset using $L = 23$ modes; and (d) test dataset using $L = 5$ modes. Black dashed lines correspond to Q^2-contour lines; white thin lines correspond to the 1/3 and 2/3 variance quantiles.

IV.2.2 Spatial structures of the modes

We now analyse the variance structures carried by the POD modes from the correlation maps in the physical space (Eq. II.9) to better understand the need to have a large number L of modes. The correlation structures are associated with physical patterns of the mean LES fields. Figure IV.5 presents five out of the first hundred POD modes, which carry patterns that are representative of the whole set of POD modes.

– The first modes have spatially widespread structures with horizontally elongated shapes, looking like streaks aligned with the streamwise direction. The first POD mode (Fig. IV.5a) consists of two anti-correlated horizontal layers. Large tracer concentrations in the lower part of the domain are unlikely to occur along with high concentrations in altitude because of the flow horizontal structure. Indeed, the plume dispersion is more or less affected by the

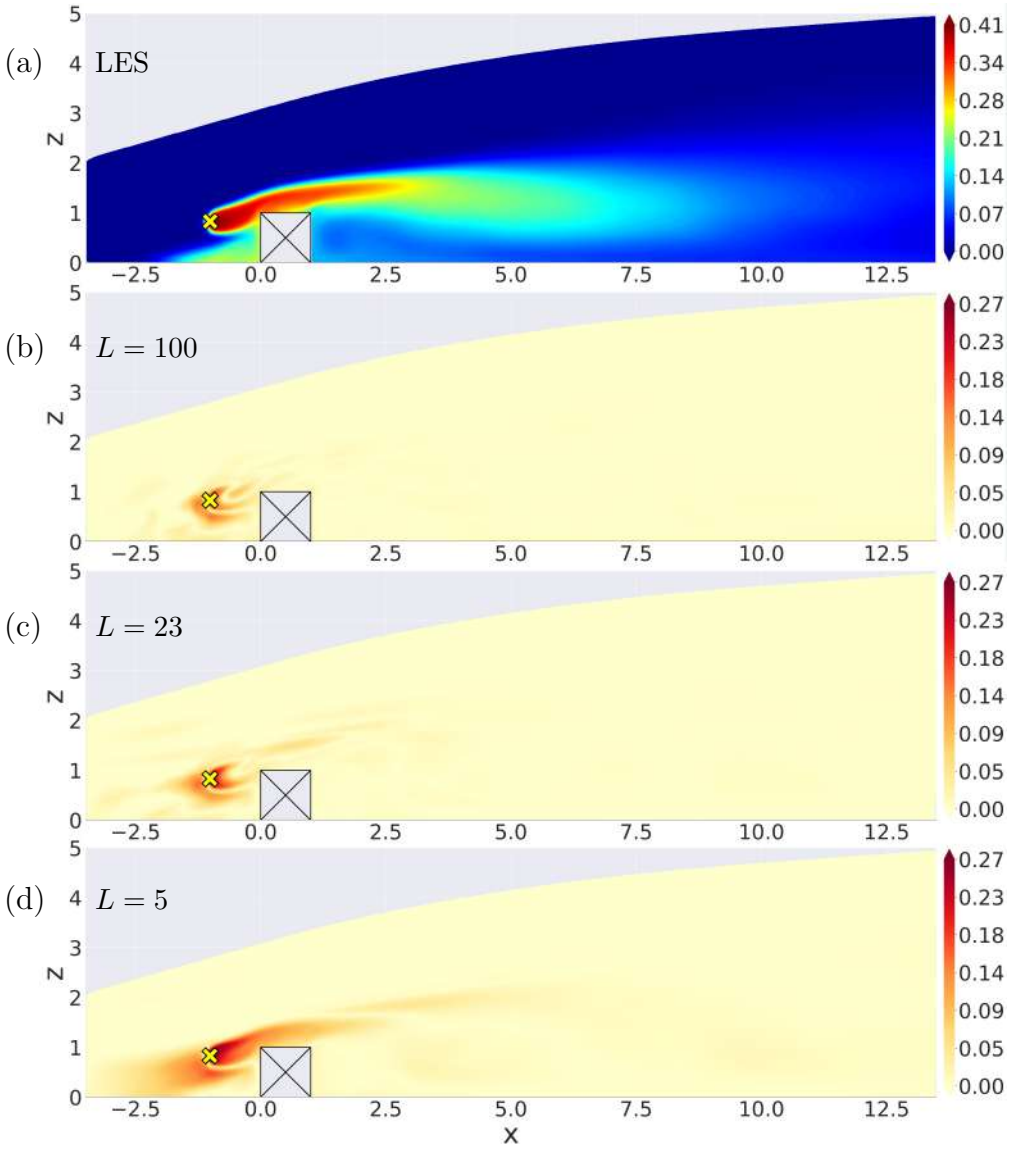

Figure IV.4: POD reconstruction for the nominal snapshot. (a) LES reference solution (mean normalised tracer concentration field). POD reconstruction absolute error using (b) $L = 100$ modes, (c) $L = 23$ modes (Kaiser rule), and (d) $L = 5$ modes (elbow rule).

obstacle depending on the emission source location. For the nominal snapshot (Fig. IV.5a), we observe that the plume dispersion is primarily controlled by the vortex shedding resulting from the flow interaction with the obstacle and the development of a recirculation region downstream of the obstacle. The corner eddy enhances the accumulation of tracer concentration upstream of the obstacle, while downstream tracer concentration is dispersed by turbulence. Overall, in the nominal case, high tracer concentration values remain localised near the ground. Differently, when the emission source is positioned higher (see example of Fig. IV.5c), the plume remains far from the ground and from the obstacle.

– The second POD mode (Fig. IV.5b) highlights the variety of concentration fields within the ensemble, and shows interactions between plume and recirculation areas, depending if the tracer source is upstream or downstream of the obstacle. When the source is located upstream and sufficiently close to the ground, the obstacle constrains the plume dispersion in an accumulation area close to its left boundary. Similarly, when the emission source is

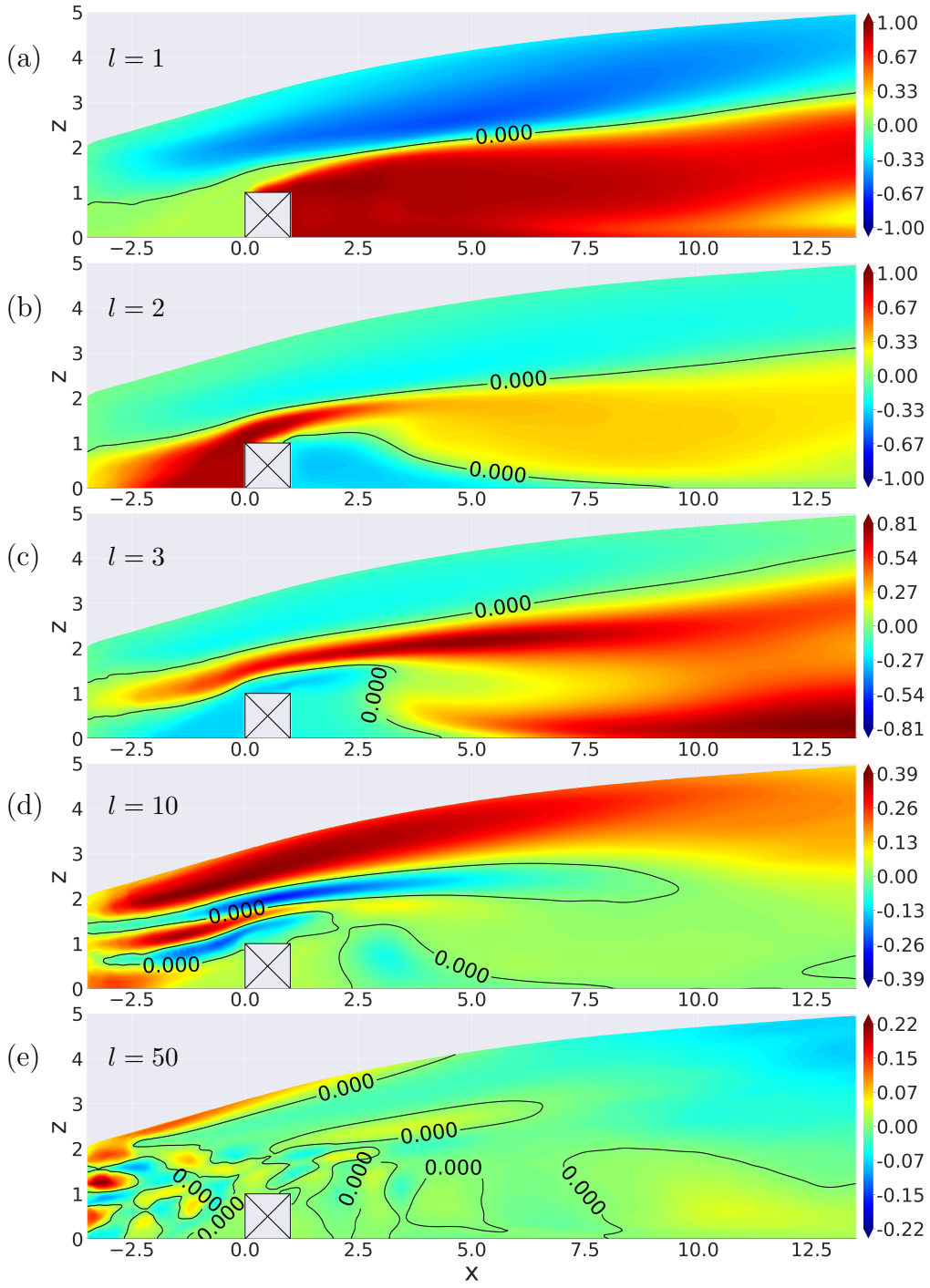

Figure IV.5: Correlation maps (between -1 and 1) between the lth POD mode $\boldsymbol{\psi}_l$ and the LES snapshots (Eq. II.9) sorted by increasing order of POD mode indices from $l = 1$ to $l = 50$ (i.e. by decreasing order of eigenvalues σ_l).

located downstream close enough to the obstacle (see example of Fig. III.8b), the tracer is trapped in a second recirculation area where large tracer concentration values can be obtained.

– The third mode (Fig. IV.5c) highlights the tracer emission position for which the plume stops being trapped in recirculation areas. This occurs when the emission source is located sufficiently far from the obstacle and from the ground. In that case, the plume is transported above the obstacle and disperses vertically further downstream because of vortex

shedding.

– From the tenth POD modes (Fig. IV.5d), we can see narrower correlation structures that are associated with the near-source advection-dominated dispersion of the tracer by the mean flow. The different streaks are directly related to the different source positions of the LES database since upstream emission source locations induce very refined, horizontally-elongated plume wakes.

– Finally, the example of the fiftieth mode (Fig. IV.5e) highlights that the very high POD modes focus on the remaining ensemble variance heterogeneity. This mode features very localised bubble structures, which match tracer concentration peaks due to emission source locations present in the POD training database. The small structure values also raise the question of the potential noise induced from the lack of training data and time-averaging convergence.

To conclude this section on POD, high-order modes contain fine spatial structures related to the near-source physics embedded in the LES snapshots. This large number of modes relates to the specificity of our problem, as changing the source location induces very sharp and localised plume structures near the source in the ensemble. Using POD, it is essential to keep a large number of modes to well represent local spatial concentration structures. We therefore choose at this stage, to keep $L = 100$ POD modes. Using this reduced basis, we investigate how to tune regression models to map POD coefficients of mean tracer concentration fields from any set of uncertain parameters.

IV.3 Comparison of regression models

In this section, we consider different regression models combined with POD and we assess their ability to represent how the POD reduced coefficients evolve with respect to the four uncertain parameters $\boldsymbol{\mu} = (u_{z_c}, z_0, x_{\text{src}}, z_{\text{src}})^T$. The choice of a regression model involves both the identification of a class (or family) of regression models and an appropriate choice for their hyperparameters. This study compares four classes: (a) Gaussian process regression, (b) gradient tree boosting, (c) polynomial chaos expansion, and (d) k-nearest-neighbours algorithm. Within each class, we search for the set of hyperparameters that maximises the regression model performance dependently from the POD mode index. It is indeed of high interest to study how the regression model classes behave in relation to the descending hierarchical scales of POD.

IV.3.1 Optimal hyperparameter search

The parameters of regression models can be classified into two groups: non-trainable hyperparameters and trainable weights (as discussed in Sect. II.3). As the hyperparameters are not optimised during the training process, we need to define a criterion to set their values. This is done using a grid search approach, which requires testing all predetermined hyperparameter settings and retaining the one (if any) that performs best on the validation dataset. It is worth

mentioning that for some hyperparameters, the problem background may help in setting appropriate values. In particular, some hyperparameter values may be intuited from POD mode behaviour and reduced-coefficient statistics.

IV.3.1.a Gaussian process regression

Gaussian process hyperparameters depend mainly on the choice of kernel (Sect. II.3.3). We explore here two kernel classes that differ in their smoothness assumptions: the radial basis function (RBF) kernel, and the Matérn kernel that is characterised by the smoothness hyperparameter ν. We test three different values of ν ($\frac{1}{2}, \frac{3}{2}$ and $\frac{5}{2}$). The anisotropic correlation length-scales, the noise variance and the Gaussian process variance are estimated here from maximum log-likelihood estimation (MLL).

Figure IV.6 shows the per-mode Q^2 performance (Eq. IV.2) for all four kernel configurations. The performance of Gaussian process regression seems to be robust regardless of the covariance kernel used. The performance on the first modes is close to the maximum (with a per-mode Q^2 close to 100%) and decreases with increasing mode index (with a per-mode Q^2 around 45% for the hundredth mode $l = 100$). For further work, we keep the Matérn 5/2-kernel since it is the most reasonable in terms of the differentiability assumption. Indeed, the RBF kernel induces an infinitely differentiable response surface, whereas Matérn 1/2 induces no differentiability at all, which is not physically reasonable. Between Matérn 3/2 and Matérn 5/2, we noted a greater stability of the Matérn 5/2 hyperparameters during the learning procedure, which explains our final choice.

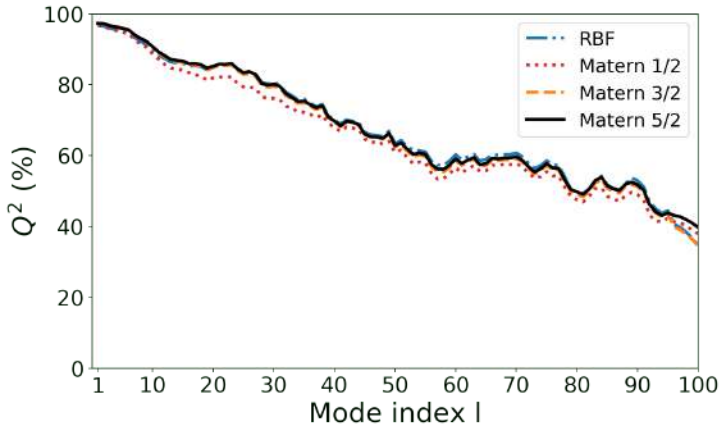

Figure IV.6: Comparison of Gaussian process regression model performance with respect to the mode index l (evaluated using the per-mode Q^2 metric in %, see Eq. IV.2) for different choices of covariance kernel (RBF versus Matérn kernel with different values of smoothness ν).

IV.3.1.b Gradient tree boosting

Gradient tree boosting combines both the hyperparameters of the decision trees and the hyperparameters of the boosting procedure (Sect. II.3.4). For consistency with explained POD explained variance maximisation, tree impurity and boosting loss are set to the MSE. To prevent overfitting, a minimum tolerance is set to 10^{-5} and 10^{-4} for tree impurity and boosting performance decay, respectively. These values are small relatively to the POD reduced-coefficient variance

equal to one due to whitening. It is worth noting that the tolerance on boosting performance decay constrains the number of trees in the boosting procedure: a minimum tolerance of 10^{-4} means that gradient boosting stops adding new trees if the performance does not decrease by at least 10^{-4} in the next twenty iterations. Similarly, the tolerance on tree impurity directly impacts the size of the trees. In order to build trees of reasonable size, the maximum depth D was set to five based on the condition $D \leq (d-1)$ with $d = 4$ the number of uncertain input parameters in this study. This ensures that all interactions are accounted for, resulting in a maximum number of $2^5 = 32$ terminal leaves.

We explore the change of gradient tree boosting performance when modifying for each reduced coefficient the learning rate ρ related to the boosting gradient descent convergence, and the cost-complexity pruning (CCP) coefficient α_{CCP}. Figure IV.7 shows how the per-mode Q^2 computed on the calibration dataset evolves with respect to the mode index l in the reduced basis when changing ρ or α_{CCP}. The choice of $\rho = 0.1$ for the learning rate maximises the performance of gradient tree boosting for each mode, even if there is a strong decrease in performance for higher-order modes. A very small value of the learning rate ($\rho = 10^{-3}$) results in poor performance on the first POD modes due to a lack of gradient descent convergence, while a very large value ($\rho = 0.5$) results in very poor performance on the high-order POD modes due to boosting instability. Moreover, gradient tree boosting performance seems almost insensitive to the choice of the cost-complexity pruning coefficient α_{CCP} in a reasonable range. Still, $\alpha_{\text{CCP}} = 0$ demonstrates a better score on the first POD modes, suggesting that pruning should be avoided for the first modes. Differently, a larger value $\alpha_{\text{CCP}} = 10^{-3}$ seems more appropriate on higher-order modes, indicating that small trees are becoming prevalent to represent the behaviour of high-order POD reduced coefficients. Too much pruning ($\alpha_{\text{CCP}} = 10^{-1}$) leads to a very poor performance over all modes, suggesting a poor generalisation capacity of the trees.

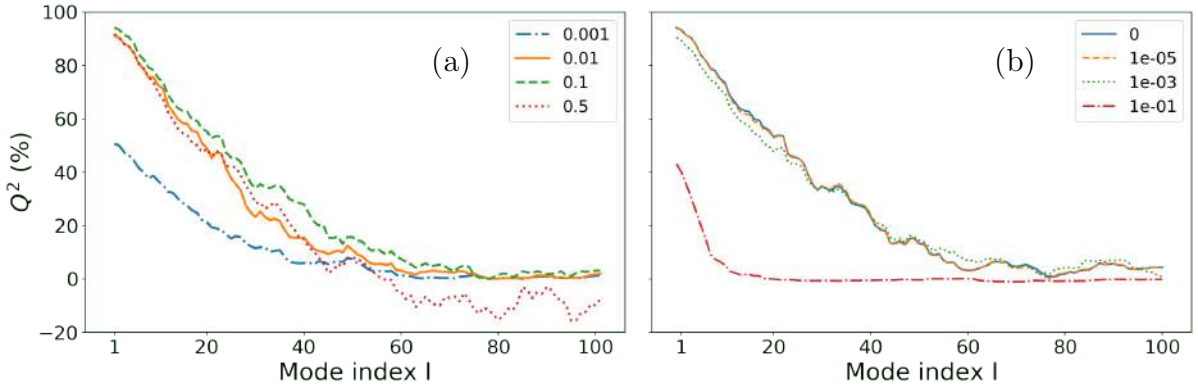

Figure IV.7: Gradient tree boosting per-mode Q^2 metric (in %) with respect to the mode index l for varying (a) learning rate ρ and (b) cost-complexity pruning coefficient α_{CCP}.

Finally, the complexity of the decision trees with respect to the mode index l is studied in Fig. IV.8. The number of trees is derived from the tolerance and the maximum number of iterations without performance improvement. It is found to be strongly influenced by the POD mode index l. The gradient boosting on the first modes can have deep trees (up to 350 trees); however, the number of trees significantly reduces with increasing mode index (less than 50 trees are considered for the hundredth reduced coefficient).

To ensure optimality of gradient tree boosting, additional tests on the hyperparameters (not shown here) were carried out. They indicated that stochastic gradient descent, robust impurity metrics (such as Friedman MSE [Friedman, 2001]) or varying minimum impurity decrease and tolerance thresholds do not improve the performance of gradient boosting for this problem.

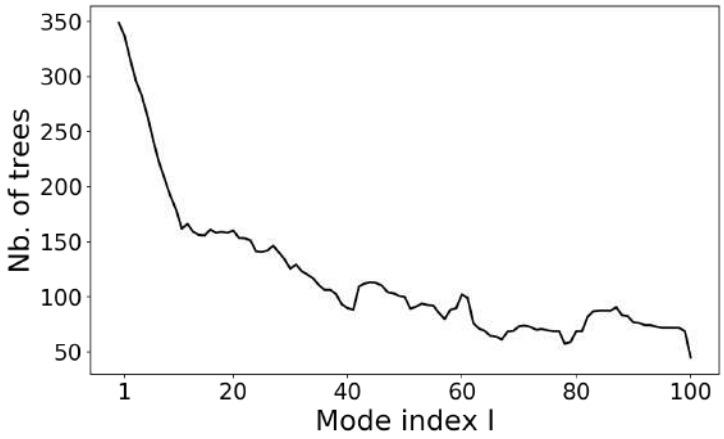

Figure IV.8: Number of trees per mode in the gradient tree boosting approach.

For regression model comparison in Sect. IV.3.2, we configure the gradient tree boosting approach with the learning rate $\rho = 0.1$ and a flexible value for the pruning coefficient α_{CCP} to have a better generalisation capacity. The automatic selection procedure for the number of trees shows that the high-order POD modes require fewer decision trees, which may be caused by increased difficulty to fit the noisy POD reduced coefficients.

IV.3.1.c Polynomial chaos expansion

Polynomial chaos expansion can be considered as a polynomial regression approach. Since the distributions on the input parameters are assumed to be uniform in this study (Sect. III.2), the polynomial basis is made of Legendre polynomials according to the Askey scheme (Sect. II.3.2). Then, the main question relates to the complexity of the polynomial combinations that should be included in the expansion to maximise performance. For this purpose, we study here how the polynomial chaos expansion performance varies according to the total polynomial order P (ranging from 1 to much more complex polynomial combinations up to a degree equal to 13). We also study if having a sparse polynomial basis can improve performance by *i*) adopting a hyperbolic truncation rule (using an hyperbolic coefficient q varying between 0.4 and $0.9 - q = 1$ corresponds to the full polynomial basis), and by *ii*) applying the cleaning strategy. In the latter case, we keep between 40 and 100 most significant coefficients and all polynomials associated with non-significant coefficients (the threshold varies between 10^{-7} and 10^{-3}) are removed from the polynomial basis.

Figure IV.9 shows how the per-mode Q^2 criterion computed on the calibration dataset evolves with respect to the mode index l for different values for the hyperparameters: the retained number of significant coefficients (Fig. IV.9a), the hyperbolic truncation strategy q (Fig. IV.9b), the significance threshold (Fig. IV.9c), and the total polynomial order (Fig. IV.9d). We first observe that the sensitivity to hyperparameters is stronger for polynomial chaos expansion than for pre-

viously tested regression models (i.e. Gaussian process regression and gradient tree boosting), especially for the first fifty modes. We also notice that on the first 10 modes, the choice of a complex model maximises performance, with maximised performance obtained when retaining 80 and 100 significant coefficients, a soft hyperbolic truncation $q = 0.9$ (meaning that many coefficients are retained in the expansion), a significance threshold equal to 10^{-7} and a high total polynomial order $P = 13$. This is possible since the first modes carry little noise. Conversely, complex polynomial models perform worse than simple models for high-order modes as data noise prevents the training of a high number of parameters. For instance, Fig. IV.9bd demonstrates a significant drop in Q^2 for a total polynomial order above 11 and a soft truncation coefficient $q = 0.9$ for the highest modes.

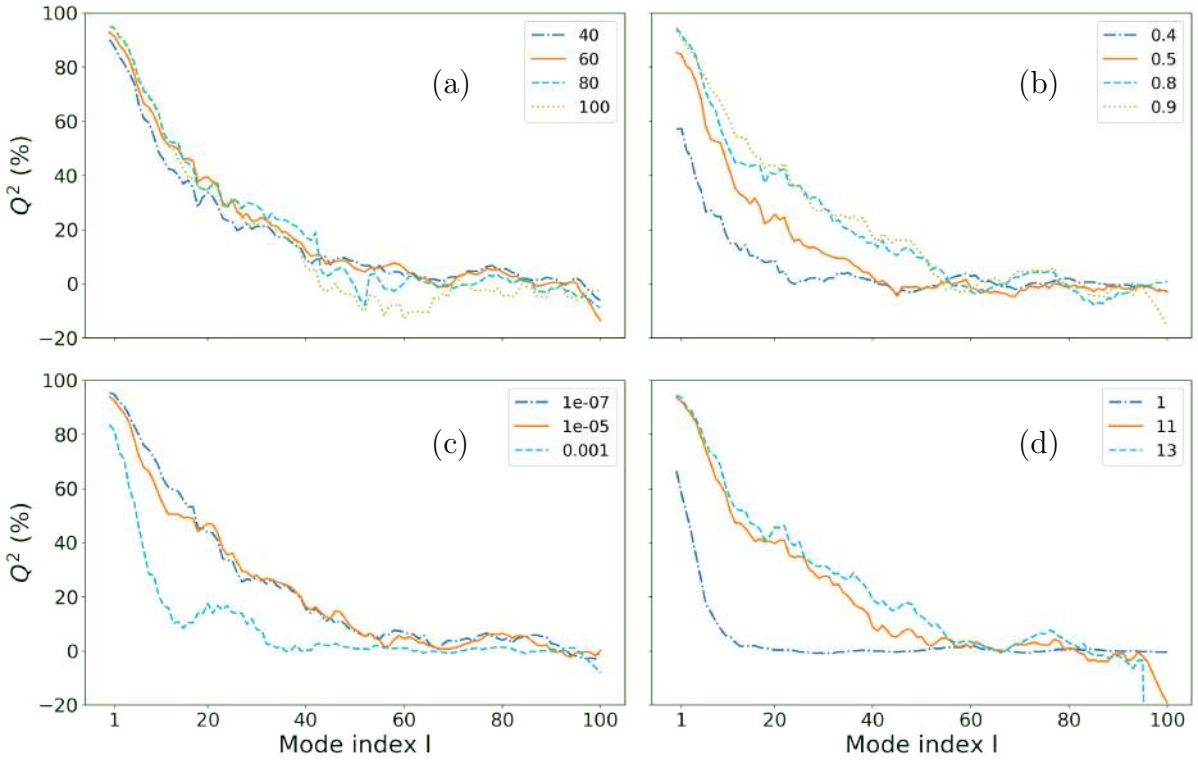

Figure IV.9: Polynomial chaos per-mode Q^2 metric (in %) with respect to the mode index l for varying (a) number of significant coefficients, (b) hyperbolic truncation value q, (c) significance threshold, and (d) total polynomial order P.

For regression model comparison in Sect. IV.3.2, we optimise the polynomial chaos hyperparameters based on the calibration performance. Thus, the first modes will be represented by a complex polynomial expansion (high total polynomial order, soft truncation coefficient and activation of the cleaning strategy), and the model's complexity will be gradually decreased for higher order modes to only retain the most significant coefficients and ensure robustness.

IV.3.1.d k-nearest-neighbours algorithm

The k-nearest-neighbours algorithm is tested as a baseline approach for the regression model comparison. Its principle is to represent each reduced coefficient as the weighted average of the values of the k-nearest neighbours in the parameter space. We study here how the performance of the algorithm evolves if we change the following hyperparameters: i) the number of neighbours

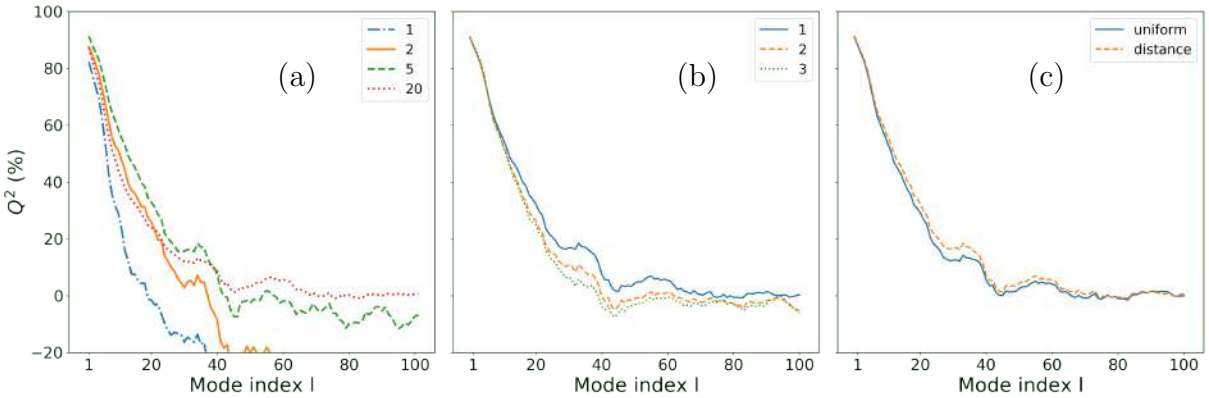

Figure IV.10: k-nearest-neighbours algorithm per-mode Q^2 performance (in %) with respect to the mode index l depending on the choice of (a) the number of neighbours, (b) the L^p-norm with p varying between 1 and 3, and (c) the weight function (uniform gives all neighbours equal weights; distance means that the weight is proportional to the inverse of the distance).

(ranging between 1 and 20), *ii*) the choice of the L^p-norm used to measure the distance between samples in the parameter space (L^1, L^2 and L^3-norms), and *iii*) the choice of the weight function (the weighted k-nearest neighbour average either depends on the distance in the parameter space, or is assigned a uniform weight independently of the distance).

We observe in Fig. IV.10 that the k-nearest-neighbours algorithm is sensitive to the number of neighbours but is not significantly impacted by the choice of the L^p-norm and of the weight function. The optimal number of neighbours changes with respect to the mode index: a low number of neighbours enhances performance in the low-order POD modes ($l \leq 40$), while a large number of neighbours improves the Q^2 score for high-order modes. This is consistent with the fact that the first reduced coefficients are noise free and the regression model can capture the local variations of the reduced coefficients by taking only five neighbors. However, on higher-order modes, POD reduced coefficients are embedded with noise and having a small number of neighbours makes the regression model prediction unstable. Increasing the number of neighbours to 20 limits the drop in performance and brings the response of the k-nearest-neighbours algorithm closer to the ensemble mean.

In the following comparison, we build the k-nearest-neighbours algorithm by adopting the L^1-norm, the distance weight function and an adaptive number of neighbours with respect to the mode index l.

In conclusion to this section on the regression model hyperparameters, this grid search approach allowed us to select the optimal hyperparameter settings within each class of regression models (Gaussian process regression, gradient tree boosting, polynomial chaos expansion, k-nearest-neighbours algorithm) using the calibration dataset. The next objective is to compare the different classes of regression models to identify the most relevant one to emulate the POD reduced coefficients and to accurately predict the mean tracer concentration fields.

IV.3.2 Performance comparison

To identify the most appropriate regression model for our problem, we consider several criteria: accuracy, efficiency (or computational cost), explicability, and robustness. In this section, we focus on the accuracy and efficiency criteria.

IV.3.2.a Accuracy

Mode-per-mode analysis of the error. As for the optimal hyperparameter search, we use the per-mode Q^2 metric to evaluate the accuracy of each regression model class (Fig. IV.11). Q^2-results show the quality of the Gaussian process regression model that achieves the best performance for all mode indices, with a net lead over the other regression models. In the opposite, the k-nearest-neighbours algorithm has the lowest Q^2-score over the first fifty modes, and it is then joined by polynomial chaos expansion. Both achieve a very poor performance for higher-order modes. This highlights the added value of more advanced regression models such as Gaussian process regression and gradient tree boosting for emulating the POD reduced coefficients.

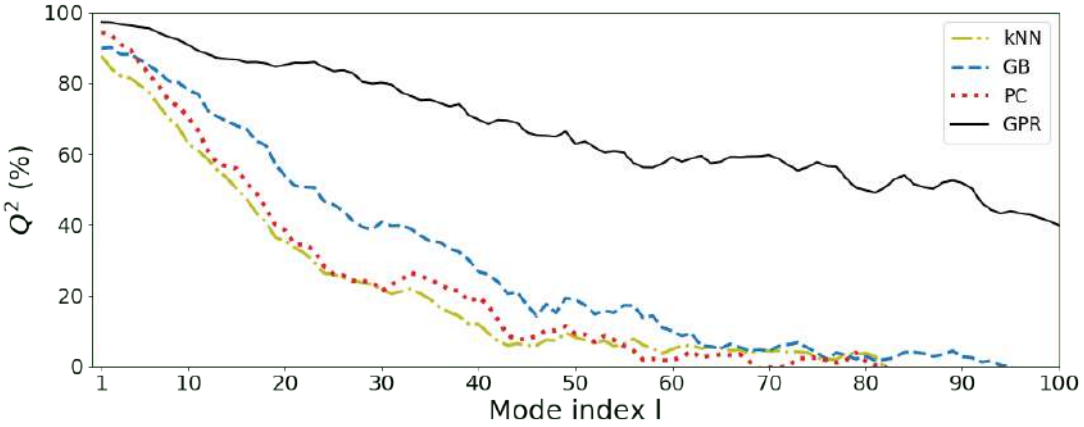

Figure IV.11: Comparison of per mode-Q^2 metric (in %) evaluated on the test dataset (Eq. IV.3) for four classes of regression models: k-nearest-neighbours algorithm (kNN) in yellow dashed-dotted line, gradient tree boosting (GB) in blue dashed line, polynomial chaos expansion (PC) in red dotted line, and Gaussian process regression (GPR) in black solid line.

Let us have a closer look at the performance evolution with respect to the mode index l. For the very first modes, the Q^2 score is close to 100%, especially for polynomial chaos expansion and Gaussian process regression. This means that the regression models perform almost perfectly on large variance structures with limited overfitting. However, all regression models cannot maintain this level of performance when moving to high-order modes, which feature more complex and localised structures and which are therefore more complex to predict. This is visible through the Q^2 performance decay rate with the mode index l. For Gaussian process regression, the decay rate remains small compared to other regression models (the Q^2-metric evolves between 60 and 40% from the fiftieth mode onwards). The Q^2 decay rate for polynomial chaos expansion is steeper than for gradient tree boosting: polynomial chaos expansion is better at predicting low-order POD reduced coefficients, but gradient tree boosting becomes better from the fifth mode onwards. This may be explained by the ability of the polynomial chaos expansion to reconstruct smooth response surface, while gradient boosting is known for its capacity to

deal with highly nonlinear response surface and noisy data. Still, polynomial chaos expansion and gradient tree boosting have a Q^2 metric below 40% from the thirtieth mode onwards, while Gaussian process regression never goes below this threshold.

Spatial analysis of the error. The analysis of the regression model performance can also be conducted in the physical space of the mean concentration fields by comparing emulated test snapshots with LES snapshots through the calculation of a spatial Q^2 field (Fig. IV.12). We can observe that the Q^2 is not homogeneous throughout the domain. Downstream of the obstacle the area near the ground is well predicted ($Q^2 > 90\%$) by all regression model classes. Well predicted areas are strongly correlated with high ensemble variance areas that are carried by the first POD modes; these modes are globally well emulated by the four regression models with a Q^2 near 100% (Fig. IV.11). The downstream area far above the ground the performance is much lower (Q^2 varies between 50 and 90%), but it is upstream of the obstacle, in the area where the emission sources are located, that we see a very severe drop in the performance of the regression models (with a Q^2 value well below 50%). In this area, information is carried by higher-order modes (Fig. IV.5), which are more challenging to predict (Fig. IV.11). Gaussian process regression is the best approach to represent the response of these high modes to parametric variations with a per mode-Q^2 value that remains over 40% and a much smaller area of very low Q^2 value (red area in Fig. IV.12a).

Table IV.2: Global and variance sorted Q^2-scores (in %) computed from the test dataset for $L = 100$ modes. The global Q^2 criterion represents the reduced-order model performance on all grid points, and local explained variance information can be obtained for lowest variance region $Q^2_{T_0}$, medium variance region $Q^2_{T_1}$ and largest variance region $Q^2_{T_2}$ as in Table IV.1.

Class	Q^2_{global} (%)	Q^2_{T0} (%)	Q^2_{T1} (%)	Q^2_{T2}(%)
POD	99.3	91.6	97.9	99.5
Gaussian process regression	96.7	81.8	94.0	97.0
Gradient tree boosting	91.1	81.5	86.0	91.5
Polynomial chaos	90.5	73.6	86.2	90.9
k-nearest neighbours	87.7	74.7	83.8	88.1

This discussion can be summarised by Table IV.2. The best global performance is obtained through Gaussian process regression ($Q^2 = 96.7\%$). This approach also achieves the best performance on each subregion of the domain consistently with the results on the modes in Fig. IV.11. Gradient tree boosting and polynomial chaos expansion achieve almost the same global performance above 90%. They obtain a very similar performance in the high-variance region (only 0.5% of $Q^2_{T_2}$-difference), while the difference is quite significant on the low-variance region (7.9% of $Q^2_{T_0}$-difference). This can be explained by the fact that the global Q^2-score favours high variance grid points by assigning a weight proportional to the overall variance (Eq. IV.4). Note that the k-nearest-neighbours algorithm achieves the lowest performance, except on the low variance region ($Q^2_{T_0}$), where it is ahead of polynomial chaos expansion by 1.1%. This may be explained by a poor estimation of the Q^2 in this area: when the variance tends towards 0, the Q^2-estimator lacks reliability because of its ratio analytical expression (Eq. IV.3).

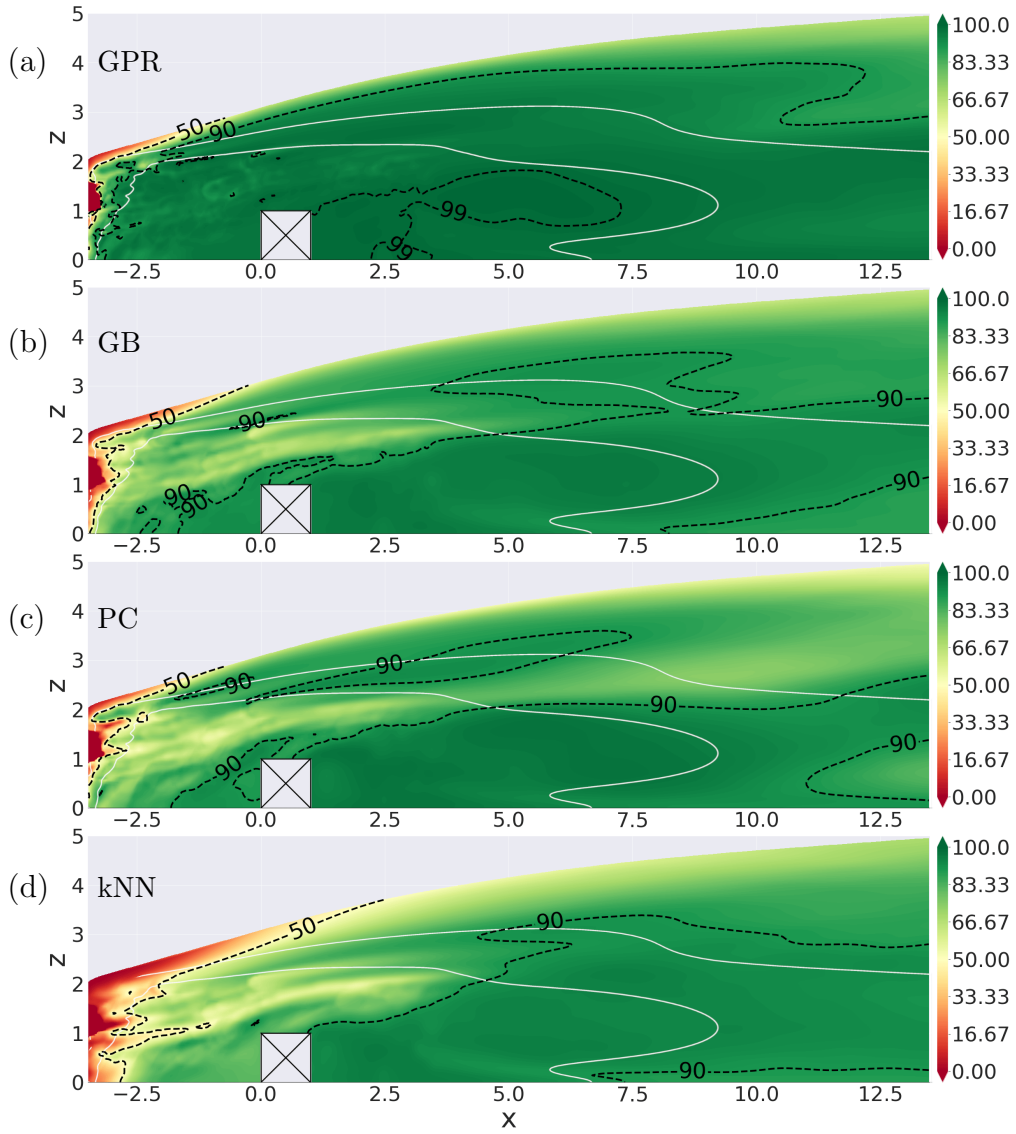

Figure IV.12: Regression model prediction error based on local Q^2-score (in %) computed at each grid point for the test dataset using $L = 100$ POD modes. (a) Gaussian process regression (GPR). (b) Gradient tree boosting (GB). (c) Polynomial chaos expansion (PC). (d) k-nearest-neighbours algorithm (kNN). Black dashed lines correspond to 50%, 95% and 99% Q^2-contour lines. White thin lines split the spatial regions with respect to the $\frac{1}{3}$ and $\frac{2}{3}$ variance quantiles.

Field prediction examples. Previous results show that Gaussian process regression is better suited to emulate mean tracer concentration fields over a wider area of the domain, including the low-variance region. As an illustration, Fig. IV.13 to Fig. IV.15 show the mean normalised tracer concentration fields predicted by the Gaussian process regression model for three snapshots of the LES test dataset.

– For the nominal snapshot (Fig. IV.13), the largest prediction errors are made *i)* close to the emission source with largely underestimated tracer concentration (about 50% of the LES reference concentration), and *ii)* in the accumulation area upstream of the obstacle with overestimated coarser-structured tracer concentration levels. Further away from the source, the tracer concentration in the wake of the obstacle is well predicted. Upstream of the obstacle, we can distinguish slight noise in no-tracer areas due to high-order POD modes that carry small noisy structures (Fig. IV.5).

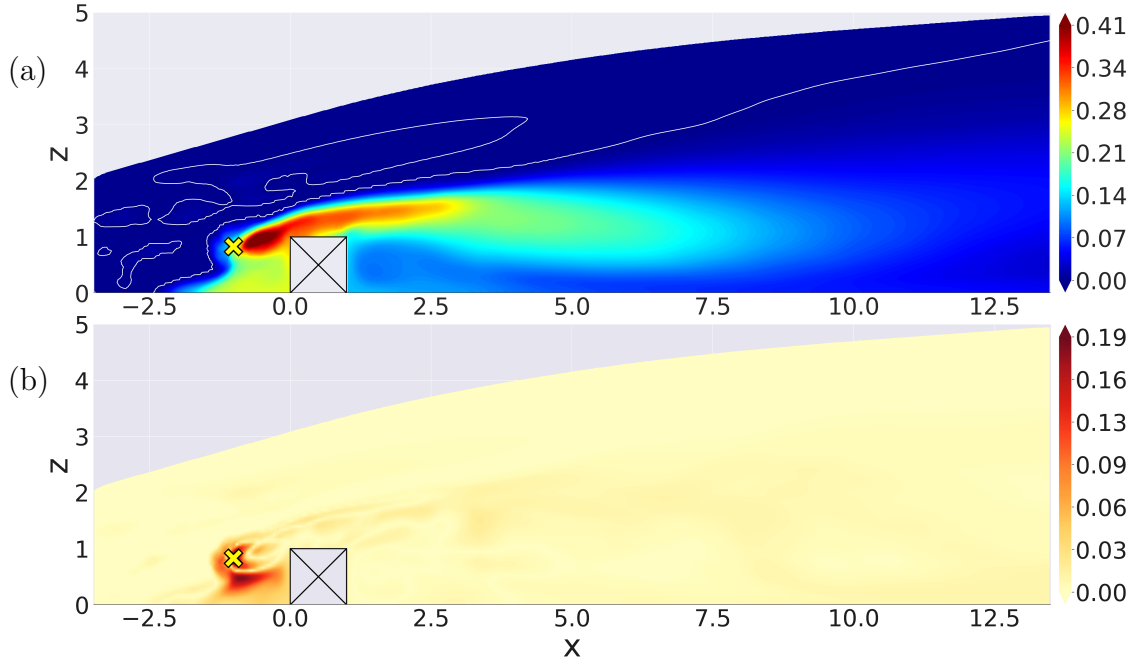

Figure IV.13: Nominal snapshot mean normalised tracer concentration field obtained with (a) Gaussian process prediction. (b) Prediction absolute error calculated with respect to LES snapshot (Fig. III.4). Contour line of the mean normalised tracer concentration equal to 5×10^{-4} is superimposed on predicted field to highlight the presence of low-magnitude noisy structures.

– We also present the prediction result for a case where the emission source is far from the obstacle and the ground (Fig. IV.14). This case emphasises what has already been observed in the nominal case. The tracer concentration at the actual emission source and along the wake is hard to predict. The prediction absolute error can reach up to 75% of the LES solution. The fine structures of the plume are hard to predict since information is mostly carried by multiple high-order POD modes in a region where there are only a few samples of the emission source in the LES dataset (this case is located at the boundary of the tracer source area – Fig. III.1).

– We finally present the prediction result for a case where the emission source is located in the recirculation area downstream of the obstacle (Fig. IV.15). This case is much better predicted by the regression model. The areas of high tracer concentrations form a wide-spread structure in areas carried by the first POD modes. Most of the information can be therefore conveniently recovered from the first reduced coefficients with accurate regression models, including in the recirculation area downstream of the obstacle. The largest prediction error is on the order of 10% of the LES solution.

These three snapshots highlight the capacity of the Gaussian process regression model to predict the main tracer concentration structures, in particular when the emission source is located in recirculation zones near the obstacle. In this situation, the tracer concentration patterns are smooth as the dispersion is dominated by turbulent diffusion, making the prediction task easier. The main challenge in this work is therefore to reconstruct the mean tracer concentration in the upstream region near the emission sources.

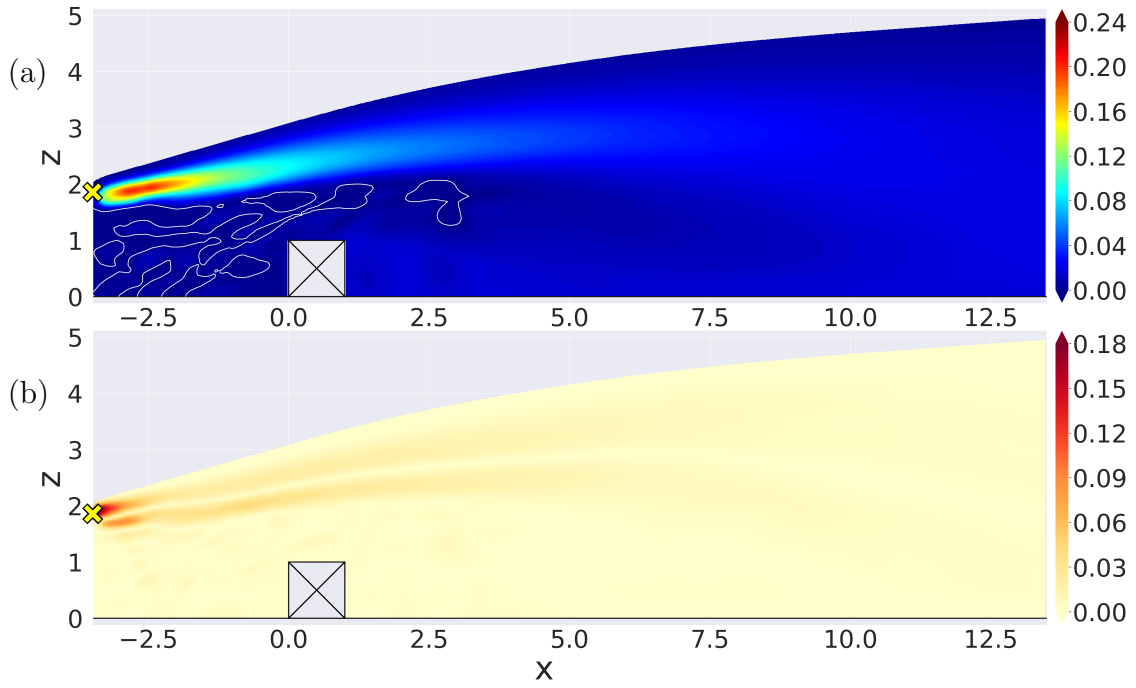

Figure IV.14: Same caption as in Fig. IV.13 for a LES snapshot with a high emission source (Fig. III.8c).

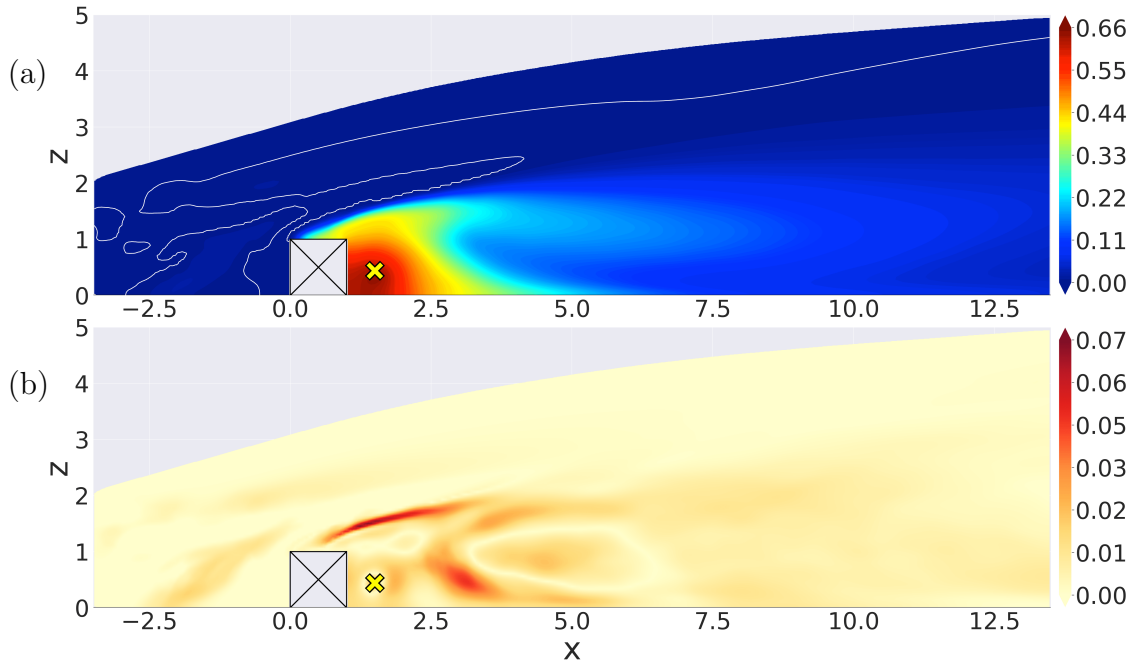

Figure IV.15: Same caption as in Fig. IV.13 for a LES snapshot with an emission source downstream of the obstacle (Fig. III.8b).

IV.3.2.b Efficiency

Although Gaussian process regression achieves the best performance for emulating the POD reduced coefficients, its training stage is very CPU intensive. Table IV.3 summarises the CPU cost for both training and prediction steps of POD and all four regression models. Even though the k-nearest-neighbour algorithm achieves the poorest Q^2-performance, the training and prediction of tracer concentration fields is very fast (there is no additional cost compared to a standalone inverse POD reconstruction). Training and prediction by the other regression models are between one and three orders of magnitude higher than the k-nearest-neighbours algorithm. The high cost of polynomial chaos training is due to the cleaning procedure, while Gaussian process regression suffers from the cost due to hyperparameter optimisation (Sect. II.3.3.c). If uniform prior distributions are assumed over the hyperparameters, gradient descent has difficulty to converge due to multiple local optima. To overcome this issue, one way is to perform multiple gradient descent iterations starting from different hyperparameter initial conditions. The final solution is then chosen as the one achieving the highest maximal log-likelihood (MLL) score. These multiple gradient descent iterations increase the computational cost of Gaussian process regression.

Table IV.3: CPU cost (in s) for the training and prediction steps for standalone POD and reduced-order models (POD combined with regression models). In the column "training", the CPU cost aggregates the score for the 100 subtraining tasks (i.e. the training task associated with each mode index between 1 and 100) that are done for a training dataset made of 450 snapshots. In the column "prediction", the score represents the CPU time for the spatial prediction of a single snapshot including both emulation and inverse POD reconstruction.

Family	Training (sCPU)	Prediction (sCPU)
POD	10^{-1}	10^{-4}
Gaussian process regression	10^{3}	10^{-3}
Gradient tree boosting	10^{1}	10^{-1}
Polynomial chaos expansion	10^{3}	10^{-3}
k-nearest neighbours	10^{0}	10^{-4}

To conclude this section on the regression model comparison, Gaussian process regression obtains the best prediction performance with a clear lead over the other regression approaches, in particular for the high-order modes that are necessary to include in the reduced basis to well represent the tracer concentration in the wake of the emission sources. However, the MLL optimisation approach for the hyperparameters is costly. Due to its great accuracy, it is of interest to find ways to make the Gaussian process optimisation step more efficient.

IV.4 Improving Gaussian process regression efficiency

Given the superior performance of Gaussian process regression over alternative metamodels, it is worthwhile to enhance the learning of Gaussian process hyperparameters in order to reduce the training computational cost. One way to accelerate the optimisation process is to provide an appropriate starting point for the hyperarameters to avoid performing multiple gradient de-

scent iterations as in the MLL procedure. For this purpose, the optimisation procedure can be informed by providing prior distributions for the hyperparameters that are not uniform. We propose in this work to infer these distributions from the POD reduced coefficients using the validation dataset. This approach is referred to as the maximum a posteriori (MAP) estimation (Sect. II.3.3.c) and is compared to the standard MLL approach in terms of accuracy and efficiency.

IV.4.1 Hyperparameter prior distribution

Gaussian process regression requires optimising hyperparameters $\boldsymbol{\theta}_l = \{s_l^2, \varrho, \lambda_{u_{z_c}}, \lambda_{z_0}, \lambda_{x_{\text{src}}}, \lambda_{z_{\text{src}}}\}$ for the lth reduced-order model. We present here how to calibrate Gamma and Gaussian distributions for the hyperparameters from POD modes and reduced-coefficient features that are required as input to the MAP optimisation procedure.

Noise prior. POD often assumes that the variability of the low-order reduced coefficients is related to systematic behaviour among the LES dataset, whereas the variability carried by the high-order reduced coefficients matches data noise [Jolliffe, 2002]. This is equivalent to treating the reduced coefficients on the first modes as unbiased data. In this study, Gaussian process regression accounts for the noise on the lth reduced coefficient through the hyperparameter s_l^2 (Eq. II.21). In this work, we assume that a prior estimate of s_l^2 may be obtained from the noise introduced by the time-averaging process performed on the LES. Note that this is not the only source of noise introduced during the generation of the LES database, but this provides a lower bound noise estimation that is useful to demonstrate the added value of our methodology.

The main parameter involved in the time-averaging process is the length of the time-averaging window used to acquire LES statistics. To evaluate the noise for the lth reduced-order model, we compare the reduced coefficient of converged simulations (k_l) with the non-converged reduced coefficient obtained by averaging over only 50% of the full simulation time window ($k_{l,50\%}$) as:

$$\hat{s_l}^2 \approx \frac{1}{2N} \sum_{n=1}^{N} \left(k_l(\boldsymbol{\mu}^{(n)}) - k_{l,50\%}(\boldsymbol{\mu}^{(n)}) \right)^2 . \tag{IV.6}$$

Figure IV.16 shows the $\hat{s_l}^2$-estimates obtained on the validation samples. The estimated noise increases from 10^{-4} to 10^{-2}. It is small compared to the variability of the reduced coefficients (normalised to unity). Estimates of s_l^2 vary over several orders of magnitude. Since s_l^2 is positive, we set a Gamma prior distribution for s_l^2. The mode of the Gamma prior is set from the smoothed estimation for each of the L reduced-order models (thick line in Fig. IV.16). Since reduced coefficients are normalised to unit-variance, we have $0 \leq s_l^2 \leq 1$. The mean of the Gamma distribution is set to 0.5 (middle value of the interval). For the hyperparameter optimisation step, the starting point of the gradient descent for s_l^2 is taken as the Gamma distribution mode value.

Mean and scaling priors. Now we establish prior information on the mean and variance of the Gaussian processes (Eq. II.36). In our Gaussian process regression formalism, the mean is

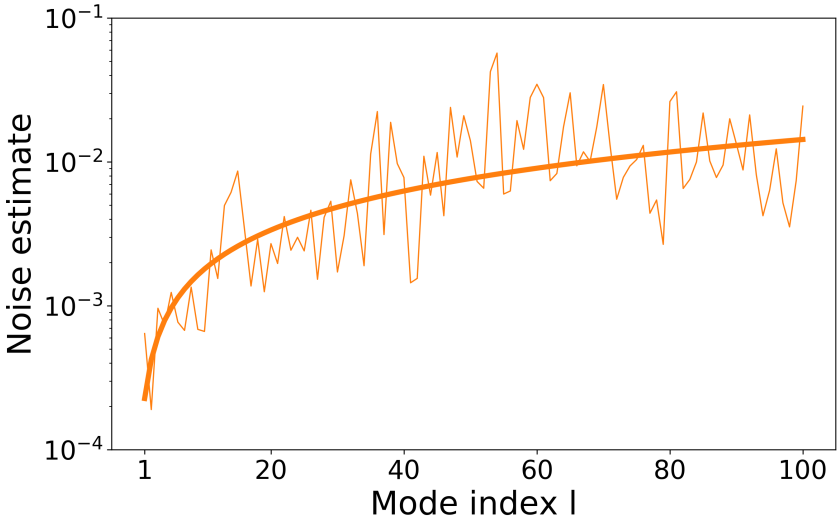

Figure IV.16: Estimation of the noise hyperparameter s_l^2 for each reduced-order model l. The thin line corresponds to the noise estimation $\hat{s_l}^2$ from the POD modes (Eq. IV.6). The thick line corresponds to the average trend found by regression and defined by the following equation $\hat{s}_l^2 = 2.16 \times 10^{-4}\, l^{0.93}$.

assumed to be constant, and the variance is decomposed into a systematic component and a noise component. From the expression of the noisy regression models, we obtain for each POD reduced coefficient l:

$$\mathbb{E}[k_l] = m_l\,, \quad \mathrm{Var}(k_l) = \varrho + s_l^2, \tag{IV.7}$$

where the signal variance hyperparameter ϱ (Eq. II.41) characterises systematic variability, while s_l^2 characterises noise variability.

Figure IV.17 shows the mean and variance statistics of the reduced coefficients on the different LES datasets (Sect. IV.1.2). The variance evaluated on the training data is equal to one by construction. This is no longer the case when the variance is evaluated on the calibration and test data. Still, the variance statistics oscillate on average around one with a Gaussian-like distribution. Similar behaviour can be observed for the mean statistics that are equal to zero on the training data and on average around zero on the calibration and test data.

In this study, to be consistent with these results, the Gaussian process mean is set as deterministic with $m_l = 0$ for all $l = 1, \cdots, 100$. Concerning the variance, this is represented using a scaled Matérn kernel. When compared to the total variability of the Gaussian process, the proportion of variability attributed to noise (associated with s_l^2) appears to be small. For this reason, we assume that the total variability is essentially induced by systematic behaviour of LES data and carried by the hyperparameter ϱ. We set a Gaussian prior distribution for ϱ with mean and variance estimated from the validation data ensemble statistics and set to 1 and 0.03, respectively. During the optimisation step, the starting point of the gradient descent for ϱ is set to the Gaussian distribution mean.

Correlation length-scale prior. We are now interested in the prior distributions of the Matérn kernel correlation length-scales (as described in Sect. II.3.3.b) $\{\lambda_{x_{\mathrm{src}}}, \lambda_{z_{\mathrm{src}}}, \lambda_{U_{z_c}}, \lambda_{z_0}\}$. The analysis of the POD modes (Fig. IV.5) reveals an increase in small-scale spatial heterogeneity for increasingly higher-order modes, which results from the ensemble variability driven by the emis-

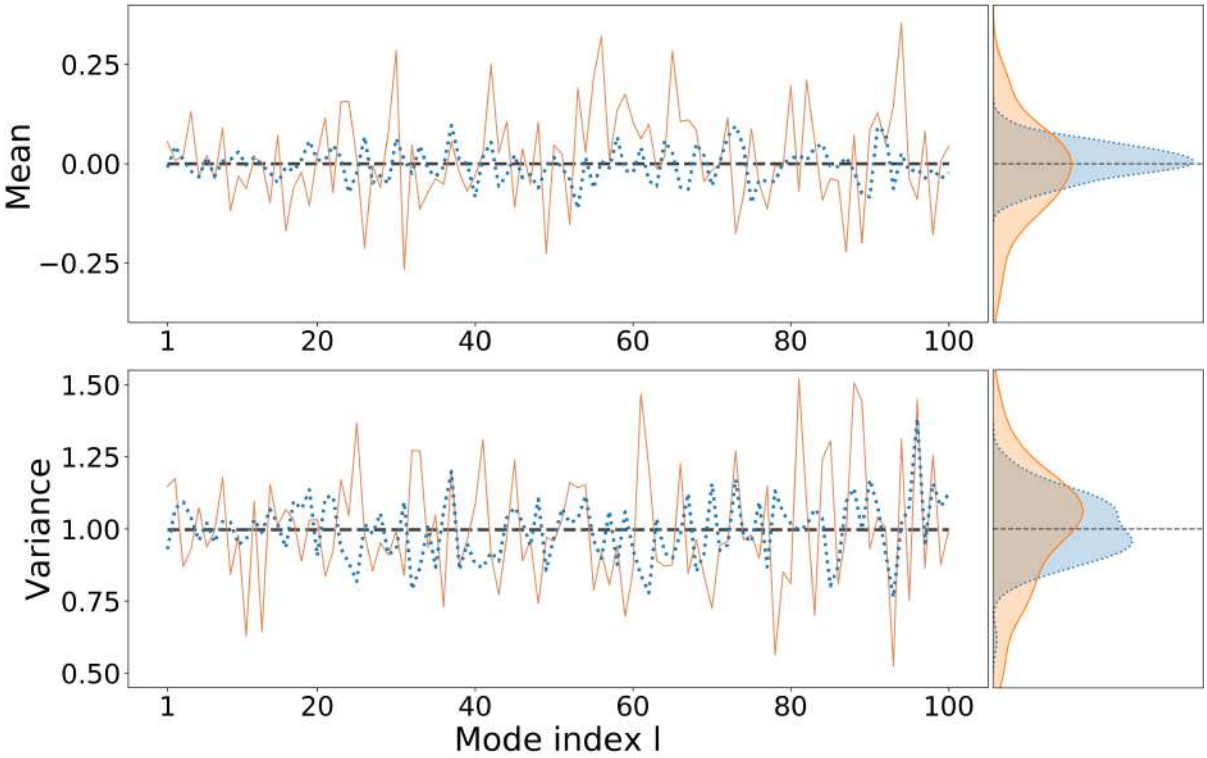

Figure IV.17: Ensemble variance of reduced coefficients $\{k_l\}_{l=1,\dots,L}$ evaluated on the three subsets of the LES data: training data (dashed line), calibration data (solid line), and test data (dotted line). Related statistical distributions are plotted on the right.

sion source location.

The first modes have spatially widespread structures with horizontally elongated shapes, looking like streaks aligned with the streamwise direction. When the POD mode index increases, the number of alternated streaks, corresponding to shorter wavelengths in the vertical direction. Due to the alternated signs of the POD modes, it seems natural to anticipate the correlation lengths of the random processes used to model the POD coefficients as being strongly influenced by these patterns. In particular, we can see that the typical correlation length along the source height should be related to the number of streaks, and should therefore decrease as we consider higher POD modes.

Based on these considerations, we model a decrease in the length-scales $\lambda_{x_{\text{src}}}$ and $\lambda_{z_{\text{src}}}$ associated with x_{src} and z_{src} when moving to higher-order modes. For low-order POD modes, the length-scales tend to be large (i.e. the process is stable relatively to source position and height). For high-order POD modes, they tend to be smaller (i.e. the process is unstable relatively to source position and height). Because the correlation length-scales are positive, we adopt a prior Gamma distribution for $\lambda_{x_{\text{src}}}$ and $\lambda_{z_{\text{src}}}$. The mode of each Gamma distribution is determined by the simple decreasing rule: $1/l$ for $l = 1, \dots, 100$. The variance of the Gamma distribution is set to 1, which corresponds to the interval length associated with the normalised input parameters ($[0,1]^4$). Since there is no analogous interpretation of decreasing length-scales for z_0 and U_{z_c}, we retain prior Gamma distributions with constant mode and variance equal to 1 for $\lambda_{U_{z_c}}$ and λ_{z_0}. During the hyperparameters optimisation step, the starting point of the gradient descent for $\{\lambda_{x_{\text{src}}}, \lambda_{z_{\text{src}}}, \lambda_{U_{z_c}}, \lambda_{z_0}\}$ is set to the Gamma distribution mode.

It is therefore possible to define prior distribution for the Gaussian process hyperparameters based on POD information. This information is then used as a starting point of the gradient descent in the MAP optimisation step of Gaussian process regression.

IV.4.2 Comparison of Gaussian process regression optimisation procedures

We now compare the reduced-order model solutions obtained from MLL and MAP hyperparameter optimisation procedures to show the added value of MAP. We also quantify the noise of the coefficients on each POD mode.

Comparison of optimisation solutions. Figure IV.18 presents the modes of the hyperparameter prior distribution (dashed lines) and the optimised hyperparameter solutions obtained with MAP (dotted lines). The MLL solution (solid lines) is also plotted for comparison with MAP. Figure IV.18abcd shows that MAP and MLL procedures converge to similar solutions. Estimated correlation length-scales are close to each other. However, MAP estimates are systematically underpredicted compared to MLL solutions on the first fifty modes.

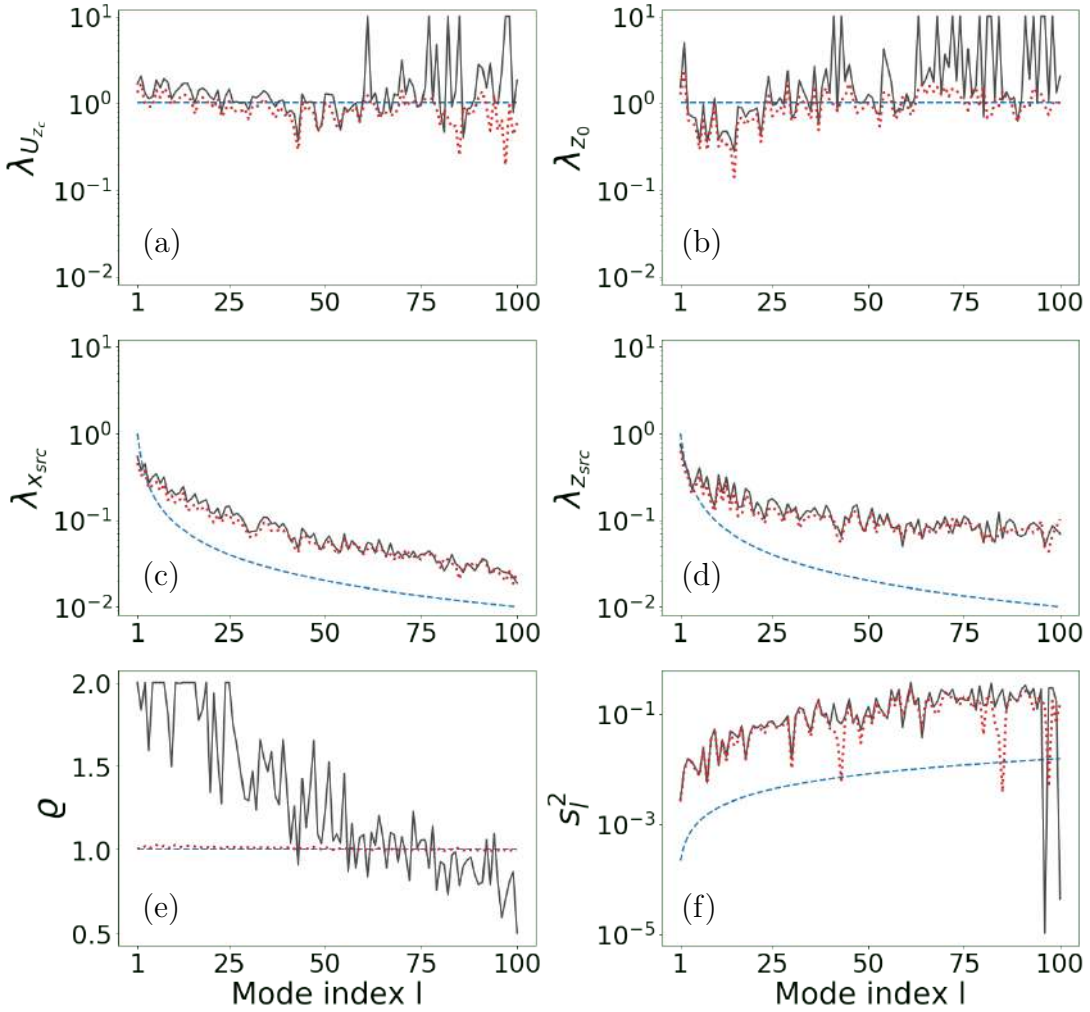

Figure IV.18: Gaussian process correlation length-scales for (a) u_{z_c}, (b) z_0, (c) x_{src}, and (d) z_{src}. Correlation length-scales can be optimised using MLL (solid lines) or MAP (dotted lines) starting from prior modes (dashed lines). Noise in the MLL and MAP estimates (e) is used to determine s_l^2 (f).

Noise estimates in Fig. IV.18ef are consistent for almost all POD modes. Only the estimates

for the variance hyperparameter ϱ strongly differ between MLL and MAP procedures. To explain this difference, we note that the prior distribution associated with ϱ features a low variance. Therefore, the MAP procedure converges to values close to the prior distribution mode. The MLL procedure is not constrained by the prior distribution, and the ϱ-estimates diverge on the first modes towards the upper bound value (set at 2). Additional tests (not shown) demonstrated that the MLL procedure applied with $\varrho = 1$ leads to correlation length-scales identical to MAP estimates. This indicates that the ϱ-estimates obtained with MLL and MAP impact the associated length-scales. For MLL, the excess of variance on the low-order modes (associated with a larger value of ϱ) is balanced by a greater stability with respect to the input parameters (associated with larger correlation length-scales). From the fiftieth mode onwards, the MLL ϱ-estimates decrease around one, and the MLL correlation length-scales are no longer systematically greater than the MAP correlation length-scales. The decrease in ϱ is explained by the fact that the variance on the reduced coefficients remains close to 1 for each POD mode (due to whitening) and at the same time s_l^2 increases (Fig. IV.18).

In the end, the major difference between the MLL and MAP results concerns ϱ on the first POD modes and consequent length-scales. Nevertheless, since noise is negligible on the first modes (because $\varrho \gg s_l^2$), the associated Gaussian process mean predictions are almost equal (Eq. II.38). Hence, MLL and MAP procedures converge towards similar optima for the Gaussian process regression hyperparameters, meaning that they correspond to equivalent predictive models but with an economy concerning numerical costs for MAP compared to MLL: the coarse prior distributions are sufficient to ensure MAP convergence in a single gradient descent when fifteen iterations are required for MLL (this factor of fifteen is all the more important as this optimisation procedure is repeated for each of the $L = 100$ regression models).

Gaussian process noise analysis. Figure IV.18f also shows that the estimated noise (s_l^2) is two orders of magnitude above the prior solution. This suggests that the temporal convergence error may not be the primary source of noise in the data. On the first POD modes, the noise is small compared to the signal variance. This is consistent with the fact that the first POD modes are almost noise free since they carry large variance structures. But the noise increases continuously on high-order modes such that it can be substantial compared to the signal variance (the maximum noise estimate is close to 0.3 for very high modes, which is of the same order of magnitude as the signal variance that is around 1). This gradual increase of noise indicates that there are not two distinct behaviours among the modes, namely those that carry systematic information on the one hand and those that carry noise on the other hand. It is worth noting that the ratio between s_l^2 and ϱ could be a way to choose the number of POD modes L to retain in the reduced basis.

Correlation length-scale analysis. We now identify the most relevant input parameters in the Gaussian process regression models. The order of magnitude for $\lambda_{U_{z_c}}$ and λ_{z_0} is equivalent to $\lambda_{x_{\text{src}}}$ and $\lambda_{z_{\text{src}}}$ on the first POD modes (Fig. IV.18abcd). This means that the variance on the reduced coefficients is equally due to the variations on all input parameters. On high POD modes, the values of $\lambda_{U_{z_c}}$ and λ_{z_0} are larger than $\lambda_{x_{\text{src}}}$ and $\lambda_{z_{\text{src}}}$. This implies that the reduced

coefficients are relatively insensitive to variations in U_{z_c} and z_0, and are mainly explained by x_{src} and z_{src}. This is why the optimisation process for $\lambda_{U_{z_c}}$ and λ_{z_0} becomes unstable for high-order POD modes. The correlation length-scales on position parameters x_{src} and z_{src} decrease in a stable manner over the first hundred POD modes as in the prior solutions. This suggests that the value of the reduced coefficients becomes more sensitive to small variations in source position and height on the higher POD modes. This is consistent with the small structures observed in the high-order modes and associated with tracer concentration wakes (Sect. IV.2). A large number of POD modes ($L = 100$) is necessary here to characterise the high sensitivity of the tracer concentration upstream of the obstacle to the source height and position.

Gaussian process regression accuracy and efficiency. We now evaluate the accuracy of the Gaussian process regression models over the test dataset for validation. Figure IV.19 shows that the models obtained from MLL and MAP procedures are equivalent as anticipated from the equivalent correlation length-scales in Fig. IV.18. Optimising the hyperparameters mode per mode greatly improves the Gaussian process accuracy compared to simply imposing prior trend on hyperparameters. This gain in Q^2 is more pronounced when moving to high-order modes.

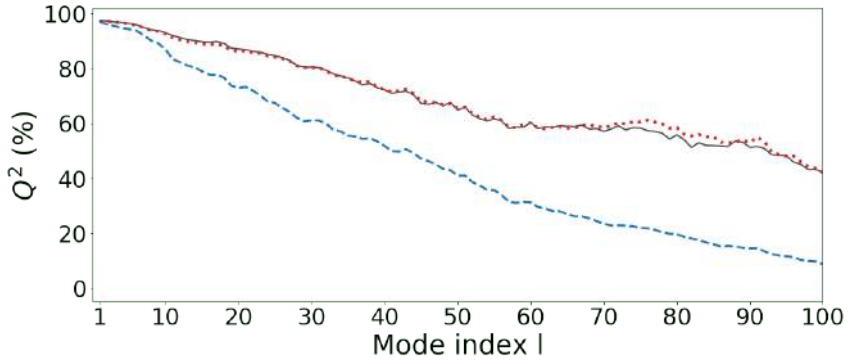

Figure IV.19: Per mode-Q^2 (Eq. IV.2) for Gaussian process regression models with noise and length-scales that are either imposed using prior information (dashed line) or optimised by MLL (solid line) or MAP (dotted line).

To conclude this section, there is a way to make Gaussian process regression more efficient without losing accuracy. This can be done through the implementation of a MAP procedure, which requires defining a non-uniform prior distribution for each of the Gaussian process hyperparameters. This prior information is partly given through a statistical analysis of the POD reduced coefficients, showing that the dimensionality reduction step and the regression step are not independent. In the end, the response time of the MAP approach is about twenty times quicker than that of MLL for the same level of accuracy. The comparison is not completely fair since the MAP procedure was solved using GPU, while the MLL procedure was implemented on CPU.

IV.5 Sensitivity of Gaussian process regression to the training dataset

We now assess the ability of Gaussian process regression to emulate a more general and classical LES problem. Previous results were obtained by considering a large number of modes in the reduced basis with a rich training dataset ($N_{\text{train}} = 450$). Such a large LES database is impracticable in all realistic atmospheric dispersion cases involving larger computational domains and three-dimensional effects. In this section, we analyse how reducing the training dataset to $N_{\text{train}} = 100$ and even 50 snapshots impacts the regression performance. This number of snapshots corresponds to a more achievable budget in practice, even if it already represents a significant computational effort.

Similarly to Sect. IV.4, 90% of the training snapshots are used to determine the POD reduced basis, while the remaining 10% is used for calibration. This implies that the maximum number of modes in the reduced basis is directly equal to the POD training size (90 for $N_{\text{train}} = 100$; 45 for $N_{\text{train}} = 50$). The only difference is that here the whole dataset is used to optimise the Gaussian process regression models since the dataset is of very limited size. The test dataset remains the same as before to avoid introducing bias during the validation stage.

Figure IV.20 compares the evolution of per-mode Q^2 for both full and reduced training datasets. The Gaussian process regression model accuracy significantly decreases for the reduced database for all reduced coefficients. There is a faster linear decrease towards $Q^2 = 0$ than with the full training dataset (the threshold $Q^2 = 0$ is approximately reached for the fiftieth mode for $N_{\text{train}} = 100$ and the the twenty-fifth mode for $N_{\text{train}} = 50$). This suggests that the number of POD modes to consider in the reduced-order models should be reduced due to the too limited size of the training dataset.

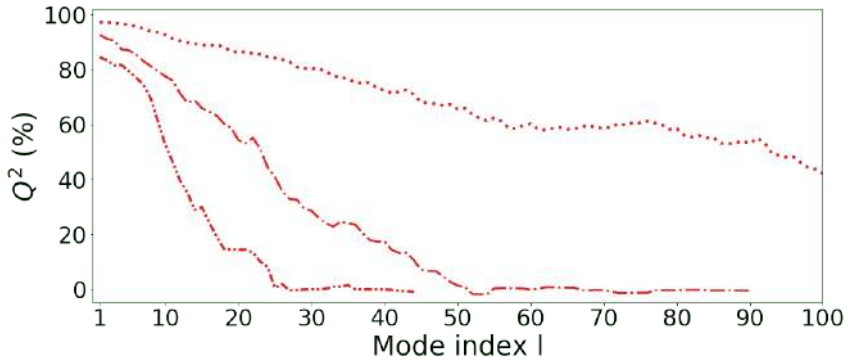

Figure IV.20: Per mode-Q^2 metric (in %, see Eq. IV.2) for Gaussian process regression model trained with 472 snapshots (dotted line corresponding to the MAP result already presented in Fig. IV.19), 100 snapshots (dash-dotted line) or 50 snapshots (densely dash-dotted line).

Including higher-order modes in the reduced-order model can even degrade its global performance. Here, the optimal choice for $N_{\text{train}} = 100$ is to keep $L = 80$ modes and $L = 38$ modes for $N_{\text{train}} = 50$. The associated global Q^2 score is equal to 90.8% and 83.7%, respectively, and can be compared to 96.8% for the full training dataset (Sect. IV.4.2). These scores may seem satisfactory but when looking at the nominal snapshot prediction, the prediction errors observed before are amplified. Figure IV.21 presents the nominal snapshot prediction result for $N_{\text{train}} = 100$. The shape of the tracer concentration wake is retrieved but the reduced-order

model has difficulty to predict the correct tracer concentration magnitude: the tracer concentration is underestimated at the emission source and upstream of the obstacle, and the high tracer concentrations correspond to a much thinner region than for the full training dataset (Fig. IV.13). This highlights that the analysis of the overall Q^2 score can be misleading about the reduced-order model accuracy, since the prediction quality is spatially heterogeneous and can fastly degrade upstream of the obstacle due to the large tracer concentration gradients near the emission source.

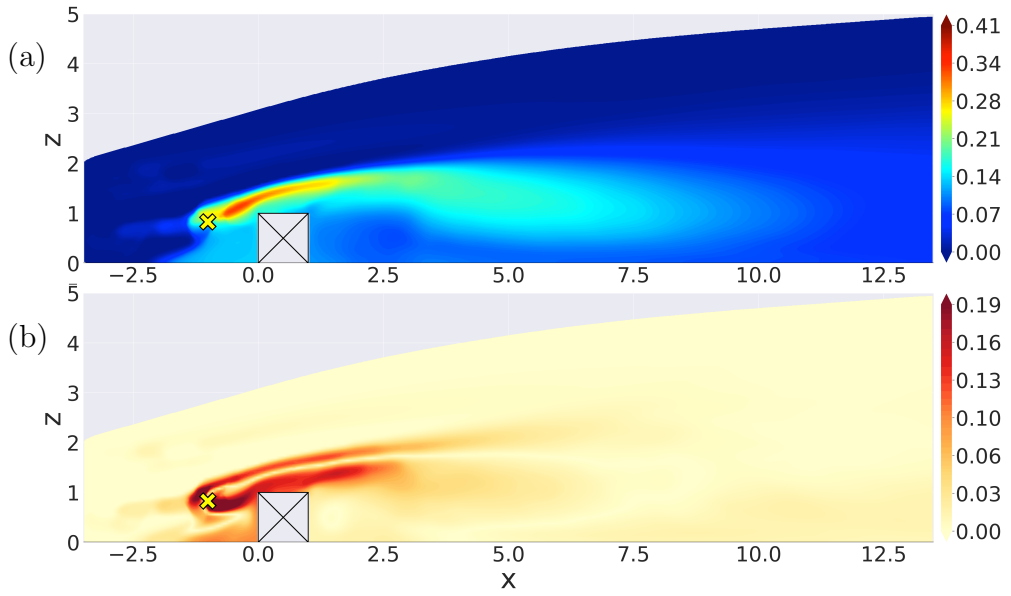

Figure IV.21: Same caption as in Fig. IV.13 for Gaussian process regression model prediction with $N_{\text{train}} = 100$ training snapshots.

Even if the prediction performance is reduced when the training dataset includes 100 snapshots, the Gaussian process regression model predictions remain physically-consistent. This is no longer the case when further reducing the training dataset since some non-physical large-scale structures appear in the predicted tracer concentration fields. Figure IV.22 shows three examples of predictions from the test dataset when considering 50 LES snapshots in the training dataset. Downstream of the obstacle, the plume shape remains relatively consistent with the reference LES statistics despite non-negligible magnitude differences (particularly visible in Fig. IV.22c near the source). However, physical consistency is lost upstream of the obstacle: there is no clear emission source position in the emulated fields (especially in Fig. IV.22ab); the low concentration magnitude isolines are much coarser, and spurious structures appear. This loss of physical consistency becomes worse when only considering 25 LES snapshots in the training dataset in Fig. IV.23.

To conclude this section, the minimum budget required to emulate the mean tracer concentration field from LES data using POD combined with Gaussian process regression is of the order of 100 LES snapshots in this configuration. We also analysed the emulation sensitivity to the level of noise in the LES data by decreasing the time-averaging window (not shown). In principle, this is an important aspect of robustness but the Gaussian process regression approach was found to be relatively insensitive to noise.

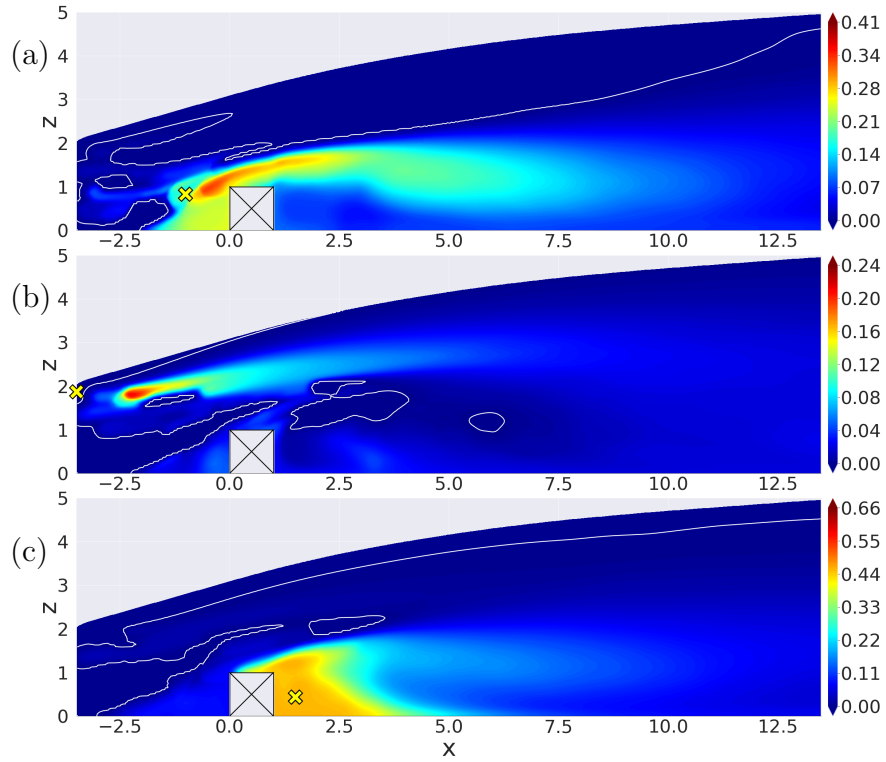

Figure IV.22: Gaussian process regression model predictions of mean normalised tracer concentration field with $N_{\text{train}} = 50$ training snapshots obtained for three LES test snapshots (see the reference solutions in Fig. III.8). White lines correspond to the mean normalised tracer concentration contour line equal to 5×10^{-4} to indicate low-magnitude noisy structures.

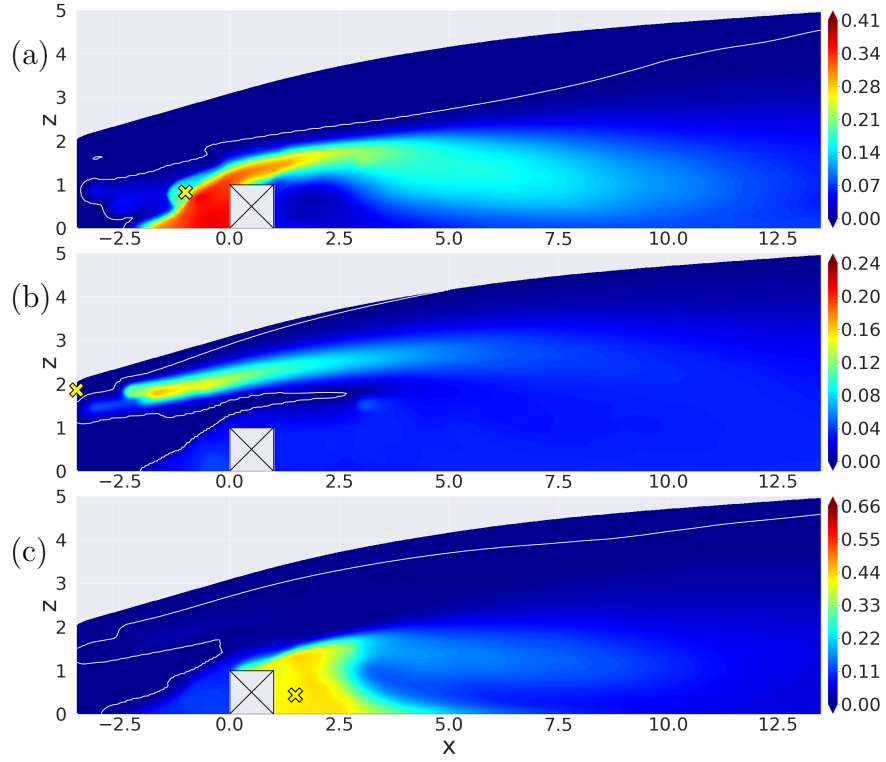

Figure IV.23: Same caption as in Fig. IV.22 but for $N_{\text{train}} = 25$ snapshots in the training dataset.

IV.6 Improving Gaussian process regression using deep learning

Until now we have explored the ability of POD-based reduced-order models to emulate mean tracer concentration fields. The approach combining POD with Gaussian process regression was shown to be sufficiently versatile to ensure proper emulation of concentration field statistics when substantial data are used for training. As POD is based on linear algebra, decomposing information with significant nonlinearity can be tricky. In this work, this issue translates into the large number of modes ($L = 100$) required in the reduced basis to reconstruct sufficient information upstream of the obstacle, highlighting the inability of POD to effectively reduce the physical space dimension. However, such a large number of modes is not feasible when LES training data are limited. To overcome this issue, we explore here the capacity of deep-learning-inspired techniques such as convolutional autoencoders [Fukami et al., 2020] to improve concentration field compression while meeting the constraint of reduced training dataset.

As explained in Sect. II.2.2, convolutional autoencoders can be considered as a nonlinear extension to POD. In the following, we present the architecture we adopt for the convolutional autoencoder and its implementation for data living on unstructured grids before evaluating its compression and emulation ability for our case study.

IV.6.1 Convolutional autoencoder structure

IV.6.1.a LES snapshots interpolation as preliminary step

Convolutional autoencoders cannot be directly applied to unstructured meshes, because their convolutional kernels rely on a cartesian grid. For this reason, the autoencoder is not implemented on the original unstructured mesh. A linear interpolation is used to map the LES data on a uniform mesh that is slightly larger (20 cm larger in each direction) than the region of interest $[-3.5, 13.5]$ m $\times$ $[0, 5]$ m to avoid difficulties on the boundaries associated with convolutional layer padding. The spatial discretisation of the uniform cartesian grid is chosen in adequation with that of the unstructured mesh, resulting in a spatial resolution of 17 cm and a total of 25,960 grid points. Resulting tensors are of dimension $\mathbb{R}^{295\times88}$.

IV.6.1.b Autoencoder neural-network architecture

As discussed in Sect. II.2.2.b and following the choices made by Murata et al. [2020], the convolutional autoencoder we adopt follows a bottleneck architecture shown in Fig. IV.24 with symmetric operations between the encoding part and the encoding part. First, the encoder reduces the original high-dimensional input tensor X to a low-dimensional latent vector γ. Then, the decoder transforms the latent vector back to the original high-dimensional image resulting in $Y \approx X$. The layers closest to the latent space correspond to a dense multilayer perceptron to handle the highly compressed data, which is surrounded by convolutional layers to handle the high-dimensional tensors at the network input and output. This architecture was already used in the work of Murata et al. [2020].

The choice of activation functions is an important point of attention as discussed in Sect. II.2.2.b.

We only use hyperbolic tangent (Tanh) functions on the hidden layers; only the last layers near the output Y contain the activation functions Sigmoid, ReLU and Softplus. We have chosen to primarily use Tanh for its symmetry properties and to avoid the use of sparse layers (e.g. ReLU) and gradient discontinuity. A single exception is made at the end of the network with the introduction of a unique ReLU layer: this choice was made to reduce the computational cost of the neural-network training and no drop in performance was observed for a single ReLU layer (adding more ReLU layers would reduce the neural-network prediction accuracy). The final sequence of activation functions – Sigmoid, ReLU and Softplus – retained in the final version of our autoencoder neural-network corresponds to a progressive debiasing of the mean concentration fields to obtain a positive output Y. Moreover, Softplus is adopted in the last layer to provide a better estimation of low tracer concentration values.

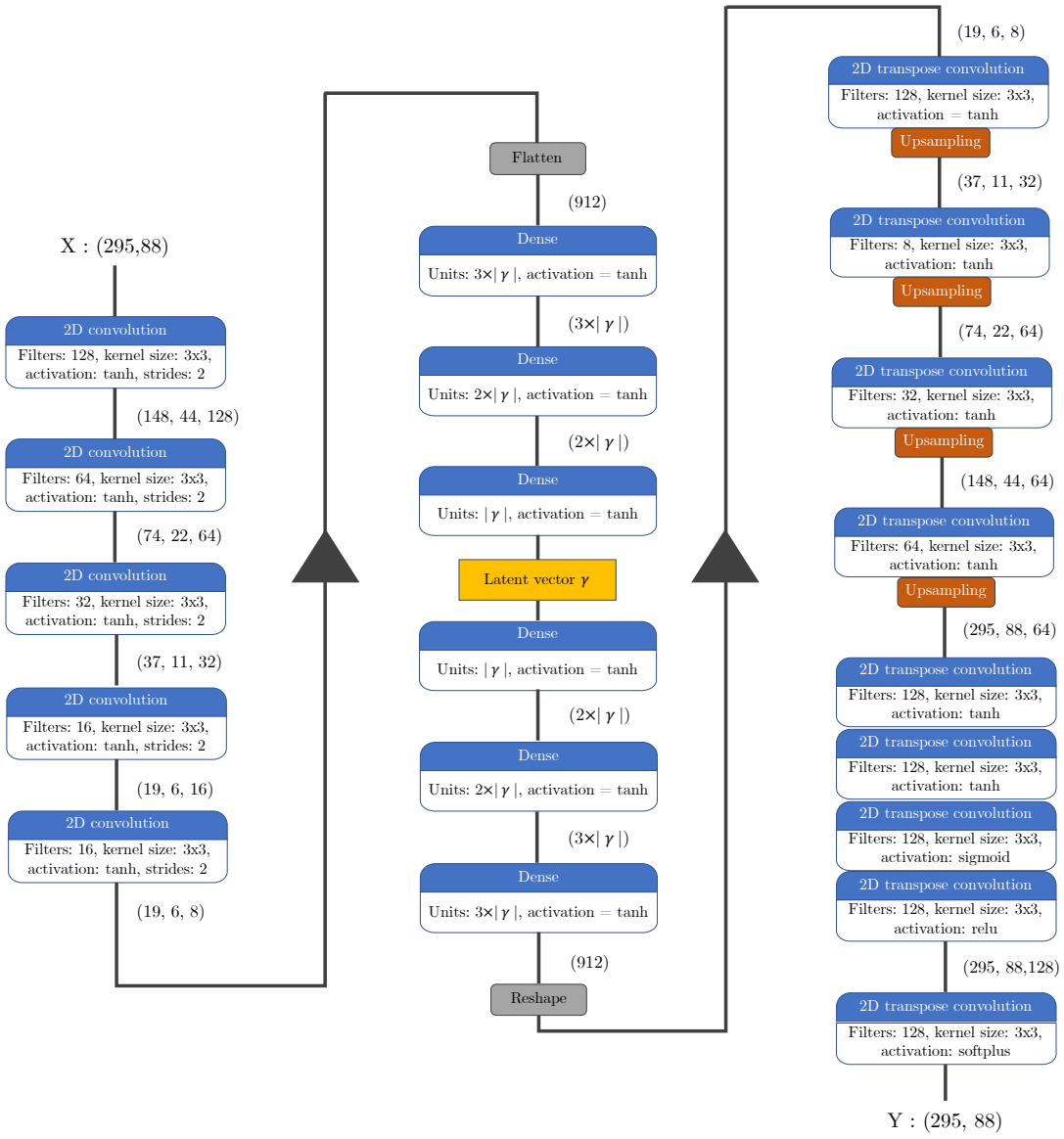

Figure IV.24: Convolutional autoencoder architecture consisting of (i) convolutional layers (first and last columns) to handle high-dimensional tensors and limit the number of network weights, and (ii) a dense multilayer perceptron (middle column) to deal with highly compressed data. The dimension of the latent vector γ which is a hyperparameter to be specified by the user. Each box describes a layer operation and input/output tensor dimensions are specified outside in brackets.

It is worth mentioning that the number of channels is large near the network input and output and is reduced near the latent space to allow for greater flexibility in the size of the central multilayer perceptron. Inside of the dense network, the dimension is reduced progressively proportionally to the size of the latent space.

IV.6.1.c Training metric

Training the network was performed minimising the MSE (which is equivalent in our case to Q^2 maximisation). The cartesian grid points associated with prediction of non-relevant areas (for instance, inside the obstacle) are assigned a zero weight and are not accounted for in the MSE formulation. The Adam method is used to optimise the network and to change the learning rate and the mini-batch size during training. For the first steps, the learning rate was set to 10^{-4} with ten-snapshot mini-batches. Near convergence (after 500 to 2,000 iterations depending on the latent space dimension and the training set size), the learning rate was lowered to 10^{-5} for online learning during a few iterations (5 to 10).

IV.6.2 Convolutional autoencoder performance

In this section, we evaluate the performance of the convolutional autoencoder in terms of compression and prediction, and we compare it to our baseline approach combining POD and Gaussian process regression.

IV.6.2.a Compression performance

Statistical analysis. Similarly to the POD performance evaluation in Sect. IV.2, the convolutional autoencoder compression performance is shown in Table IV.4. The compression performance is presented through the Q^2-explained variance criterion for different sizes of the training dataset and of the latent space. Results show that the convolutional autoencoder outperforms POD in every tested configuration, both globally and per area.

When considering the full training dataset (made of 450 LES snapshots) and 25 latent variables, the convolutional autoencoder achieves a global score of $Q^2_{\text{global}} = 99.7\%$, which is better than the POD score obtained with 25 modes (97.2%) and even 100 modes (99.3%, see Table IV.1). Still, the convolutional autoencoder is subject to the same bias towards better representation of high variance areas. Indeed, higher ensemble variance area obtains a better score ($Q^2_{\text{T}_2} = 99.7\%$) than middle and lower range ensemble variance areas ($Q^2_{\text{T}_1} = 99.3\%$ and $Q^2_{\text{T}_0} = 98.3\%$, respectively) .

The ability of the convolutional autoencoder to further compress the information to only 10 latent variables is also evaluated in Table IV.4. There is a slight loss of information with a 0.7% drop in the global Q^2 when moving from 25 to 10 latent variables. However, this loss of information is not homogeneously distributed over the domain; it is mainly concentrated in the low-variance area where the difference in Q^2 is 7.1% (compared to equal or less than 1.1% in the high- and medium-variance areas). This suggests that 25 latent variables are not excessive for a learning database of 450 snapshots, and that the additional 15 features carry relevant

Table IV.4: Explained variance ratio (Q^2 in %) obtained for the convolutional autoencoder over the test dataset configured with varying dimension of the latent space and of the training dataset (a comparison to POD performance is given in brackets). The global Q^2 criterion represents the compression performance on all grid points. Local explained variance information can be obtained for lowest variance region $Q^2_{T_0}$, medium variance region $Q^2_{T_1}$ and largest variance region $Q^2_{T_2}$.

Training size N_{train}	Latent space dimension L	Q^2_{global}	Q^2_{T0}	Q^2_{T1}	Q^2_{T2}
450	25	99.7 (97.2)	98.3 (87.1)	99.3 (94.0)	99.7 (97.5)
450	10	99.0 (92.6)	91.2 (80.6)	98.2 (88.3)	99.1 (93.1)
100	10	96.7 (92.2)	89.3 (76.2)	94.9 (87.5)	96.9 (92.7)
50	10	92.6 (90.8)	72.6 (72.5)	91.2 (85.4)	92.8 (91.4)
50	5	89.8 (84.3)	66.2 (63.6)	84.5 (80.6)	90.4 (84.8)

information about the mean concentration spatial patterns, mainly in areas that represent a smaller share of the ensemble variance.

As in Sect. IV.5, we test the sensitivity of the convolutional autoencoder compression performance to the learning dataset by reducing the training size from 450 to 100 and even 50 snapshots. The dimension of the latent space is reduced accordingly to 10 and even 5. When considering 10 latent variables, the global Q^2-performance equals 96.7% and 92.6% for 100 and 50 training snapshots, respectively. It should be noted that reducing the number of training snapshots mainly affects the areas of low ensemble variance. While the difference in global Q^2 is 6.4% between 450 and 50 snapshots, the difference in the low-variance area Q^2 reaches 18.6% ($Q^2_{T_0}$), while it is limited to 6.3% in the high-variance area ($Q^2_{T_2}$). These results are consistent with the trends observed for POD. It should also be noted that the choice of 10 latent variables for 50 training snapshots seems to be a coherent choice since there is a non-negligible difference in performance when compared to 5 latent variables.

Snapshot reconstruction example. The reconstruction of the reference nominal snapshot in Fig. IV.25 reflects the improved quality when using a convolutional autoencoder instead of POD for dimension reduction. We can observe that for all configurations (varying training size and latent space dimension), the plume shape matches well the reference LES, both downstream of the obstacle and upstream near the emission source. When considering the full training dataset but only 25 latent variables (Fig. IV.25b), we retrieve almost identical levels of tracer concentration, including near the emission source. Reducing the training dataset, for instance to 100 or 50 LES snapshots (Fig. IV.25cd), has a noticeable impact on the plume shape close to the emission source and near the upstream face of the obstacle. In Fig. IV.25c, the peak concentration is shifted downstream from the actual source position. The autoencoder training procedure is not stable, and some behaviour is difficult to anticipate. A larger training database does not increase reconstruction quality consistently as shown in Fig. IV.25cd; the solution obtained from the autoencoder trained with 50 LES snapshots appears to be more accurate than the solution obtained from the autoencoder trained with 100 LES snapshots.

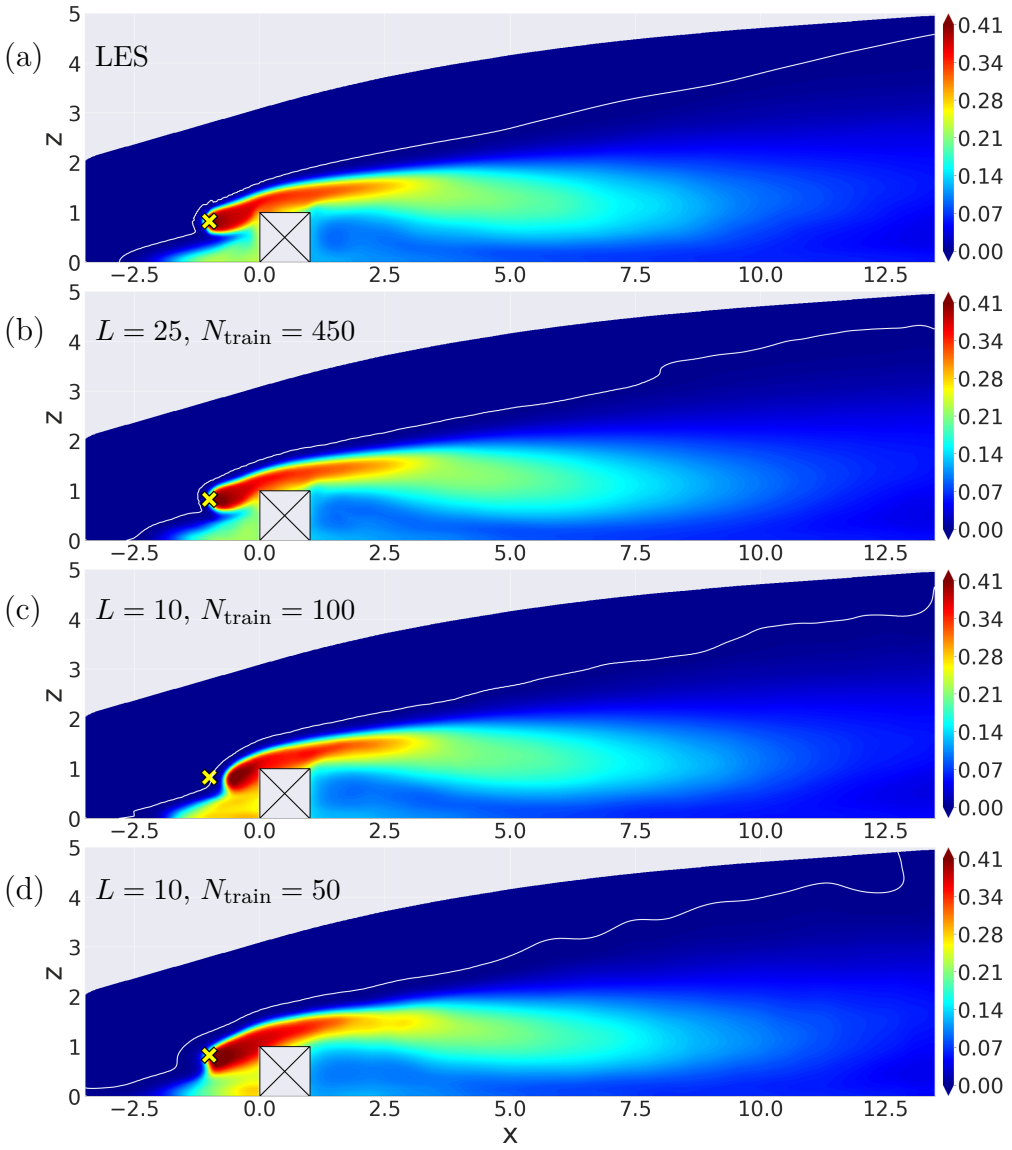

Figure IV.25: Reconstruction of the (a) LES nominal snapshot of mean normalised tracer concentration field, obtained from autoencoder with training dataset and latent space dimension of (b) $N_{\text{train}} = 450$ and $L = 25$ modes, (c) $N_{\text{train}} = 100$ and $L = 10$ modes, and (d) $N_{\text{train}} = 50$ and $L = 10$ modes. Contour line of the mean normalised tracer concentration equal to 5×10^{-4} is superimposed on mean tracer concentration fields to highlight low-magnitude concentration patterns.

IV.6.2.b Prediction performance

We now evaluate the prediction performance of the full reduced-order model that includes the convolutional autoencoder for dimension reduction and a Gaussian process regression model for representing the dependency of each latent variable to the four uncertain parameters $\boldsymbol{\mu} = (u_{z_c}, z_0, x_{\text{src}}, z_{\text{src}})^T$. This implies that the reduced-order model architecture remains identical to the previous sections, with the difference that the POD step is now replaced by the convolutional autoencoder to reduce the dimension of mean concentration fields. An interesting question is to analyse how Gaussian process regression models can deal with the nonlinearities introduced by the latent space of the convolutional autoencoder.

Statistical analysis. Table IV.5 summarises the prediction performance of the autoencoder-based reduced-order model for different configurations of the training dataset (the number

of training snapshots varies from 450 to 50) and of the latent space (the dimension of the latent space varies from 25 to 5 in adequation with the number of training snapshots as in Sect. IV.6.2.a).

Table IV.5: Explained variance ratio (Q^2 in %) obtained for the convolutional autoencoder-based reduced-order model over the test dataset for varying dimension of the latent space and of the training dataset (the Q^2 statistics for the POD-based reduced-order model are given in brackets for comparison). The global Q^2 criterion represents the prediction performance on all grid points. Local explained variance information can be obtained for lowest variance region $Q^2_{T_0}$, medium variance region $Q^2_{T_1}$ and largest variance region $Q^2_{T_2}$ as in previous analyses.

Training size N_{train}	Latent space dimension L	Q^2_{global}	Q^2_{T0}	Q^2_{T1}	Q^2_{T2}
450	25	97.9 (95.2)	92.4 (82.0)	96.8 (91.2)	98.0 (95.6)
450	10	97.4 (91.1)	87.1 (76.1)	96.7 (86.1)	97.5 (91.7)
100	10	93.4 (88.6)	76.0 (71.5)	89.9 (81.1)	94.5 (89.3)
50	10	84.8 (83.1)	43.4 (60.9)	74.9 (74.0)	85.9 (84.1)
50	5	81.6 (78.7)	73.3 (56.5)	84.0 (72.9)	81.5 (79.4)

Results show that when considering the full training dataset, the autoencoder-based reduced-order model improves the prediction performance compared to the POD-based reduced-order model. With 25 latent variables, it achieves a global Q^2-metric of 97.9%, which is 3% higher than for the POD-based reduced-order model (this corresponds to a MSE decrease of 60%). A more significant improvement occurs in the low-variance area with a 10%-difference in $Q^2_{T_0}$ between autoencoder- and POD-based reduced-order model predictions.

When reducing the training dataset, for instance in the case of 10 latent variables and 50 training snapshots, the difference between autoencoder- and POD-based reduced-order model predictions is narrowing, and even the autoencoder performance in the low-variance area ($Q^2_{T_0}$) drops to 43.4%, well below the POD performance (60.9%). This suggests that the POD-based reduced-order model could be more robust for predicting the physical processes in the low-variance areas when the number of training snapshots is very limited.

It should be noted that in the case of 10 latent variables and 50 training snapshots the convolutional autoencoder compression performance is very good (92.6% for the global Q^2 and 72.6% for $Q^2_{T_0}$, see Table IV.4. The lack of training snapshots is therefore difficult to handle for the Gaussian process regression mode component. This may be due to the high complexity of the response surface on some latent variables. An argument in favour of this explanation is the much better $Q^2_{T_0}$-performance achieved in the case of 5 latent variables and 50 training snapshots (73.3% for 5 latent variables comapred to 43.4% for 10 latent variables). Decreasing the dimension of the latent space may be a way to have only smooth response surfaces and thus improve the robustness of the emulation process in low-variance areas. However, this will be at the expense of areas where the variance is better represented (for instance, the Q^2-score in the large variance area – $Q^2_{T_2}$ – is 4.5% higher when considering 10 latent variables instead of 5).

Snapshot prediction example. The prediction of the reference nominal snapshot in Fig. IV.26 also reflects the improved quality of emulation when using a convolutional autoencoder instead

of POD for dimension reduction. Similarly as for the reconstruction step (Sect. IV.6.2.a), we can observe that for all configurations (varying training size and latent space dimension), the plume shape still matches well the reference LES solution, both downstream of the obstacle and upstream near the emission source. When considering the full training dataset but only 25 latent variables (Fig. IV.26b), we retrieve the right levels of tracer concentration, including near the emission source, which is not the case for the POD-based reduced-order model (Fig. IV.13). As already observed for the reconstruction, the only exception is the concentration close to the upstream face of the obstacle that is slightly overestimated by the autoencoder-based reduced-order model. Reducing the training dataset, for instance to 50 snapshots (Fig. IV.26c), has a slight impact on the plume shape near the emission source and on the peak concentration location. Nevertheless, it is intriguing to observe that the position error of peak concentration observed for reconstruction in Fig. IV.25c does not occur for prediction in Fig. IV.26c. This means that the prediction outperforms the reconstruction, which is unexpected. The exact reason for such behaviour has not yet been explored.

Although one can appreciate the quality of the autoencoder predictions, Fig. IV.26 also illustrates the difficulties arising from using complex convolutional autoencoders. It is difficult to identify the spatial patterns of the errors, and to understand where they come from. When considering the full training dataset (Fig. IV.26b), we observe concentration overprediction upstream and underprediction downstream of the obstacle, close to the ground. In contrast, when reducing the training dataset to 100 snapshots (Fig. IV.26c), it is inverted. When considering only 50 snapshots (Fig. IV.26d), we observe concentration underprediction upstream and downstream of the obstacle. The mechanisms governing the integration of information in the autoencoder seem rather unstable, which limits its explicability.

To conclude this section, convolutional autoencoder can lead to improved concentration field data compression and emulation when used instead of POD in the Gaussian process-based reduced-order model. However, some latent variables may be subject to strong nonlinearities with respect to the uncertain input parameters, which makes the regression problem more difficult to solve for Gaussian processes. The performance increase is achieved at the expense of model explicability. The nonlinear expression of the encoder and decoder no longer allows for an easy hierarchical representation of the latent variables (or modes). Further analysis of these nonlinearities is required to better understand the behaviour of the convolutional autoencoder-based reduced-order model when changing the dimension of the latent space and the training dataset. In complement, we further investigated innovative designs of hierarchical convolutional autoencoder architectures [Saegusa et al., 2004; Murata et al., 2020; Fukami et al., 2020] but our preliminary results were not conclusive, even compared to POD. Other dimension reduction tools were also investigated to replace POD, for instance non-negative matrix factorisation (NMF) and Gaussian process latent variable model (GPLVM). A GPLVM does not provide an encoder, which limits its use for reduced-order modelling, but preliminary tests showed interesting performance that could be further investigated.

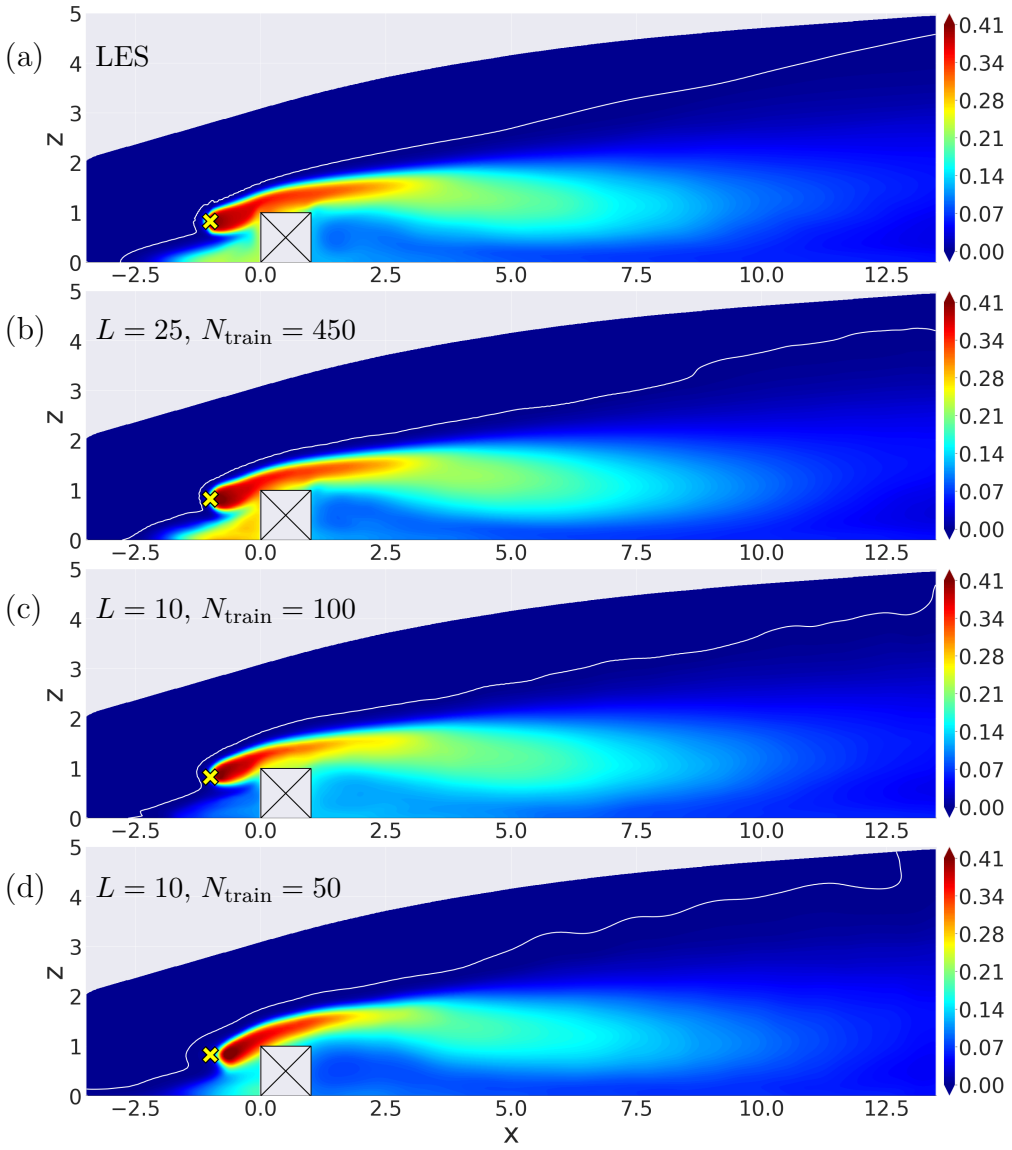

Figure IV.26: Nominal snapshot of mean normalised tracer concentration field obtained with (a) LES (b) $N_{\text{train}} = 450$ and $L = 25$ modes, (c) $N_{\text{train}} = 100$ and $L = 10$ modes and (d) $N_{\text{train}} = 50$ and $L = 10$ modes. Contour line of the mean normalised tracer concentration equal to 5×10^{-4} is superimposed on mean tracer concentration fields to highlight low-magnitude concentration patterns.

IV.7 Conclusion

In this study, the objective was to design and evaluate a non-intrusive reduced-order modelling approach suitable to emulate near-source tracer concentration field statistics simulated by LES in a multi-query uncertainty quantification context. This reduced-order modelling approach includes a dimension reduction approach to efficiently represent the fields of interest as a reduced set of latent variables or modes, and a regression model to map the evolution of these latent variables with respect to uncertain inputs. A two-dimensional case study corresponding to a turbulent atmospheric flow over a surface-mounted obstacle was considered to generate a large LES ensemble database (750 snapshots in total) based on perturbed inflow boundary conditions and emission source position and height. The performance of the different configurations of the reduced-order model was evaluated through the Q^2-metric and was illustrated here through the example of the mean tracer concentration field.

POD was used as the baseline approach for dimension reduction since it is widely used approach in CFD problems. Low-order modes were found to carry concentration information in widely-spread areas and near the obstacle, whereas high-order modes were found to characterise the concentration high variability near the emission sources upstream of the obstacle. To accurately represent this concentration variability, it was necessary to include a large number of POD modes (up to 100) in the reduced basis, which multiplies the number of regression problems to be solved. To overcome POD limitations, we implemented a convolutional autoencoder neural network inspired from the work by Murata et al. [2020], combining convolutional layers and a dense multilayer perceptron. The convolutional autoencoder demonstrated remarkable compression capabilities (25 latent variables were sufficient instead of 100 modes for POD) but this came at the expense of model explicability. Further research on the hierarchical expression of latent variables is needed for this approach to become a good candidate for reduced-order modelling, especially for risk assessment studies, where there is a need for interpretability.

The regression component was formulated as a set of regression subproblems, with the key idea of building a regression model for each POD mode, since each POD mode has its own length-scales and since a significant performance gain is obtained by optimising the regression model hyperparameters mode-by-mode. The choice of the regression model class is reported to be problem-dependent in the literature. To find the most suitable class for the near-source atmospheric dispersion problem studied in this work, a detailed comparison of several regression model classes was carried out, including the well-known polynomial chaos expansion, the Gaussian process regression model and even gradient tree boosting, i.e. a more recent machine learning approach combining decision trees and a boosting procedure. The Gaussian process regression model performed by far the best in mapping the POD reduced coefficients over a wide range of input parameter variation. The high-order modes were found to be the most challenging to emulate. They feature very localised spatial structures associated with perturbed emission location, which are difficult to predict and which are prone to more noise than low-order modes. This issue worsens when the training database is reduced. Still, the Gaussian process regression model prediction performance remained acceptable when considering about 100 LES training snapshots. Moreover, the Gaussian process hyperparameter optimisation was made more efficient by running a MAP optimisation procedure informed by hyperparameter prior distributions and requiring a single gradient descent per POD mode (instead of a standard N-restart MLL maximisation approach). All these results tend to demonstrate that Gaussian process regression has the potential to tackle real cases of pollutant dispersion problems such as the MUST field-scale experiment.

Most of this chapter content is the subject of a first article “Reduced-order modeling for parameterized large-eddy simulations of atmospheric pollutant dispersion”, which is currently under revision for publication in the Stochastic Environmental Research and Risk Assessment (SERRA) journal [Nony et al., 2022]. The article focuses on the performance evaluation of the reduced-order model combining POD and Gaussian process regression. The intercomparison of regression models (Sect. IV.3) and the evaluation of convolutional autoencoder-based reduced-order model (Sect. IV.6) are recent complements to this article.

Chapter V

Reduced-order model based on LES-informed Reynolds-averaged tracer transport equation

The ability to emulate tracer concentration fields in a parametric setting obtained from an ensemble of LES using a reduced-order model was demonstrated in Chapter IV. The emulation process allows to accurately reproduce the LES predictions, while reducing the computational cost to query a snapshot for a new set of parameters by several orders of magnitude. However, a large LES training dataset is required to achieve accurate emulation without artefacts, which may be out of reach for practical three-dimensional realistic applications. The need for a large training dataset is mostly due to fine plume structures in the near-source regions caused by uncertainty in the emission source location. By reducing the number of training snapshots, a loss of consistency with physics principles has also been observed: for instance, non-physical noisy structures may appear in tracer-free regions (as seen in Fig. IV.21), as the reduced-order model is built in a purely data-driven manner, which is not constrained by physics principles. To overcome these limitations, this chapter presents an alternative hybrid RANS/reduced-order modelling approach, which is based on the key idea of injecting detailed flow information from LES into a lower fidelity tracer transport equation in the RANS formalism. A reduced-order model combining POD with Gaussian process regression following the methodology of Chapter IV is used in this chain as intermediate step to efficiently represent the relevant LES statistics. This hybrid LES/RANS approach is compared to the direct approach of Chapter IV. Since models with different levels of fidelity (LES, hybrid LES/RANS) are available in this context, a multi-fidelity strategy is also investigated to exploit the benefits of each approach.

Contents

V.1 Construction of the hybrid RANS/reduced-order model approach

V.1.1 Principle

We propose an alternative approach to Chapter IV, which aims to combine emulation of relevant flow quantities from LES that are injected in a lower fidelity representation of the tracer dispersion. Similar ideas of exploiting the rich LES information with a simplified transport model for the tracer dispersion for fast inference are also found in the literature: for instance, Du et al. [2020] proposed a simplified transport model (*transport-based recurrence*) for the tracer, which is based on precomputed LES data of the flow statistics. This two-step process allows fast inference time for the evaluation of the tracer dispersion, while preserving the rich information of the LES dataset. However, their study is limited to a single LES case, which does not address

atmospheric parametric uncertainties. Following this idea of a two-step process, we propose in this chapter an hybrid method in parametric setting, which includes three steps:

1. parametric LES data generation;
2. emulation of the relevant LES statistics with a reduced-order model;
3. integration of the emulated LES statistics into a simplified transport model: for this simplified transport model, we make the choice to use a Reynolds average transport equation for the tracer, which is several orders of magnitude cheaper than a full LES (it could be inserted into other formalisms, like the recurrence CFD model of Du et al. [2020]).

Combining a reduced-order model and a more conventional transport equation for the tracer is expected to benefit from the advantages of each approach:

– A first strength of the method is to exploit the rich information of the LES dataset in an efficient way with the reduced-order model: exploiting the fact that the flow dynamics is decoupled from the tracer dynamics, we make the choice to build the approach by only emulating LES quantities related to flow properties (mean flow field, turbulent kinetic energy, etc.). This drastically reduces the number of uncertain parameters, as tracer related uncertainties are no longer required in the LES training database. Thus, the machine learning approach only handles atmospheric uncertainties (i.e. the reference velocity magnitude u_{z_c} and the aerodynamic roughness length z_0) to map relevant flow quantities. Since the flow quantities do not exhibit the fine characteristic structures related to tracer source location uncertainty, it is expected for POD to better reduce dimension of the quantities of interest and for Gaussian process regression to perform well from a smaller number of LES training samples compared to the direct approach of Chapter IV. The cost of the LES ensemble generation is thus expected to be lower, which makes the approach computationally appealing in the offline phase.

– The second strength of the method is that the uncertainties related to the emission source location are directly handled through the RANS tracer transport equation. The source position uncertainty is readily handled by the source term in the RANS transport equation for the tracer. Relying on a transport equation also guarantees essential physical properties (e.g. tracer conservation) and can smear out potential noise introduced by direct reduced-order model prediction of the quantities of interest, while being several orders of magnitude cheaper than a direct LES prediction. However, since the LES training data are agnostic to tracer dispersion properties, this implies to rely on conventional RANS closure for unclosed term in the tracer transport equation, which may limit the approach accuracy.

This hybrid approach is referred to as **EMUL-RANS-TE** (emulated quantities in a RANS transport equation) in the following. For the method construction and validation, LES field statistics can be used directly in the same framework instead of their emulated counterparts to bypass the emulation step. It allows to decouple error related to the lower order transport equation and to the reduced-order modelling errors: this approach used for validation is referred to as **LES-RANS-TE** (LES quantities in a RANS transport equation).

With the objective of constructing the hybrid framework, a first step is to establish the link between LES and RANS formalisms. The Reynolds-averaged governing equations are derived along with the modelling assumptions, which are verified with an *a priori* and *a posteriori* validation of the framework. After recalling the RANS governing equations, it is demonstrated by formally averaging the LES equations that the mean tracer field obtained from the LES equation can be obtained with a RANS transport equation. This can be done by injecting relevant LES statistics into these equations. The formalism is demonstrated to be exact if the true turbulent tracer flux (see example in Fig. III.5) can be inserted in the averaged tracer equation. However, since the LES data generation and emulation procedure is agnostic to the tracer dispersion here, an appropriate tracer turbulent flux closure is required. The nominal snapshot is used to highlight the impact of the RANS closure limitations in this context.

V.1.2 Conventional modelling of tracer transport in a RANS context

As discussed in Chapter I, the RANS equations can be obtained by averaging in time ($\overline{\cdot}$ operator) the original set of governing equations from Eq. (I.6) related to incompressibility constraint, momentum conservation and tracer transport conservation. They read:

$$\frac{\partial \overline{u}_i}{\partial x_i} = 0, \tag{V.1}$$

$$\frac{\partial \overline{u}_i \overline{u}_j}{\partial x_j} = -\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_j} + \frac{\partial}{\partial x_j}(2\nu \overline{s}_{i\,j}) - \frac{\partial}{\partial x_j}\left(\overline{u_i' u_j'}\right), \tag{V.2}$$

$$\frac{\partial \overline{\mathbf{K}}\overline{u_j}}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\frac{\nu}{\mathrm{Sc}}\frac{\partial \overline{\mathbf{K}}}{\partial x_j}\right) - \frac{\partial}{\partial x_j}\left(\overline{\mathbf{K}' u_j'}\right). \tag{V.3}$$

In these equations, unclosed terms related to second-order correlations between velocity components (the Reynolds stress tensor $\overline{u_i' u_j'}$) and velocity-tracer cross correlations ($\overline{\mathbf{K}' u_j'}$) appear. These unclosed terms require to define appropriate closures. A widely-used assumption to close the momentum equations is the Boussinesq assumption, which states that the Reynolds stress tensor is related to the mean velocity gradients in the form:

$$\overline{u_i' u_j'} = -2\,\nu_T^{\mathrm{RANS}}\,\overline{s}_{ij}\,, \quad \text{with } \overline{s}_{ij} = \frac{1}{2}\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right), \tag{V.4}$$

where ν_T^{RANS} is the turbulent eddy-viscosity. By analogy, a closure of the same form is usually adopted for the relation between velocity-tracer cross correlation and mean tracer gradient:

$$\overline{\mathbf{K}' u_j'} = -\frac{\nu_T^{\mathrm{RANS}}}{\mathrm{Sc}_T^{\mathrm{RANS}}}\left(\frac{\partial \overline{\mathbf{K}}}{\partial x_j}\right), \tag{V.5}$$

where $\mathrm{Sc}_T^{\mathrm{RANS}}$ is the turbulent Schmidt number.

In the closures in Eqs. (V.4)–(V.5), the estimation of ν_T^{RANS} is usually based on a transport equation for second-order statistics related to turbulence. A common model is the $k-\epsilon$ model, for which two transport equations are solved or the turbulent kinetic energy $\mathbf{k}_{tke} = \frac{1}{2}u_i' u_i'$ and

the turbulent dissipation rate $\boldsymbol{\epsilon} = \overline{2\nu s'_{ij} s'_{ij}}$. These two equations read:

$$\overline{u}_j \frac{\partial \mathbf{k}_{tke}}{\partial x_j} = 2\,\nu_T^{\mathrm{RANS}}\,\overline{S}_{ij}\overline{S}_{ij} - \boldsymbol{\epsilon} + \frac{\partial}{\partial x_j}\left(\left(\nu + \frac{\nu_T^{\mathrm{RANS}}}{\sigma_k}\right)\frac{\partial \mathbf{k}_{tke}}{\partial x_j}\right), \tag{V.6}$$

$$\overline{u}_j \frac{\partial \boldsymbol{\epsilon}}{\partial x_j} = \frac{\boldsymbol{\epsilon}}{\mathbf{k}_{tke}}\left(2C_{1\epsilon}\,\nu_T^{\mathrm{RANS}}\,\overline{S}_{ij}\overline{S}_{ij} - C_{2\epsilon}\,\boldsymbol{\epsilon}\right) + \frac{\partial}{\partial x_j}\left(\left(\nu + \frac{\nu_T^{\mathrm{RANS}}}{\sigma_\epsilon}\right)\frac{\partial \boldsymbol{\epsilon}}{\partial x_j}\right). \tag{V.7}$$

Based on these two quantities, the turbulent eddy-viscosity ν_T^{RANS} can be estimated as:

$$\nu_T^{\mathrm{RANS}} = C_\mu\,\mathbf{k}_{tke}\,\tau_T \quad \text{with } \tau_T = \frac{\mathbf{k}_{tke}}{\boldsymbol{\epsilon}}, \tag{V.8}$$

where τ_T is a relevant turbulent time scale that is assumed to be equal to the eddy turnover time. An alternative to Eq. (V.8) is to solve an additional transport equation for the turbulent eddy-viscosity itself, as proposed by Spalart and Allmaras [1994] or Yoshizawa et al. [2012]. In the model of Yoshizawa et al. [2012], the transport equation takes the following form:

$$\overline{u}_j \frac{\partial \nu_T^{\mathrm{RANS}}}{\partial x_j} = C_{\mu P}\,\boldsymbol{k}_{tke} - C_{\mu\epsilon}\,\frac{1}{\tau_T}\,\nu_T^{\mathrm{RANS}} + \nabla\cdot\left(\left(\nu + \frac{\nu_T^{\mathrm{RANS}}}{\sigma_\nu}\right)\nabla\nu_T\right), \tag{V.9}$$

where $C_{\mu P}$, $C_{\mu\epsilon}$, σ_ν are modelling constants associated with turbulent production, dissipation, and diffusion. Again, τ_T is a relevant flow time scale estimated by Yoshizawa et al. [2012] as:

$$\begin{cases} \tau_T = \dfrac{\boldsymbol{k}_{tke}}{\boldsymbol{\epsilon}\,\Lambda}, \\ \Lambda = \sqrt{1 + C_s\left(\dfrac{\boldsymbol{k}_{tke}}{\boldsymbol{\epsilon}}\,\overline{s}_{i\,j}\right)^2 + C_\Omega\left(\dfrac{\boldsymbol{k}_{tke}}{\boldsymbol{\epsilon}}\,\overline{\Omega}_{i\,j}\right)^2}. \end{cases} \tag{V.10}$$

Compared to the conventional $k-\epsilon$ model of Eq. (V.8), a correction factor Λ is introduced for the time scale estimation to blend the original time scale estimation with time scales related to the mean strain rate $\overline{s}_{ij}$ and mean vorticity tensors $\overline{\Omega}_{ij}$. The Yoshizawa model can be seen as an extension of the original $k-\epsilon$ model, which improves the capability to deal with non-local transport effects such as advection in turbulent flows. In the limit of negligible advection and diffusion transport in Eq. (V.9), the turbulent eddy-viscosity can be expressed as:

$$\nu_T^{\mathrm{RANS}} = \frac{C_{\mu P}}{C_{\mu\epsilon}}\,\boldsymbol{k}_{tke}\,\tau_T\,, \tag{V.11}$$

which recovers the same form as Eq. (V.8).

V.1.3 Link between LES and RANS formalisms

As the main goal of the method is to substitute LES prediction by RANS tracer transport equation for the passive tracer, the formal connection between the two approaches is demonstrated here. It also enables to identify the exact closure term for the RANS tracer equation: with this exact term, the mean tracer field obtained from the RANS solution should match the mean LES tracer field. This property is useful to guide the closure of the RANS equations.

In the context of LES, the transport equation for the passive tracer is recalled as:

$$\frac{\partial \widetilde{\mathbf{K}}}{\partial t} + \widetilde{u_j}\frac{\partial \widetilde{\mathbf{K}}}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\left(\frac{\nu}{\mathrm{Sc}} + \frac{\nu_T^{\mathrm{LES}}}{\mathrm{Sc}_T^{\mathrm{LES}}}\right)\frac{\partial \widetilde{\mathbf{K}}}{\partial x_j}\right) + R\,. \tag{V.12}$$

It is reminded that any LES-filtered quantity $\widetilde{\Phi}$ can be formally decomposed using Reynolds decomposition in a mean and fluctuating parts:

$$\widetilde{\Phi} = \overline{\Phi} + \widetilde{\Phi}'\,. \tag{V.13}$$

By applying the Reynolds operator to the LES tracer transport equation of Eq. (V.12), an equivalent time-averaged equation is obtained for the tracer transport:

$$\frac{\partial \overline{\mathbf{K}}}{\partial t} + \overline{u_j}\frac{\partial \overline{\mathbf{K}}}{\partial x_j} = \overline{\frac{\partial}{\partial x_j}\left(\left(\frac{\nu}{\mathrm{Sc}} + \frac{\nu_T^{\mathrm{LES}}}{\mathrm{Sc}_T^{\mathrm{LES}}}\right)\frac{\partial \widetilde{\mathbf{K}}}{\partial x_j}\right)} - \frac{\partial}{\partial x_j}\overline{\widetilde{u_j}'\widetilde{\mathbf{K}}'} + R\,, \tag{V.14}$$

which has the same form as the original equation of Eq. (V.3). This implies that the unclosed RANS turbulent tracer flux of the original model can be identified from the LES data as:

$$\overline{u_j'\mathbf{K}'} = \overline{\widetilde{u_j}'\widetilde{\mathbf{K}}'} - \overline{\frac{\partial}{\partial x_j}\left(\frac{\nu_T^{\mathrm{LES}}}{\mathrm{Sc}_T^{\mathrm{LES}}}\frac{\partial \widetilde{\mathbf{K}}}{\partial x_j}\right)}\,. \tag{V.15}$$

This equivalency between time-averaged LES and RANS equations is verified in Sect. V.1.5. This demonstrates the potential of a RANS tracer transport equation to predict the mean tracer concentration field, provided that a satisfactory closure for the turbulent tracer fluxes $\overline{u_j'\mathbf{K}'}$ is found. In this perspective, as the "true" closure can be obtained from LES data using Eq. (V.15), it is used here to evaluate the relevant RANS closure and potential model deficiencies.

V.1.4 LES and RANS coupling in the hybrid approach

The spirit of the hybrid approach (**EMUL-RANS-TE**) is to solve the tracer transport equation of Eq. (V.3) by feeding the relevant LES information. The idea of injecting rich data (DNS or LES) into lower-order equations such as RANS is a common strategy found in the literature, in particular to build machine learning based closure of RANS equations: from this injection, corrective closures terms can be learned [Steiner et al., 2022] or inverse modelling can be used to infer corrective fields [Parish and Duraisamy, 2016], in order to improve RANS accuracy, with significant generalisation capability. In the present case, we take a more direct approach, as we do not seek generalisation capability in this context: the main mean flow quantities (velocity components, kinetic energy, and turbulent dissipation) are directly replaced by the LES fields or their emulated counterparts (noted with a $\star$ superscript in the following). This has the advantage of discarding the resolution of these quantities with a conventional RANS approach, which is subjected to significant model sensitivity, as highlighted for instance by Rodi [1997] for RANS calculations of surface-mounted obstacles, in particular for the prediction of the turbulent kinetic energy. For the reduced-order modelling perspective, these quantities are relatively straightforward to emulate, as *i*) they can be represented by a small number of POD

modes, and *ii*) they do not depend on the tracer source parameters as flow quantities are not dependent on the tracer dynamics.

With this substitution, the tracer transport equation of Eq. (V.15) reads:

$$\overline{u_j}^{\star}\frac{\partial\overline{\mathbf{K}}}{\partial x_j}=\frac{\partial}{\partial x_j}\left(\frac{\nu}{\mathrm{Sc}}\frac{\partial\overline{\mathbf{K}}}{\partial x_j}\right)-\frac{\partial}{\partial x_j}\left(\overline{\mathbf{K}'u_j'}\right)\,. \tag{V.16}$$

where $\overline{u_j}^{\star}$ is the emulated time-averaged velocity jth component from LES data.

The turbulent transport closure $\overline{\mathbf{K}'u_j'}$ in Eq. (V.16) is obtained by adopting the transport equation for the turbulent eddy-viscosity from Yoshizawa et al. [2012] (Eq. V.9):

$$\overline{u_j}^{\star}\,\frac{\partial\nu_T^{\mathrm{RANS}}}{\partial x_j}=C_{\mu P}\,\boldsymbol{k}_{\mathrm{tke}}^{\star}-C_{\mu\epsilon}\,\frac{1}{\tau_T^{\star}}\,\nu_T^{\mathrm{RANS}}+\nabla\cdot\left(\left(\nu+\frac{\nu_T^{\mathrm{RANS}}}{\sigma_\nu}\right)\nabla\nu_T^{\mathrm{RANS}}\right)\,. \tag{V.17}$$

It should be noted that the terms $\boldsymbol{k}_{\mathrm{tke}}^{\star}$ and $\tau_T^{\star}$ can be directly extracted for the LES solution. This removes the error associated with predicting these quantities, which can be difficult in a RANS context. It should also be noted that the use of the Yoshizawa transport equation is preferred to the direct algebraic model as *i*) it can smear out noise in the injected LES data, and *ii*) from a physical perspective it better deals with non-local advection effects.

Once the turbulent eddy-viscosity ν_T^{RANS} is obtained, the turbulent tracer flux can then be estimated with different forms of closure. Detailed investigations of different closures in a RANS context have been performed by Rossi et al. [2010] and Gamel [2015]. The main available closures are detailed below.

– The **standard gradient diffusion hypothesis (SGDH)** assumes that the diffusion is isotropic and occurs in the direction of the mean tracer gradient:

$$\overline{\mathbf{K}'u_j'}=-\frac{\nu_T^{\mathrm{RANS}}}{\mathrm{Sc}_T^{\mathrm{RANS}}}\,\frac{\partial\overline{\mathbf{K}}}{\partial x_j}\,. \tag{V.18}$$

– Anisotropic model with a tensorial tracer diffusivity has also been proposed and is investigated here. A first model from Daly and Harlow [1970] is the **generalised gradient diffusion model (GGDH)**, which takes the form:

$$\overline{\mathbf{K}'u_j'}=-\frac{\nu_T^{\mathrm{RANS}}}{\mathrm{Sc}_T^{\mathrm{RANS}}}\,\frac{\overline{u_i'u_j'}}{\boldsymbol{k}_{tke}}\,\frac{\partial\overline{\mathbf{K}}}{\partial x_i}\,. \tag{V.19}$$

This model was further refined by Abe and Suga [2001] in the form of the **high-order generalised gradient diffusion model (HOGGDH)**:

$$\overline{\mathbf{K}'u_j'}=-\frac{\nu_T^{\mathrm{RANS}}}{\mathrm{Sc}_T^{\mathrm{RANS}}}\,\frac{\overline{u_i'u_k'}\,\overline{u_k'u_j'}}{\boldsymbol{k}_{tke}^2}\,\frac{\partial\overline{\mathbf{K}}}{\partial x_i}\,. \tag{V.20}$$

In both the GGDH and HOGGDH models, the isotropic eddy-diffusivity is replaced by a tensorial representation, which can account for the anisotropic nature of turbulent tracer transport and therefore give more reliable tracer transport prediction [Rossi, 2010].

V.1.5 Validation of the turbulent tracer flux closure

A priori validation. To determine the turbulent tracer flux closure to retain in the hybrid approach, the different aformentioned closures (SGDH, GGDH and HOGGDH) are evaluated *a priori*, meaning that the ground truth LES mean tracer field $\overline{\mathbf{K}}$ is injected in the closure forms of Eqs. (V.18)–(V.19)–(V.20). This is useful to evaluate the accuracy of the different closures by comparing the resulting horizontal and vertical components of the turbulent tracer flux $\overline{\mathbf{K}'u_j'}$ with the ground truth obtained from LES data (Eq. V.15). This *a priori* evaluation is performed on the nominal snapshot. Standard values are used for all closure related parameters ($\mathrm{Sc}_T^{\mathrm{RANS}} = 1$, $C_{\mu P} = 4/15$, $C_{\mu\epsilon} = 2.22$, and $\sigma_\nu = 3$ in Yoshizawa transport equation of Eq. V.17). The comparison of the vertical turbulent tracer flux is shown in Fig. V.1.

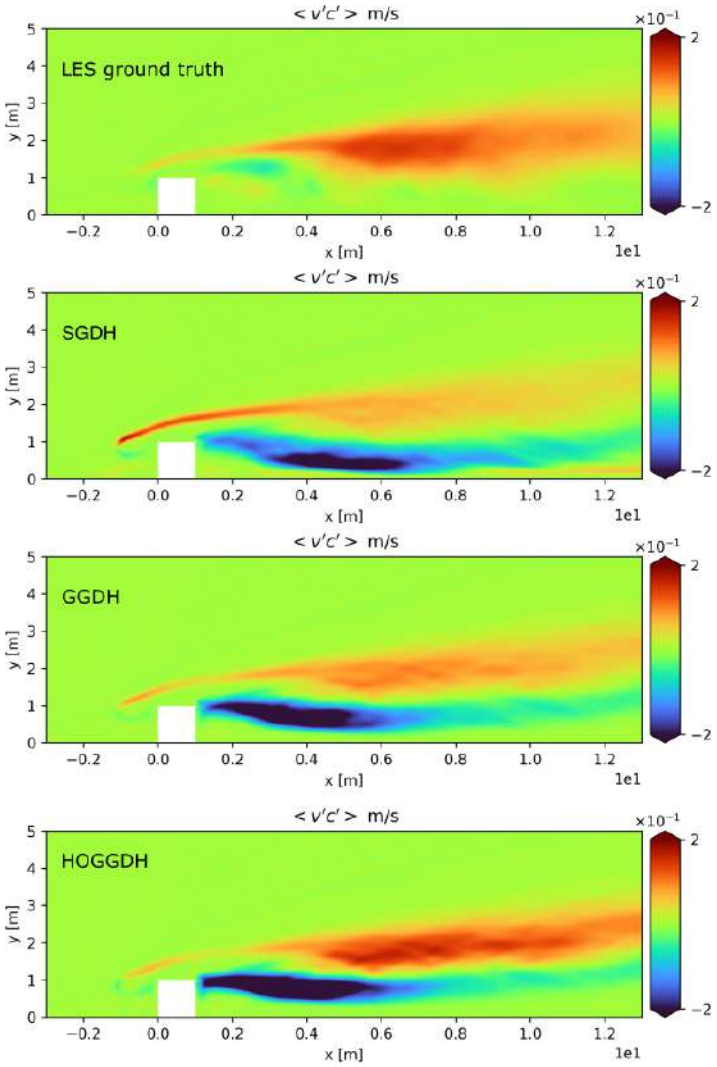

Figure V.1: Spatial fields of vertical turbulent tracer flux $\overline{\mathbf{K}'v'}$ for (a) ground truth data from LES, (b) SGDH closure, (c) GGDH closure, and (d) HOGGDH closure. The closure term is obtained using the *a priori* concentration field $\overline{\mathbf{K}}$ from LES.

The solution obtained with the LES ground truth data indicates that a strong upward transport by turbulence occurs in the upper part of the plume in the wake of the obstacle. For the three approximate closures, the same feature is observed and the vertical flux magnitude is rather well predicted with the standard closure coefficients. A slight improvement in the mag-

nitude prediction in the downstream region is observed for the HOGGDH closure. However, a significant departure is observed in the recirculating flow area behind the obstacle: a significant downward turbulent tracer flux is predicted by the three closures, while LES predicts weak vertical flux in this area. As for the horizontal fluxes shown in Fig. V.2, a misprediction is obtained for all three models compared to the ground truth data. Similar deficiencies of RANS modeling approach for horizontal flux prediction in the wake of obstacles were reported numerically by Gamel [2015] and experimentally by Vinçont et al. [2000], with significant counter-gradient diffusion that is not well represented by gradient-based closures. This could also be an artefact of the two-dimensional setup, which may not be representative of the three-dimensional turbulent transport process here. Still, this local misprediction can have a limited impact if the transport is dominated locally by the mean flow advection.

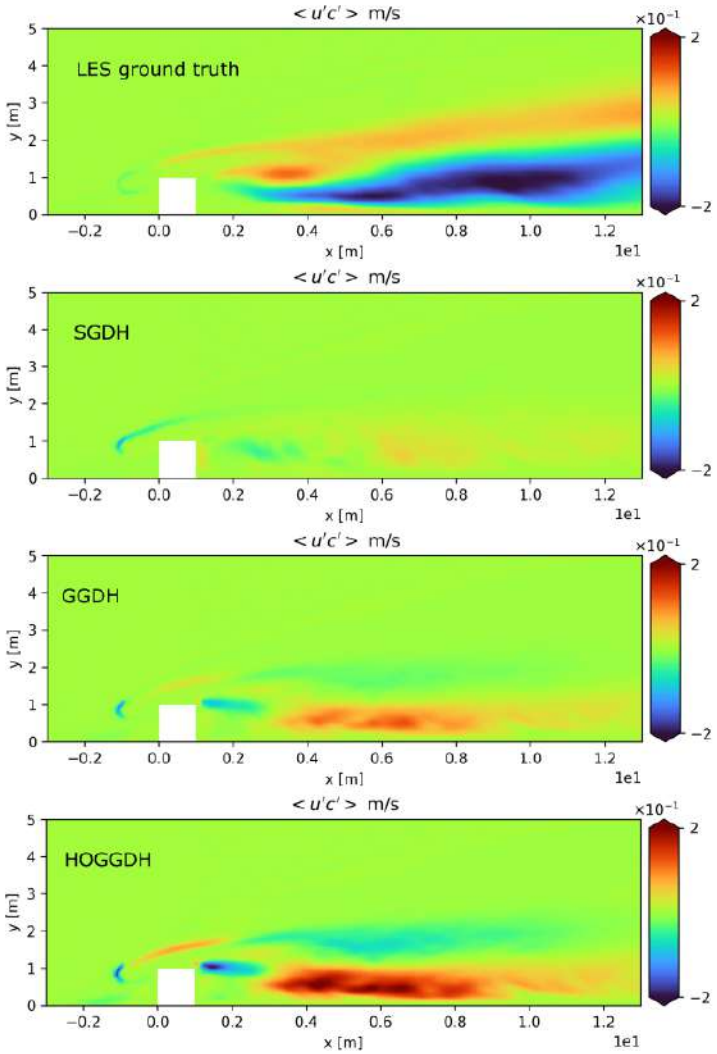

Figure V.2: Same caption as in Fig. V.1 but for the horizontal turbulent tracer flux $\overline{\mathbf{K}'u'}$.

A posteriori validation. To evaluate how the discrepancies in the prediction of the turbulent tracer flux translate on the prediction of the tracer field, an *a posteriori* evaluation is performed: the set of two transport equations for the tracer and the turbulent eddy-viscosity are solved. This corresponds to the LES-RANS-TE framework in which the LES terms are directly injected

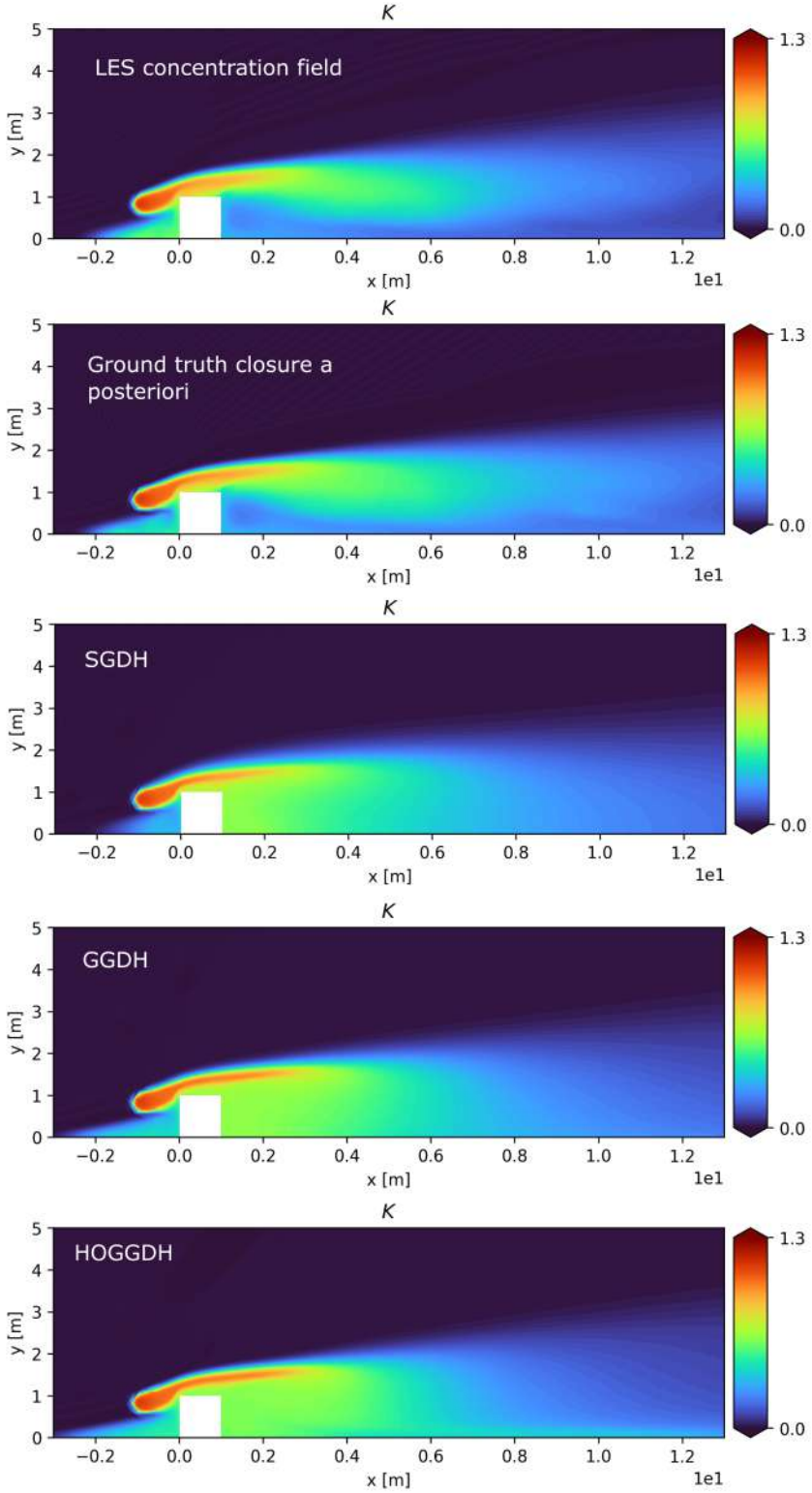

Figure V.3: Nominal snapshot of mean tracer concentration. Comparison between direct LES prediction and LES-RANS-TE prediction obtained with different closures: ground truth LES (exact closure), SGDH, GGDH and HOGGDH approximate closures.

in the RANS transport equations. The comparison of the resulting mean tracer concentration field in Fig. V.3 shows that the ground truth closure solution is very close to the original LES concentration field. This validates the time-averaged formalism introduced previously. This also confirms that the accuracy of the method lies in the choice of the tracer flux closure.

The predictions of the mean concentration field obtained with the three approximate closures (SGDH, GGDH, and HOGGDH) look qualitatively similar to the LES reference solution. Still, a significant deviation from the LES reference solution is observed near the ground and the leeward wall of the obstacle. This is attributed to the overly strong downward flux in these regions for the approximate closure solutions compared to the LES reference solution, as evidenced in the *a priori* analysis of the turbulent tracer flux.

To conclude, this analysis highlights the potential of the hybrid approach, which can recover the LES solution almost exactly if combined with an accurate turbulent transport closure, as illustrated on the nominal snapshot. By construction of the framework, which is agnostic to tracer dispersion in the training phase, the accuracy of the method is limited by the turbulent tracer flux closure required in the Reynolds average formalism: severe modelling deficiencies are identified for the flux prediction, which directly translate in terms of mean tracer concentration. Acknowledging this lack of precision, the framework might still be relevant in a low-order modelling context, as it may require a significantly cheaper LES database compared to the prediction framework presented in Chapter IV. A full assessment of the method is carried out in the following, for which the SGDH model is retained as *i*) it does not require to evaluate the Reynolds-stress tensor, which simplifies the emulation process; and *ii*) more advanced models based on tensorial diffusivity (GGDH and HOGGDH) do not provide a significant gain in accuracy in the present case study.

V.1.6 Summary of the hybrid approach

The hybrid model relies on a two-equation system (turbulent eddy-viscosity and mean tracer concentration) that requires information on the mean flow components ($\overline{\boldsymbol{u}}^\star, \overline{\boldsymbol{v}}^\star$), the turbulent kinetic energy $\boldsymbol{k}^\star_{tke}$ and the turbulent flow time-scale $\tau^\star_T$. The turbulent viscosity is computed based on the transport equation from Yoshizawa et al. [2012], and the SGDH model is used to close the turbulent tracer flux. The final formulation of the hybrid model is summarised below.

Formulation of the LES-/EMUL-RANS-TE framework:

$$\overline{u_j}^\star \frac{\partial \overline{\mathbf{K}}}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\left(\frac{\mu}{\mathrm{Sc}} + \frac{\nu_T^{\mathrm{RANS}}}{\mathrm{Sc}_T^{\mathrm{RANS}}}\right)\frac{\partial \overline{\mathbf{K}}}{\partial x_j}\right) + R\,, \tag{V.21}$$

$$\overline{u_j}^\star \frac{\partial \nu_T^{\mathrm{RANS}}}{\partial x_j} = C_{\mu P}\,\boldsymbol{k}^\star_{\mathrm{tke}} - C_{\mu\epsilon}\,\frac{1}{\tau^\star_T}\,\nu_T^{\mathrm{RANS}} + \frac{\partial}{\partial x_j}\left(\left(\nu + \frac{\nu_T^{\mathrm{RANS}}}{\sigma_\nu}\right)\frac{\partial \nu_T^{\mathrm{RANS}}}{\partial x_j}\right)\,, \tag{V.22}$$

with the baseline constants from the litterature:

$$\mathrm{Sc}_T^{\mathrm{RANS}} = 1.0\,, C_{\mu P} = 4/15\,, C_{\mu\epsilon} = C_{\mu P}/C_\mu = 2.22\,, \sigma_\nu = 3\,. \tag{V.23}$$

The quantities to be emulated in the EMUL-RANS-TE framework (or directly extracted from the LES solutions for the LES-RANS-TE framework) are the mean flow field components $\overline{u_i}^\star$, the turbulent kinetic energy $k^\star_{tke}$, and the turbulent flow time-scale $\tau^\star_T$. The resulting workflow shown in Fig. V.5 involves the use of four data-driven reduced-order models to emulate the LES fields injected in the RANS transport equations over the range of variation of the atmospheric uncertainties (U_{z_c}, z_0).

To isolate the error related to the emulation from the modeling error in the transport equation of the EMUL-RANS-TE approach, it is of interest to bypass the use of the emulated quantities and to directly use the LES data. This results in the LES-RANS-TE approach (schematically described in Fig. V.5) which can be compared to the EMUL-RANS-TE predictions to isolate the impact of the emulation step on the prediction. The LES-RANS-TE approach servers solely for the validation of the method. From a practical point of view it does not bring any cost reduction since a new integration of the LES model must be performed to query any new snapshot.

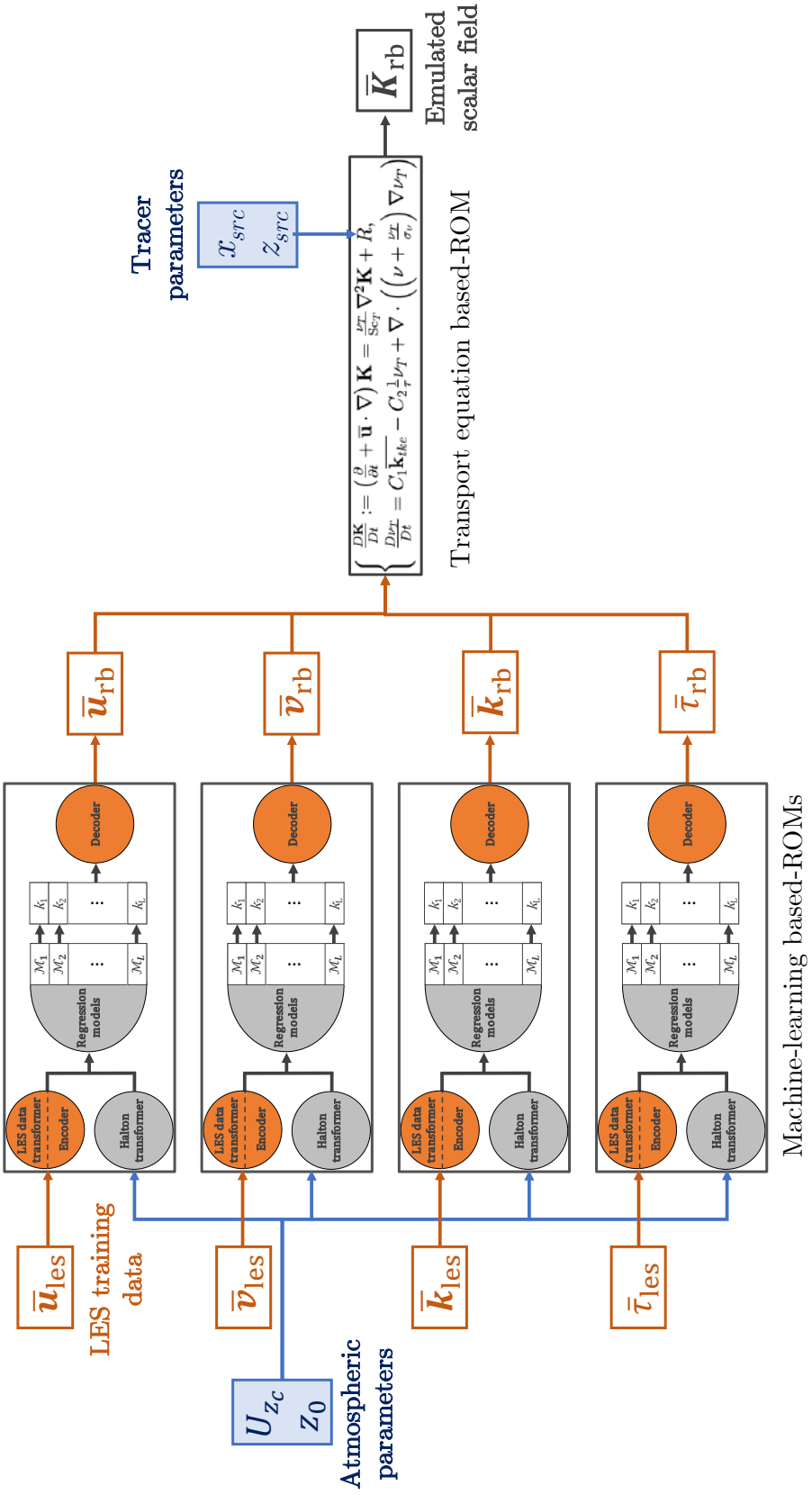

Figure V.4: Schematic of the EMUL-RANS-TE approach based on the emulation of LES fields injected in the RANS transport equations. Atmospheric uncertainties are handled by four machine-learning-based reduced-order models (Fig. II.8), which emulate the LES airflow quantity fields of interest from the atmospheric uncertainties (U_{z_c}, z_0). The two-equation system is solved using emulated fields ($\star$ superscript) and a given emission source position ($x_{\mathrm{src}}, z_{\mathrm{src}}$) that acts on the source term R.

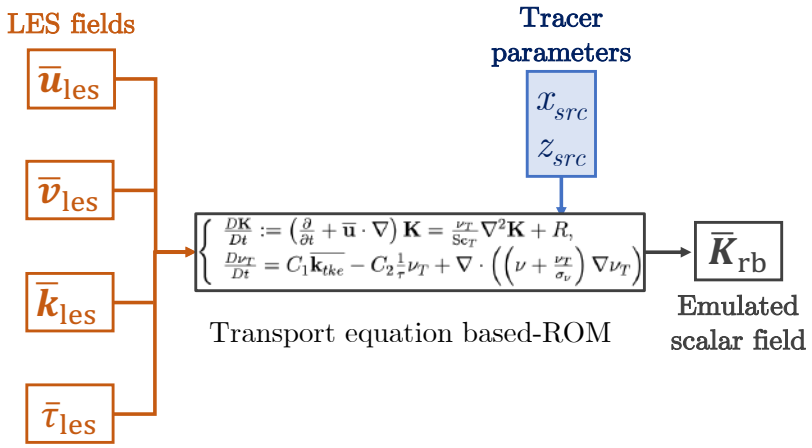

Figure V.5: Schematic of the LES-RANS-TE approach using ground truth LES fields (not using the data-driven reduced-order models).

V.2 Performance evaluation of the hybrid approach

In this section, the objective is to evaluate the prediction performance and the cost of the training phase (the required size of the LES ensemble) of the EMUL-RANS-TE hybrid approach developed in Sect. V.1 for our two-dimensional test case. The same four uncertain parameters are retained: two atmospheric parameters, i.e. the inlet boundary horizontal wind magnitude U_{z_c} at height $z_c = 10H$ and the roughness length z_0, as well as the two parameters on tracer emission source location $(x_{\text{src}}, z_{\text{src}})$. We evaluate the prediction performance of the EMUL-RANS-TE framework using the same performance metrics as for the direct prediction framework of Chapter IV.

V.2.1 Performance evaluation of the airflow reduced-order models

V.2.1.a Reduced-order model configuration

To build the EMUL-RANS-TE framework, we design four machine-learning-based reduced-order models to map at low-cost the high-fidelity LES fields from the atmospheric uncertainties (U_{z_c}, z_0), including the mean velocity components $(\overline{\boldsymbol{u}}, \overline{\boldsymbol{v}})$, the turbulent kinetic energy $\boldsymbol{k}_{\text{tke}}$, and the turbulent flow time-scale τ_T.

To emulate these LES quantities, we rely on the optimal architecture deduced from the comparison carried out in Sect. IV.3, based on POD for dimension reduction and adaptive Gaussian process regression (with Matérn 5/2 kernel) for reduced-coefficient metamodelling. Each reduced-order model is built independently, thus the number of POD modes to be retained may differ for the different quantities of interest depending on the complexity of their spatial features. For the specific case of turbulent kinetic energy and turbulent flow time-scale, a logarithmic transformation is applied before applying POD to account for the strong disparity of scales.

For the direct prediction framework of Chapter IV, a strong sensitivity to the size of the training database N_{train} was shown, which is also the main driver of the computational cost of the training phase. A strong degradation of the prediction was observed for $N_{\text{train}} = 50$ (see

Fig. IV.22). This same size is retained here, with the objective to evaluate if the EMUL-RANS-TE framework can maintain good prediction performance for the mean tracer concentration when the training dataset is very limited.

V.2.1.b Reduced-basis truncation

As in Chapter IV (Sect. IV.2), we analyse the impact of the POD basis truncation on the reconstructed quantities of interest on the test dataset (made of 150 LES snapshots).

- For the horizontal wind velocity component $\overline{\boldsymbol{u}}$, POD cumulative explained variance exceeds 99.9% when only retaining the first three modes. This corresponds to a global Q^2-criterion equal to 99.6% on the test dataset.

- For the vertical wind velocity component $\overline{\boldsymbol{v}}$, four POD modes are required to reach a cumulative explained variance of 99.6%. This corresponds to a global Q^2-criterion equal to 96.5%.

- For the turbulent kinetic energy $\boldsymbol{k}_{\text{tke}}$, optimal performance $Q^2 = 99.1\%$ is obtained when retaining the first five modes in the POD basis (this corresponds to a cumulative explained variance of 99.9%).

- Ten modes are required for the turbulent flow time-scale τ_T. to achieve a total explained variance of 98.2% and a global-Q^2 performance of 92.0%.

These results show that only a few POD modes are necessary for airflow quantities. Dimension reduction appears to be more challenging for the turbulent flow time-scale τ_T than for the other three quantities. Still, the number of POD modes is much smaller than for the tracer concentration quantity in Chapter IV for which 100 modes are required to explain 99.3% of the ensemble variance.

V.2.1.c Prediction performance

Figure V.6 shows the spatial Q^2 performance of the four data-driven reduced-order models. Almost all regions of interest are correctly emulated. Decreased performance is strongly correlated with low-variance areas. Indeed, Fig. V.6ad indicates weaker prediction performance downstream of the obstacle that are spatially correlated with low-$\overline{\boldsymbol{u}}$ areas. These areas indicate the boundary of the recirculation areas. These flow critical point vary from one snapshot to another but to a very limited extent (this is an area where the ensemble variance is very low). This implies that there is a drop in the prediction performance near these critical points. The same behaviour can be observed for other quantities. The vertical wind velocity component $\overline{\boldsymbol{v}}$ in Fig. V.6b obtains weaker performance in areas corresponding to the low-variance areas close to zero vertical velocity, located close to the obstacle and downstream at $x \approx 7.5\,H$. The turbulent time-scale (Fig. V.6d) is also poorly predicted in a band that corresponds to a shear-free region downstream of the obstacle, corresponding to large values of τ_T. As for the turbulent kinetic energy shown in Fig. V.6c, the reduced-order model performs well in all regions of the

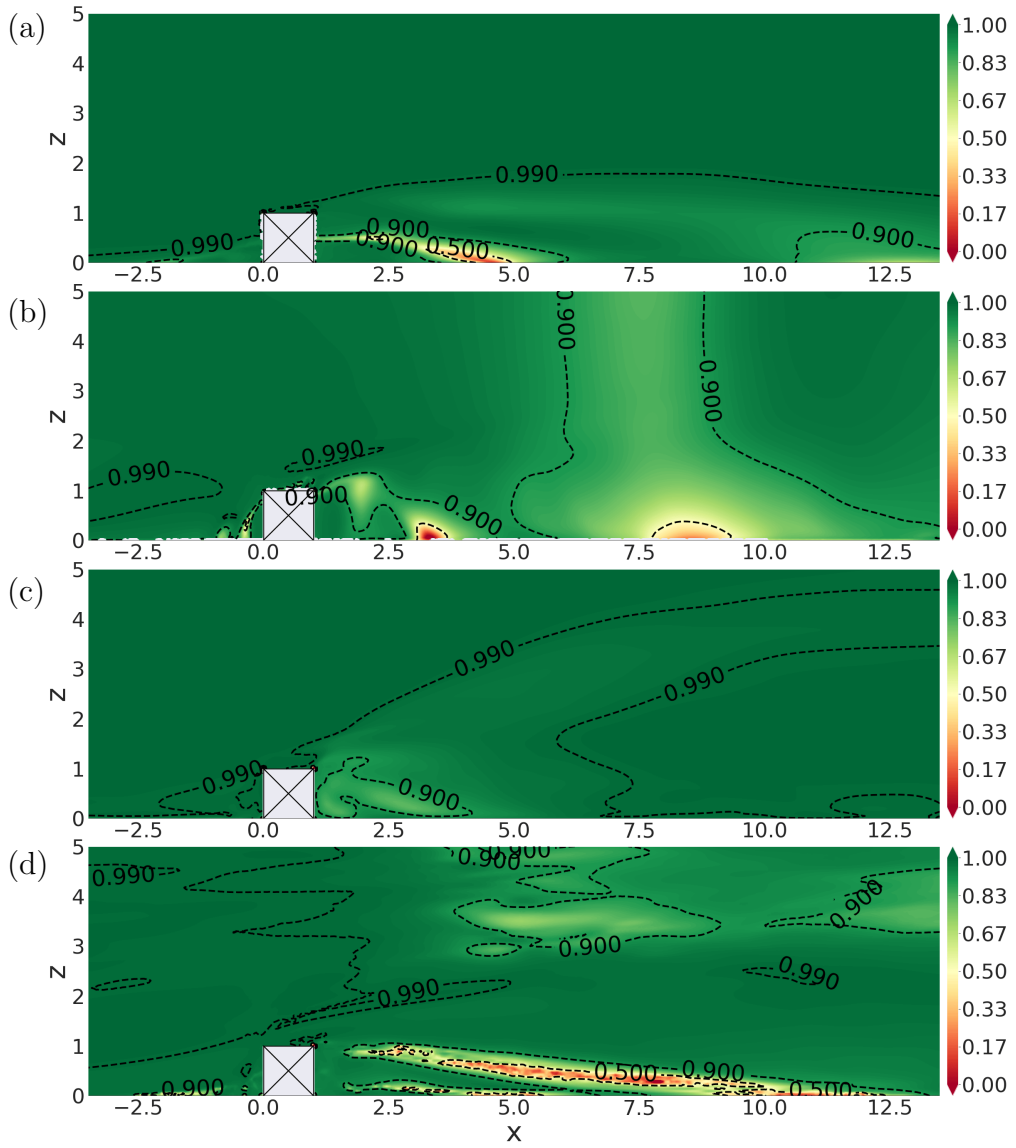

Figure V.6: Spatial fields of Q^2-criterion showing reduced-order model emulation performance for (a) horizontal mean flow $\overline{\boldsymbol{u}}$, (b) vertical mean flow $\overline{\boldsymbol{v}}$, (c) turbulent kinetic energy $\boldsymbol{k}_{\text{tke}}$, and (d) turbulent flow characteristic time-scale $\boldsymbol{\tau_T}$.

domain, except close to the leeward face of the obstacle where this a slight decrease in the Q^2-performance.

V.2.2 Prediction performance of the mean tracer concentration field

V.2.2.a Statistical analysis

Figure V.7 shows the Q^2-performance field of the EMUL-RANS-TE hybrid approach combining the airflow data-driven closure models (Sect. V.2.1) and the resolution of the RANS tracer transport equation. The predicted mean tracer concentration fields corresponding to the EMUL-RANS-TE solutions are directly compared to the LES test snapshots, resulting in a global Q^2-performance of 69.6%. We find that upstream of the obstacle and far above the ground, the tracer concentration is well recovered with high Q^2 values ($Q^2 > 90\%$). Moreover, the upstream accumulation region is well represented in the EMUL-RANS-TE solutions. This is due to the fact that most of the variance in this area is driven by snapshots whose emission source is

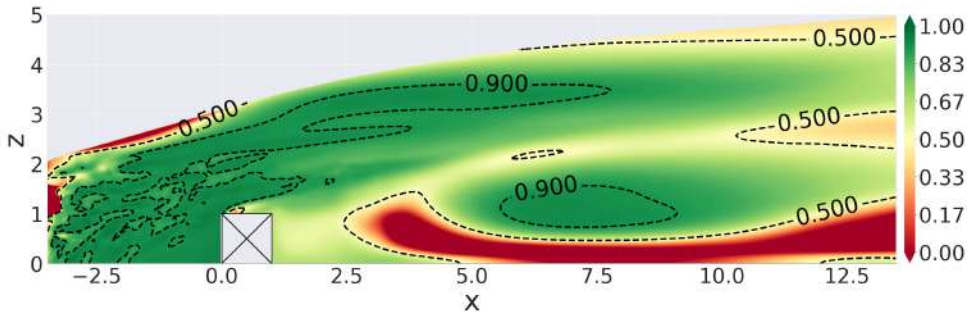

Figure V.7: Q^2 spatial performance of the mean tracer concentration field prediction over the test database obtained for the EMUL-RANS-TE hybrid approach when compared with the LES reference snapshots.

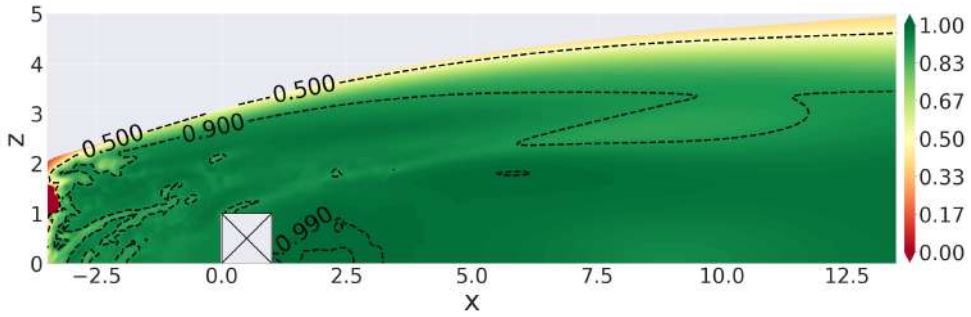

Figure V.8: Q^2 spatial performance of the mean tracer concentration field prediction over the test database obtained for the EMUL-RANS-TE hybrid approach when compared with the LES-RANS-TE solutions.

located in the accumulation region, and that the EMUL-RANS-TE approach well represents these snapshots. However, downstream of the obstacle, near the ground surface, there is a significant drop in Q^2-performance. This is consistent with the *a posteriori* analysis of the nominal snapshot (Section V.1.5, which showed the poor accuracy of the turbulent transport model in the wake region). This performance analysis on the whole ensemble confirms that the EMUL-RANS-TE approach performs well in the upstream region of the obstacle, and but the performance degrades significantly in the wake region, especially close to the ground.

In complement, Fig. V.8 compares the performance of EMUL-RANS-TE to the LES-RANS-TE solutions (instead of the LES reference solutions in Fig. V.7). Both approaches are found to be very close to each other. The LES-RANS-TE approach is associated with a global Q^2-performance of 71.7%, which is slightly higher than for the EMUL-RANS-TE approach (69.6%). This implies that injecting the emulated LES flow statistics in the RANS transport equation results in a low performance decrease compared to the ideal LES closure statistics.

To summarise, the decoupling strategy effectively deals with both atmospheric and tracer uncertainties. However, in its current state, the transport equation model is not fully suited for modelling tracer dispersion in recirculation areas and the EMUL-RANS-TE framework would greatly benefit from better modelling assumptions in the transport equations.

V.2.2.b Snapshot prediction example

The EMUL-RANS-TE approach is further evaluated through the example of the nominal snapshot (which is part of the LES test database).

Figure V.9 shows the tracer concentration prediction for the LES-RANS-TE approach (exact LES data in the RANS transport equation) and compares it to the LES solution in the nominal case. It corresponds to the *a posteriori* validation case already shown in Sect. V.1.5, with a strong tracer concentration misprediction close to the ground: some tracer quantity is trapped in the leeward face recirculation zone for LES-RANS-TE, whereas there is no tracer accumulation in this zone for the LES data.

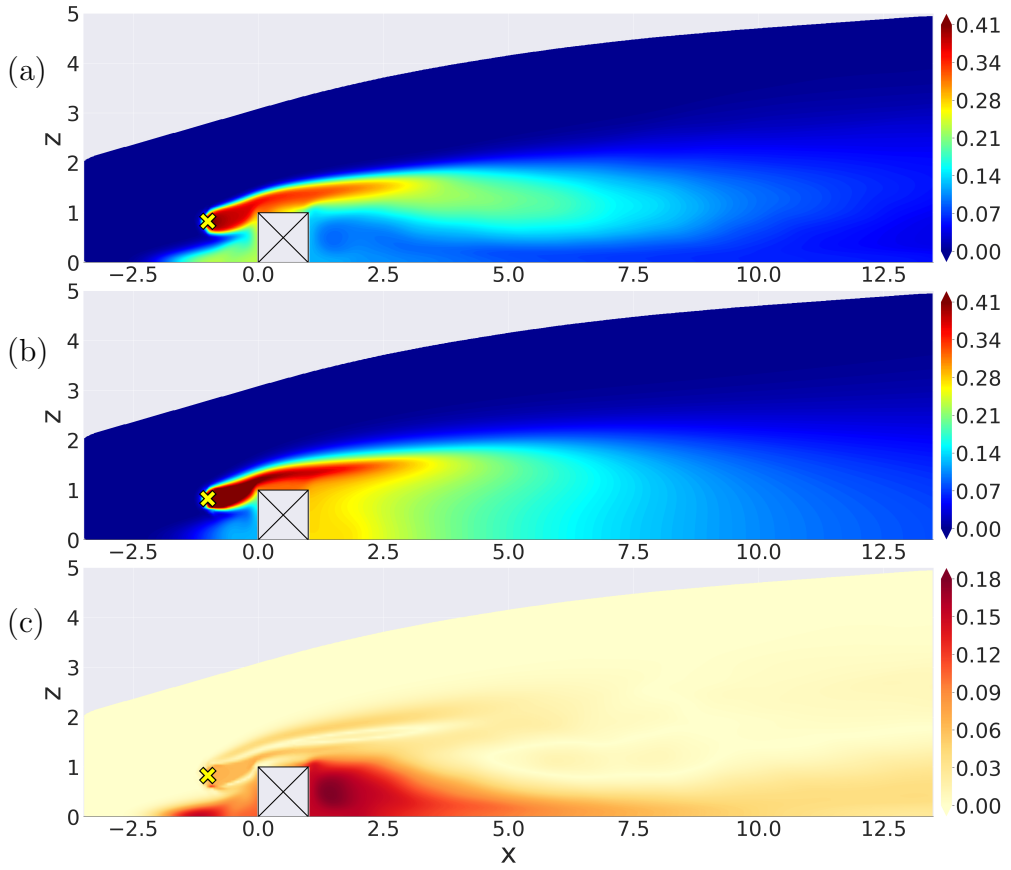

Figure V.9: Nominal snapshot mean normalised tracer concentration field obtained with: (a) LES solution and (b) LES-RANS-TE prediction. (c) Prediction absolute error measuring the discrepancy between the LES solution and the LES-RANS-TE prediction.

In a second step, the impact of the LES data emulation for the EMUL-RANS-TE approach is examined in Fig. V.10. The predicted tracer field with airflow statistics emulation (Fig. V.10a) is very close the field obtained without emulation (Fig.V.9b). This highlights that the emulation step has no significant impact on the prediction performance, consistently with the previous statistical analysis in Sect. V.2.2. This is confirmed by the field of absolute error between EMUL-RANS-TE and LES-RANS-TE (Fig. V.10b), which shows only minor differences for the tracer concentration in the near-obstacle region. Overall, this confirms that the weak point of the EMUL-RANS-TE hybrid framework is the turbulent tracer flux closure, which primarily leads to a tracer concentration overprediction in the recirculation region downstream of the obstacle.

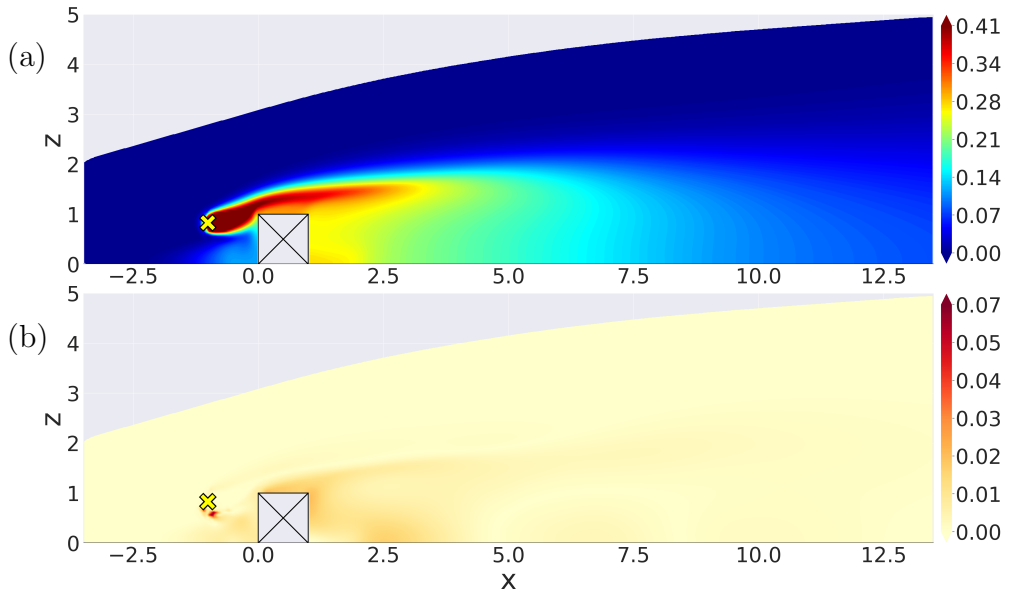

Figure V.10: Nominal snapshot mean normalised tracer concentration field obtained with (a) the EMUL-RANS-TE solution. (b) Prediction absolute error measuring the discrepancy between the EMUL-RANS-TE solution and the LES-RANS-TE solution (Fig. V.9b).

To conclude this section, the LES-informed hybrid approach (EMUL-RANS-TE) has been designed to decouple atmospheric uncertainties from tracer location uncertainties, since emulating the LES response to variations in the tracer location was found to require a very large training dataset in Chapter IV (at least 100 LES snapshots), which induces to significant computational cost for the training phase. The hybrid approach has the advantage of being efficiently trained using a small number of LES snapshots (50), while maintaining field physical consistency. The use of the RANS scalar transport equation provides physical constraints, which can absorb noise arising for the emulation process, and limits the occurrence of prediction artefacts. In particular, the hybrid approach limits noise on the tracer prediction in the advection dominated upstream region compared to Chapter IV. Still, the hybrid approach prediction performance is limited by the RANS closure of turbulent fluxes, which is well suited for shear flows but not fully adequate for a wake behind an obstacle. The two-dimensional nature of the turbulence in this setup may exaggerate this departure, as RANS closures are constructed and calibrated for three-dimensional turbulence. EMUL-RANS-TE prediction performance is therefore reduced in the wake of the obstacle, but is relatively good in regions dominated by mean advection (close to tracer source) and shear.

Moreover, the direct approach and EMUL-RANS-TE offer different offline and online cost. There is a factor of 10 on the size of the training dataset (500 for the direct approach versus 50 snapshots for EMUL-RANS-TE). Even if it could be optimised, integrating the RANS transport equation to produce new online predictions remains several orders of magnitude higher than a machine learning model such as Gaussian process regression (a few milliseconds for the direct approach versus a few CPU hours for EMUL-RANS-TE). The two approaches are therefore complementary and could yield a very efficient and accurate approach if effectively combined in a multi-fidelity framework, which is the object of the next section.

V.3 Towards a multi-fidelity reduced-order model

In Chapter IV, we investigated direct reduced-order modelling approaches from LES solutions for real-time assessment of tracer concentration. Since such direct approaches require large training data to achieve acceptable accuracy, the EMUL-RANS-TE hybrid approach combining airflow data-driven reduced-order models and a RANS tracer transport equation may be used as a cheaper training data generator than the direct LES model. This idea is explored in this section through two approaches, a reduced-order modelling approach based exclusively on EMUL-RANS-TE solutions on the one hand (Sect. V.3.1), and a multi-fidelity reduced-order modelling approach combining EMUL-RANS-TE and LES solutions on the other hand (Sect. V.3.2).

V.3.1 Emulation of the hybrid approach from low-fidelity solutions

In this section, we investigate the prediction performance of a reduced-order model trained from a database of EMUL-RANS-TE solutions, considered as low-fidelity, by contrast with the LES data which is considered as "high-fidelity". This reduced-order model is referred to as **DIRECT-ROM-LF** (for direct prediction with reduced-order model based on low-fidelity solutions), and provides prediction of the mean tracer concentration field when varying the four uncertain parameters $\boldsymbol{\mu} = (U_{z_c}, z_0, x_{\text{src}}, z_{\text{src}})$, as in Chapter IV. The main expected benefit is a faster inference time compared to the direct LES model or to the LES-informed RANS tracer transport equation (EMUL-RANS-TE). This is attractive for two main reasons:

- Computational time is saved during the offline training phase: less full-order LES snapshots (50 in this case) are required to construct the EMUL-RANS-TE hybrid model and a very large ensemble of EMUL-RANS-TE snapshots is affordable because of its reduced computational cost compared to LES.
- The resulting data-driven model is very efficient to evaluate for any new set of parameters in the online phase (compared to solving the RANS tracer transport equation for any new source location).

The DIRECT-ROM-LF reduced-order model is built using POD and adaptive Gaussian process regression. We use a training database of 450 EMUL-RANS-TE snapshots, themselves derived from 50 full-order LES solutions to emulate the airflow field statistics (as previously explained in Sect. V.2). We then evaluate the prediction performance of the resulting reduced-order model against reference LES solutions. The performance of DIRECT-ROM-LF is illustrated in Fig. V.11 through the example of the nominal snapshot. It should be noted that in the following, the reduced-order modelling approach developed in Chapter IV is referred to as DIRECT-ROM-HF (for direct prediction with reduced-order mode from LES high-fidelity data by opposition to DIRECT-ROM-LF based on lower fidelity data).

Figure V.11 shows the predicted field from the DIRECT-ROM-LF reduced-order model, and compares it to the reference LES solution. We observe that the reduced-order model prediction suffers from the weakness of both the EMUL-RANS-TE approach and the DIRECT-ROM-HF

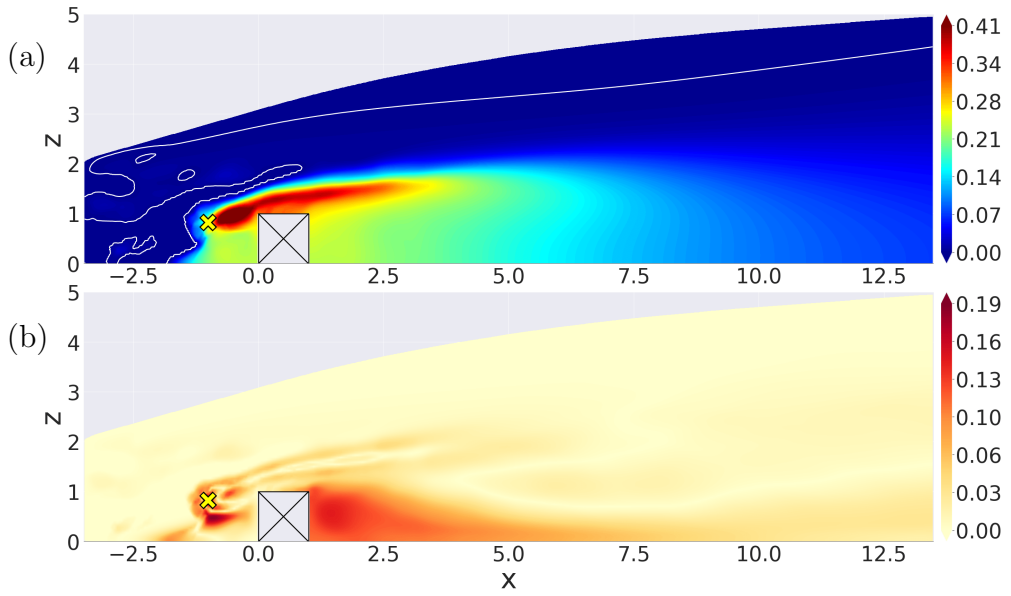

Figure V.11: Prediction of the mean normalised tracer concentration field from the DIRECT-ROM-LF reduced-order model (trained on 450 EMUL-RANS-TE snapshots). (a) Reduced-order model prediction. (b) Prediction absolute error computed with respect to the reference LES snapshot (Fig. V.9b).

approach developed in Chapter IV. On the one hand, the wake close to the emission source is poorly reconstructed with the same numerical artefacts already observed in Chapter IV. On the other hand, the prediction in the recirculation region downstream of the obstacle suffers from the same biased pattern as the EMUL-RANS-TE solutions, with overly strong accumulation of tracer concentration in this region. The resulting global performance compared to the reference LES solution drops to $Q^2_{\text{global}} = 64.8\%$. For comparison, the DIRECT-ROM-HF reduced-order model trained on 50 LES snapshots (in Chapter IV) gives a performance of 83.1% (see Table IV.5). The weak performance of the DIRECT-ROM-LF reduced-order model is clearly explained because of the biased training snapshots due to imperfect turbulent closure.

From this first result one may conclude that it is better to train the data-driven reduced-order model directly from the 50 LES snapshots rather than going through the DIRECT-ROM-LF approach (which requires the integration of 450 EMUL-RANS-TE solutions and 50 LES snapshots for emulating the airflow field statistics).

V.3.2 Multi-fidelity emulation approach

To solve this issue, we investigate if mixing LES solutions and EMUL-RANS-TE solutions can provide a way to obtain a more robust and more efficient data-driven reduced-order model. This mix of solutions of different fidelity levels is referred to as multi-fidelity [Goodfellow et al., 2016]. The key idea is to take advantage of the benefits of the different types of solutions to have a more physically-consistent prediction of the mean tracer concentration field. In particular, multi-fidelity allows to generate a larger training database at a reduced computational cost (compared to the DIRECT-ROM-HF approach presented in Chapter IV). The resulting multi-fidelity reduced-order model is referred to as DIRECT-ROM-MF in the following.

In practice, in the present framework, the available training database is made of 50 LES snapshots and 450 EMUL-RANS-TE solutions (that are considered as biased data, see Sect. V.3.1).

Thus, the training database now consists of multi-fidelity data composed of high-fidelity LES and biased lower-fidelity EMUL-RANS-TE solutions. It is worth noting that the same 50 LES snapshots are also used to emulate the airflow field statistics for the EMUL-RANS-TE approach.

V.3.2.a Introduction to multi-fidelity methods

The quality of the reduced-order model prediction output heavily relies on the training data. When the amount of data is excessively limited (as for expensive solvers such as LES), or when data are of poor quality (as for low-fidelity models such as the EMUL-RANS-TE hybrid approach), fully data-driven reduce-order models perform poorly. The aim of this section is to investigate how to jointly use LES and EMUL-RANS-TE solutions to generate a satisfactory training dataset.

Statistical approaches presented in Chapter II stand as a particular class of data-driven techniques suited for single-level data. Here we focus on several extensions to multilevel response models, which feature a similar structure to the two-step approach introduced in Fig. II.8 and developed in Chapter IV: in a first step, dimension reduction is carried out to compress the high-dimensional mean tracer concentration fields $\mathbf{K}_{\text{les}}(\boldsymbol{\mu})$ and $\mathbf{K}_{\text{emul}}(\boldsymbol{\mu})$ obtained from LES and EMUL-RANS-TE, respectively; and in a second step, the uncertainty parameters $\boldsymbol{\mu}$ are mapped onto the resulting compressed coefficients $\boldsymbol{k}_{\text{les}}(\boldsymbol{\mu})$ and $\boldsymbol{k}_{\text{emul}}(\boldsymbol{\mu})$ in the latent space using a regression model.

Multi-fidelity should be integrated at two stages, in the dimension reduction component and in the latent-variable emulation component. Basic machine learning tools such as POD or simple Gaussian processes cannot handle multi-fidelity data. A possible alternative is to use convolutional autoencoders and more advanced Gaussian processes that are well suited to the multi-fidelity setting, with autoencoders based on transfer learning, and Gaussian processes based on co-kriging plus autoregressive models. These techniques are briefly introduced in the following [Le Gratiet, 2013; Brevault et al., 2020].

Transfer learning for convolutional autoencoders. For dimension reduction, we implement a convolutional autoencoder using transfer learning [Goodfellow et al., 2016]. In our case, transfer learning simply defines the network training process. Once the network architecture has been set up, the network weights are trained with a two-step learning approach. In a first step, the network is trained on the large dataset of low-fidelity EMUL-RANS-TE solutions $\mathbf{K}_{\text{emul}}$ (the large dataset ensures loss function convergence). The autoencoder is then suitable for EMUL-RANS-TE data compression, but not for LES data compression. Stated differently, both the encoder, decoder and latent space components are optimal for $\mathbf{K}_{\text{emul}}$ but not yet for $\mathbf{K}_{\text{les}}$. The second learning step consists of restarting weight training from the pre-trained solution obtained at the end of the first step. The new descent procedure is carried out until loss function convergence is reached again. At the end of this second step, we obtain the final network weights that are used to derive the latent variables $\boldsymbol{k}_{\text{emul}}$ and $\boldsymbol{k}_{\text{les}}$ (from the high-dimensional statistics $\mathbf{K}_{\text{emul}}$ and $\mathbf{K}_{\text{les}}$, respectively). The resulting decoder will be used in the final multi-fidelity reduced-order model DIRECT-ROM-MF.

In this work, we use the same convolutional autoencoder architecture as in Chapter IV (Sect. IV.6.1). The size of the latent space is set to $L = 10$. Both gradient descent procedures (in the two learning steps) are identical and are based on the Adam descent scheme with an initial learning rate set to 10^{-3} and manually decreased to 10^{-4} and 10^{-6}.

Multi-fidelity Gaussian processes using co-kriging and autoregressive models. The Gaussian process regression framework is enhanced to model the mapping from the uncertain input parameters to the latent variables, i.e. $\boldsymbol{\mu} \mapsto \boldsymbol{k}_{\text{les}}$. We adopt the standard formulation introduced in Chapter II, meaning that L independent Gaussian process regression models are fitted to the L latent variables. However, what differs from Chapter II is that here we use a multilevel formulation for Gaussian processes to be built from co-kriging and autoregressive models [Kennedy and O'Hagan, 2000; Le Gratiet, 2013].

As for standard Gaussian process regression, we assume $\boldsymbol{k}_{l,\text{les}}(\boldsymbol{\mu})$ and $\boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu})$ to be realisations of two Gaussian processes (recall that the index l corresponds to the index of the latent variables, with l varying between 1 and L). In addition, co-kriging assumes that the joint process $(\boldsymbol{k}_{l,\text{les}}(\boldsymbol{\mu}), \boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu}))$ is also Gaussian given some parameters. The training of the multi-fidelity Gaussian process regression model is done through a two-step approach. In a first step, a standard Gaussian process regression model (as in Chapter IV) is trained to emulate the low-fidelity latent variables $\boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu})$. The second step is to emulate the high-fidelity latent variables $\boldsymbol{k}_{l,\text{les}}(\boldsymbol{\mu})$ by modelling a dependency on the low-fidelity latent variables $\boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu})$. To build the Gaussian process regression model for the most accurate data $\boldsymbol{k}_{l,\text{les}}(\boldsymbol{\mu})$, the objective is to determine the predictive conditional distribution of $\boldsymbol{k}_{l,\text{les}}(\boldsymbol{\mu}^*)$ for new inputs $\boldsymbol{\mu}^*$, given the observations $(\boldsymbol{k}_{l,\text{les}}(\boldsymbol{\mu}), \boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu}))$. We assume the Markov property:

$$\text{Cov}(\ \boldsymbol{k}_{l,\text{les}}(\boldsymbol{\mu}^*), \boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu}) \mid \boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu}^*)\ ) = 0, \qquad \forall\, \boldsymbol{\mu} \neq \boldsymbol{\mu}^*. \tag{V.24}$$

Equation (V.24) means that nothing can be learned about the value of $\boldsymbol{k}_{l,\text{les}}(\boldsymbol{\mu}^*)$ from any other solution of the EMUL-RANS-TE model than $\boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu}^*)$. From this Markov property and the Gaussianity assumptions, Le Gratiet [2013] models the high-fidelity response as the following autoregressive model:

$$\boldsymbol{k}_{l,\text{les}}(\boldsymbol{\mu}) = \rho_l(\boldsymbol{\mu})\, \boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu}) + \delta_l(\boldsymbol{\mu}), \tag{V.25}$$

where $\delta_l(\boldsymbol{\mu})$ and $\boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu})$ are assumed to be independent standard Gaussian processes (as introduced in Sect. II.3.3). Modelling choices must be made such as selecting an appropriate kernel (e.g. Matérn, RBF). Hyperparameters for $\delta_l(\boldsymbol{\mu})$ and $\boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu})$ are denoted by $\boldsymbol{\theta}_{l,\delta}$ and $\boldsymbol{\theta}_{l,\text{emul}}$, respectively. The additional Gaussian process $\rho_l(\boldsymbol{\mu})$ corresponds to a scale factor between $\boldsymbol{k}_{l,\text{les}}(\boldsymbol{\mu})$ and $\boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu})$ satisfying:

$$\rho_l(\boldsymbol{\mu}) = \frac{\text{Cov}(\ \boldsymbol{k}_{l,\text{les}}(\boldsymbol{\mu}), \boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu})\ )}{\mathbb{V}(\ \boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu})\ )}. \tag{V.26}$$

From such modelling assumptions, a Bayesian prediction model for the high-fidelity latent variables $\boldsymbol{k}_{l,\text{les}}(\boldsymbol{\mu}^*)$ can be derived from the posterior Gaussian distribution of $\boldsymbol{k}_{\text{les}}(\boldsymbol{\mu})$ given

the dataset collection of EMUL-RANS-TE and LES solutions $\mathcal{D}$, the scale factor ρ, and the hyperparameters $\boldsymbol{\theta}_{l,\text{emul}}$ and $\boldsymbol{\theta}_{l,\delta}$:

$$\boldsymbol{k}_{l,\text{les}}(\boldsymbol{\mu}^*) \mid \mathcal{D}, \rho_l, \boldsymbol{\theta}_{l,\text{emul}}, \boldsymbol{\theta}_{l,\delta} \sim \mathcal{N}(m_l(\boldsymbol{\mu}^*), r_l(\boldsymbol{\mu}, \boldsymbol{\mu}^*)), \tag{V.27}$$

with m_l and r_l the mean and covariance function of the Gaussian process. A closed form for the expression of $m_l(\boldsymbol{\mu}^*)$ is described in the work of Le Gratiet [2013], allowing for a fast numerical implementation of the emulation strategy.

In this work, since we choose to have a 10-latent variable convolutional autoencoder, we train $L = 10$ Gaussian process regression models and adopt the co-kriging approach for each regression model. For each regression model indexed by l, we rely on the autoregressive model (Eq. V.26) proposed by Le Gratiet [2013] to link the high-fidelity latent variable to the low-fidelity latent variable in the co-kriging approach. We assume $\delta_l(\boldsymbol{\mu})$ and $\boldsymbol{k}_{l,\text{emul}}(\boldsymbol{\mu})$ are modelled using noisy Gaussian processes based on the anisotropic Matérn kernel of type $\nu = 5/2$ following choices made in Chapter IV. The scale factor ρ is assumed to be constant, which is a usual assumption in practice (this constant value is calibrated during the training process). The Gaussian process hyperparameters are optimised using the standard MLL procedure (with 10 restarts).

Figure V.12 shows a schematic representation of the multi-fidelity approach from the perspective of response surfaces. The mapping to be reproduced is the relationship between the uncertain parameters $\boldsymbol{\mu}$ and the latent mode representation of the time-averaged tracer response. First, the multi-fidelity data (both the LES and EMUL-RANS-TE samples denoted by black points and green stars) is handled using transfer learning to learn a rather efficient low-dimensional representation. This step produces a first loss of information between the performed dimension reduction representation mode (denoted by dashed line – AE-CNN (MF) mode) and the theoretical optimal mode (denoted by the solid line – LES reference). The AE-CNN mode approximate is rather satisfactory considering a reduced number of LES samples and systematically wrong EMUL-RANS-TE samples. Note that the mode representation is just a mental picture and is not known in practice. Indeed, the interpolation model attempts to approximate it from a small number of samples. Using co-kriging to handle multi-fidelity data for interpolation, the final DIRECT-ROM-MF (dotted line) succeeds in approximating the AE-CNN mode (dotted and dashed lines are very similar).

In the end, we aim at designing a DIRECT-ROM-MF approach, whose response surface is the closest to the LES reference (i.e. the dotted line to be the closest to the solid line in Fig. V.12).

V.3.2.b Prediction performance evaluation

Dimension reduction. As a first step, we study how the mixture of multi-fidelity data can help to improve dimension reduction. In the present case, multi-fidelity data are complementary. Indeed, EMUL-RANS-TE solutions are well-adapted to model near-source patterns and upper region of the domain (Fig. V.7). These regions feature low-ensemble variance and are associated with high wavelength spatial structures, thus requiring a large training dataset to be correctly represented:

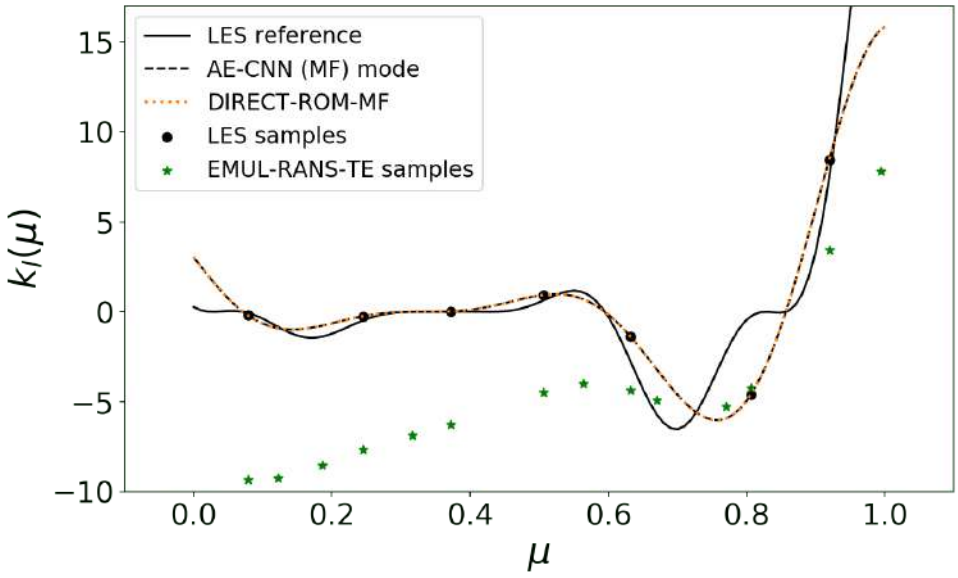

Figure V.12: Response surface schematic representation of the multi-fidelity approach using neural autoencoders along with transfer learning and co-kriging.

the lack of LES snapshots due to high computational cost may not allow satisfactory description of these low-variance areas. Since EMUL-RANS-TE data is abundant (low computational cost in the training phase) and well represents these regions, significant improvements are expected if this low-fidelity data can be exploited. For high-variance areas, a small number of simulations of high-fidelity is sufficient to ensure appropriate representation, and LES is well-adapted for this purpose (Sect. IV.5).

Figure V.13a shows the local Q^2 performance obtained when reconstructing the mean tracer concentration field using the convolutional autoencoder to evaluate the compression/decompression capacity of the autoencoder. This performance is evaluated directly against the LES test database. All regions in the domain of interest feature very high Q^2 values (except close to the upper boundary). This suggests that the autoencoder benefits from the LES information in the recirculation area and from the EMUL-RANS-TE information in the lower-ensemble variance areas. The global test performance of reconstruction is equal to $Q^2_{\text{global}} = 96.6\%$ (Table V.1). The multi-fidelity approach outperforms the 10-latent space convolutional autoencoder trained on only 50 LES snapshots that is included in the DIRECT-ROM-HF approach in Chapter IV (the global test performance is equal to $Q^2_{\text{global}} = 92.6\%$ in the DIRECT-ROM-HF case, see Table IV.4 in Sect. IV.6.2.a). It also outperforms the dimension reduction component of the DIRECT-ROM-LF approach (for which the global test performance could not go above $Q^2_{\text{global}} = 71.7\%$). This demonstrates that the complementary information coming from the LES and EMUL-RANS-TE snapshots improves the reconstruction capability of the convolutional autoencoder.

Latent variable emulation. Dimension reduction being improved through multi-fidelity, we train the $L = 10$ multi-fidelity Gaussian process regression models. All regression models achieve good performance: the minimum performance is obtained for the ninth latent variable with $Q^2 = 88.5\%$, while the maximum performance is obtained for the first latent variable with

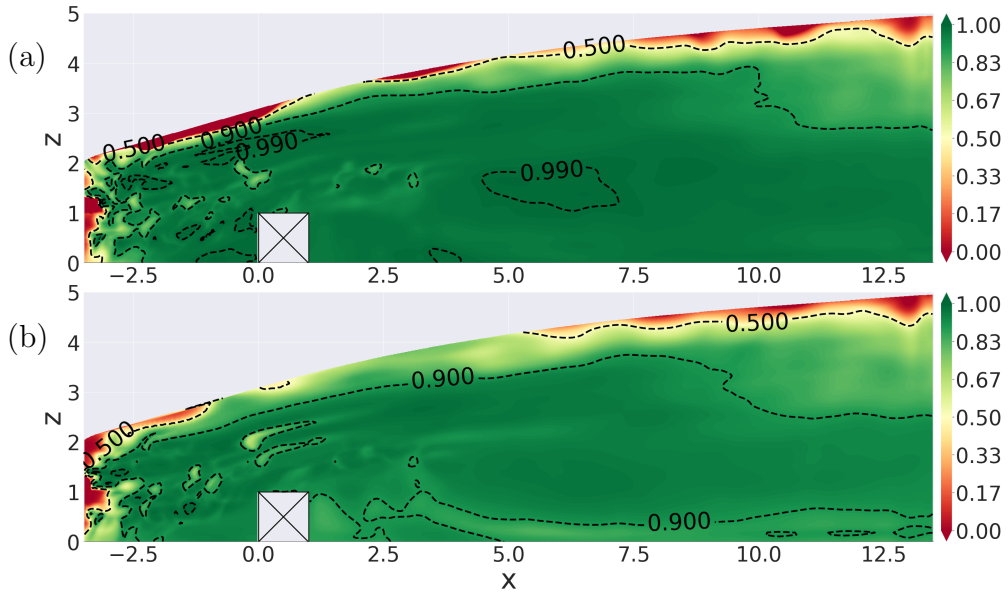

Figure V.13: Local Q^2 performance for (a) standalone reconstruction and (b) dimension reduction combined with Gaussian process regression based on co-kriging for the DIRECT-ROM-MF reduced-order model. DIRECT-ROM-MF is based on a 10-latent space convolutional autoencoder trained on 50 LES snapshots and 450 EMUL-RANS-TE snapshots. The Q^2-criterion is computed against the reference LES tests snapshots (recall that there are 150 snapshots in the test database).

$Q^2 = 97.5\%$. This leads to a global Q^2 test performance of 92.6% for DIRECT-ROM-MF (Table V.1). We observe in Table V.1 that the loss of accuracy between standalone reconstruction and DIRECT-ROM-MF prediction is more limited than in the DIRECT-ROM-HF approach trained on 50 LES snapshots only, implying that the Gaussian process regression models are successful in representing the variability of the latent variables with respect to the uncertain parameters $\boldsymbol{\mu}$. We also observe that multi-fidelity improves performance in all subregions of the domain compared to the DIRECT-ROM-HF model. These results are consistent with the spatial variability of the Q^2 test performance in Fig. V.13, where there is no significant difference between standalone reconstruction (Fig. V.13a) and DIRECT-ROM-MF prediction (Fig. V.13b).

It is worth noting that the performance of the DIRECT-ROM-MF reduced-order model is even relatively close to the DIRECT-ROM-HF approach trained on 100 LES snapshots (see Table IV.5). However, the latter is almost twice as expensive as the DIRECT-ROM-MF multi-fidelity approach (requiring the integration of 50 LES and 450 EMUL-RANS-TE solutions).

Table V.1: Multi-fidelity reduced-order model Q^2-scores (in %) obtained by comparison to the LES test database for standalone reconstruction and for the full DIRECT-ROM-MF approach emulation (including convolutional autoencoder and co-kriging). The latent space is of dimension $L = 10$. The number of LES snapshots used in the training step is 50. Note that the Q^2-scores for the DIRECT-ROM-HF approach based on the convolutional autoencoder (already presented in Table IV.4 for the reconstruction and in Table IV.5 for the full reduced-order model) are given in brackets for comparison.

	Q^2_{global}	Q^2_{T0}	Q^2_{T1}	Q^2_{T2}
Reconstruction	96.6 (92.6)	88.7 (72.6)	95.6 (91.2)	96.8 (91.4)
Prediction	92.6 (84.8)	84.6 (43.4)	91.4 (74.9)	92.7 (85.9)

Example of the nominal case. Figure V.14 shows the nominal prediction using the DIRECT-ROM-MF multi-fidelity approach along with the absolute error with respect to the reference LES snapshot. The coarse plume shape is very similar to the LES solution, especially downstream of the obstacle in the recirculation area, which does not show the defects of the EMUL-RANS-TE solutions (Fig. V.11). The recirculation area is also better reconstructed than in the DIRECT-ROM-HF solution (Fig. IV.26d), which is subject to tracer concentration underprediction in the recirculation area. The tracer concentration magnitude near the emission source is globally well reconstructed, without noisy structure in the upstream region. Nevertheless, some errors appear, mainly in the sharp tracer gradients in the accumulation area. Still, the DIRECT-ROM-HF solution (Fig. IV.26d) is also not perfect in the accumulation area. While the DIRECT-ROM-MF has a tendency to overpredict tracer concentration in the accumulation area, the DIRECT-ROM-HF solution has a tendency to underpredict it.

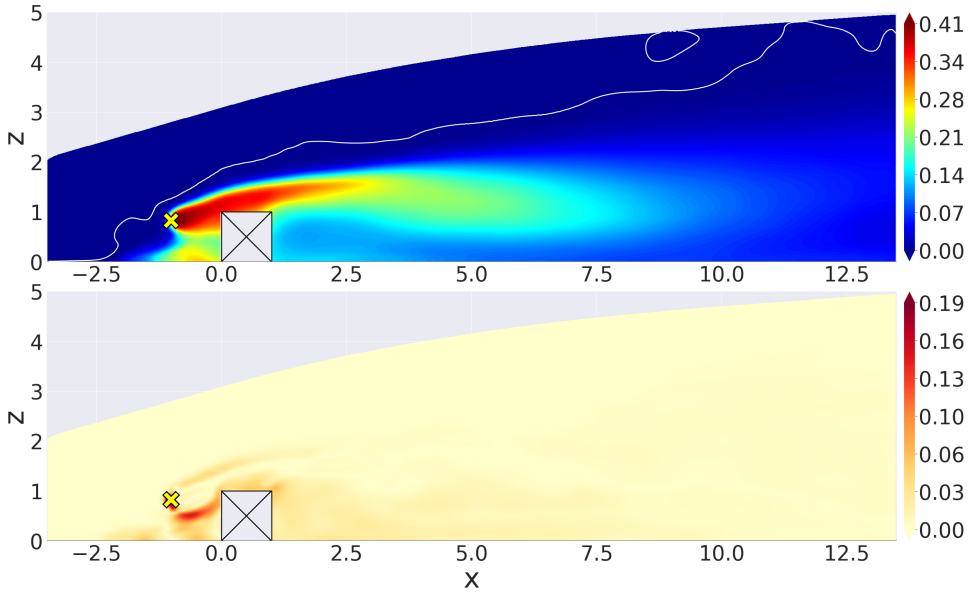

Figure V.14: Nominal snapshot mean normalised tracer concentration field obtained with (a) multi-fidelity ROM prediction. (b) Prediction absolute error calculated with respect to LES snapshot (Fig. III.8a). Contour line of the mean normalised tracer concentration equal to 5×10^{-4} is superimposed on reduced-order model predicted field to highlight the presence of low-magnitude noisy structures.

All these results indicate that the multi-fidelity procedure offers an interesting avenue of research to optimise the reduced-order model performance of tracer concentration real-time assessment from low-fidelity training data.

V.4 Conclusion

This chapter presents an alternative hybrid RANS/reduced-order modelling approach, which is based on the key idea of injecting detailed flow information from LES into a lower fidelity tracer transport equation in the RANS formalism.

The potential of the hybrid approach relies on the accuracy of the closure model to accurately represent the turbulence mass fluxes in the tracer transport equation. Several modelling techniques have been developed in the literature to address the representation of the flux tensor in complex urban flow dispersion applications. To identify the most appropriate closure model

for the flux tensor, we performed a comparison of SGDH, GGDH and HOGGDH closure models in the present case study (Sect V.1.5).
We demonstrated SGDH is the most appropriate closure model among the tested options as *i)* it does not require to evaluate the Reynolds-stress tensor, which simplifies the emulation process; and *ii)* more advanced models based on tensorial diffusivity (GGDH and HOGGDH) do not provide a significant gain in accuracy. For both closure models, the RANS transport equation performs well in the upstream region near the emission source (compared to the direct prediction approach in Chapter IV) but faces some difficulty in the wake of the obstacle with coarser dispersion structures when compared to the LES reference solutions.
Using SGDH, the LES-informed EMUL-RANS-TE hybrid approach aims at decoupling the atmospheric uncertainties from the tracer location uncertainties (Sect. V.2). The quantities to be emulated in the EMUL-RANS-TE framework are the mean flow field components $\boldsymbol{u}$ and $\boldsymbol{v}$, the turbulent kinetic energy $\boldsymbol{k}_{\text{tke}}$, and the turbulent flow time-scale τ_T. The resulting workflow involved the use of four data-driven reduced-order models to emulate the LES fields injected in the RANS transport equations over the range of variation of the atmospheric uncertain parameters.

Compared to the direct emulation of the mean tracer concentration of Chapter IV, the hybrid approach has the advantage of being efficiently trained using a small number of LES snapshots (50), without suffering from a significant loss of information about the physical structures. The use of the RANS tracer transport equation provides physical constraints to the emulation process, and limits the occurrence of numerical artefacts in the prediction of the mean tracer concentration field. In particular, the hybrid approach limits noise in the upstream region. The use of a reduced LES training database in the hybrid approach is also a very positive point since the number of required LES snapshots in the training database directly controls the computational cost of the offline training phase.

The hybrid approach main flaw comes from the RANS closure in the transport equation, which is well suited for shear flows but not fully adequate in the context of shedding behind an obstacle. The emulation performance of EMUL-RANS-TE is therefore reduced in the wake of the obstacle, but performs relatively well in regions dominated by mean advection (close to tracer emission source) and shear. Thus, the EMUL-RANS-TE framework would greatly benefit from better modelling assumptions in the transport equations. Acknowledging this lack of accuracy in the wake region, the hybrid framework might still be relevant in a low-order modelling context, as it requires a significantly cheaper LES database compared to the prediction framework in Chapter IV (at least 100 LES snapshots are required in the direct approach). In such context, the EMUL-RANS-TE approach may be used as a cheaper training data generator than LES. Both types of data can then be combined in a multi-fidelity reduced-order model framework. In this work, multi-fidelity was integrated for both dimension reduction and latent space emulation. We used convolutional autoencoders and co-kriging combined with autoregressive models to handle the multi-fidelity data collection of 450 EMUL-RANS-TE solutions and 50 full-order LES solutions. The multi-fidelity approach achieves improved accuracy compared to single-fidelity reduced-order models only using the 450 EMUL-RANS-TE solutions, or the 50 LES solutions.

This multi-fidelity approach opens up interesting prospects for taking advantage of high-fidelity LES solutions despite their high computational cost. These data can be used to improve the accuracy of multi-fidelity trained reduced-order models.

Conclusions and perspectives

The central idea of this PhD thesis work was to explore how machine and deep learning approaches can best be exploited to emulate accurate but expensive CFD modelling approaches to predict microscale urban flow and tracer dispersion in a parametric setting, i.e. when there are significant uncertainties in the inflow boundary conditions and in the emission source location. The focus was made on learning the spatial variability of field statistics of interest (mean tracer concentration, mean flow velocity components, turbulent kinetic energy, turbulent flow time-scale) from LES data and its dependency to uncertain inflow and source parameters. We carried out a detailed comparison of several reduced-order models on a two-dimensional canonical flow configuration with dispersion, with the idea of defining statistical learning guidelines to prepare for future large-scale dispersion applications.

Synthesis of main findings

To identify the most suitable emulation approach for the microscale dispersion context, a detailed comparison of dimension reduction approaches (POD, convolutional autoencoder) and regression models (polynomial chaos expansion, gradient tree boosting, Gaussian process regression) was carried out. This was made possible by studying a canonical two-dimensional case study corresponding to a turbulent atmospheric flow over an isolated surface-mounted obstacle. Despite its limitation on turbulence representation, this case study features complex flow patterns. Upstream of the obstacle, the plume dispersion is bi-modal: the tracer is either trapped in a first recirculation region on the windward face of the obstacle, or advected downstream of the obstacle. Downstream of the obstacle, the flow is driven by a combination of the quasi-periodic vortex shedding induced by the flow-obstacle interaction and the background turbulence propagating from the inlet. Plus, a reverse flow occurs near the ground, associated with a second recirculation region, transporting the tracer back towards the obstacle. Simulating such case using LES in a multi-query context is affordable in an offline way. Thus, a very large LES database made of 750 snapshots corresponding to different sets of uncertain input parameters (reference velocity magnitude, aerodynamic roughness length, emission source axial position and height) was generated, and includes a diversity of flow and tracer field topologies. This diversity makes the emulation process complex due to the nonlinear response of the field statistics to variations in the uncertain parameters. This issue becomes even more acute as the field statistics are of very high dimension (of the order of 10^5 grid points for this case study). To avoid introducing biases in the emulation process evaluation, the ensemble was split into training, calibration/validation and test subsets.

First main goal: Emulating the LES field statistics of interest using machine learning to design a data-driven reduced-order model

The ability to emulate mean tracer concentration fields from an ensemble of parameterised LES simulations using a purely data-driven reduced-order model was demonstrated in Chapter IV. The emulation process allows to accurately reproduce the LES predictions for a large training dataset (made of at least 100 LES snapshots), while reducing the computational cost to query a snapshot for a new set of parameters by several orders of magnitude.

– POD was used as the baseline approach for compressing the high-dimensional field statistics. A detailed examination of the POD modes highlighted the POD difficulties to deal with uncertain source position, which induces by definition spatial variability of the tracer concentration wake upstream of the obstacle. This close link between the uncertainty space and the space domain is the main reason for these difficulties, as it requires a very large number of modes (about 100 modes for training database of 450 LES snapshots) to represent the high spatial variability of the tracer concentration near the emission sources. The first low-order modes are sufficient to capture the tracer concentration spatial variability in areas where diffusion is significant, for instance in the recirculation area behind the obstacle. In the opposite, high-order modes are necessary to include in the POD basis to capture the fine plume structures in the near-source regions upstream of the obstacle, where advection dominates.

– The best metamodelling approach to represent the response of the POD reduced-coefficients to changes in the uncertain parameters was found to be Gaussian process regression, with a mode-per-mode optimisation of the hyperparameters to adapt to the wide range of spatial scales across the POD basis, leading to very good global Q^2-score (above 95%) for a large LES training database. This best performance of Gaussian processes over other regression models may be explained by the training data's low noise and the presence of strong non-linearities in the mapping between uncertain parameters and POD reduced coefficients. Interpolation models like Gaussian processes excel under these conditions.
  We further analysed the behaviour of the Gaussian process regression model through a mode-per-mode Q^2-score to understand the spatial variability of the prediction performance. We showed that mean tracer concentration in recirculation zones are well predicted, while there is a drop in performance near the emission sources. There, the mean tracer concentration response is carried by high-order POD modes. These high-order modes feature very localised structures associated with perturbed emission locations, which are difficult to predict and which are prone to more noise than low-order modes. We also showed that this difficulty of predicting high-order modes increases when the training dataset is reduced, but the prediction performance remains acceptable when considering at least 100 LES snapshots in the training database.
  In complement, the Gaussian process hyperparameter optimisation process done for each POD reduced coefficient was improved to gain computational time. We showed that a satisfactory prior distribution for the hyperparameters can be obtained from POD modes and can be used to inform a MAP optimisation procedure. MAP was found to provide similar results to a standard N-restart MLL maximisation approach but using a single gradient descent, providing a reliable efficient offline training framework.

– The ability of convolutional autoencoder to better compress the field statistics within the Gaussian process-based reduced-order model was studied to overcome POD limitations, starting from the work by Fukami et al. [2020]. We showed that for the full training database (made of 450 LES snapshots), it is possible to reduce the latent space size by a factor of 10 compared to POD (10 latent variables versus 100 POD modes) without losing information; there is even a slight improvement in prediction accuracy. We also highlighted that the autoencoder training becomes challenging when reducing the training database. The consistency of the results was not guaranteed, and the autoencoder behaviour was difficult to anticipate due to its black-box nature.

Second main goal: Informing the RANS scalar transport equation from LES airflow data to design a hybrid reduced-order model

The purely data-driven reduced-order model requires a large LES training database to achieve accurate prediction without numerical artefacts, which may be out of reach for practical three-dimensional realistic applications. The need for a large training dataset is mostly due to fine plume structures in the near-source regions caused by uncertainty in the emission source location. By reducing the number of training snapshots below 100, a loss of consistency with physics principles was observed: for instance, non-physical noisy structures appear in tracer-free regions, as the reduced-order model is built in a purely data-driven manner, which is not constrained by physics principles. To overcome these limitations, we designed an alternative hybrid RANS/reduced-order modelling approach (EMUL-RANS-TE) in Chapter V, which is based on the key idea of injecting detailed flow information from LES into a lower fidelity tracer transport equation in the RANS formalism.

– The proposed EMUL-RANS-TE reduced-order model relies on the emulation of airflow statistics (mean flow field components, turbulent kinetic energy, turbulent flow time-scale) from LES using a specific data-driven reduced-order model for each quantity. This reduced-order model combines POD and Gaussian process regression as in Chapter IV, with the difference that the number of POD modes to be retained in the reduced basis is now very small (less than or equal to 10) as the emulation problem is simpler with only the inflow boundary conditions as source of uncertainty. In a second step, the emulated LES statistics are integrated into the RANS scalar transport equation to close the Reynolds-averaged turbulent tracer flux. The hybrid model can then be run for varying emission source axial position and height to obtain mean tracer concentration field predictions. The main limitation is that the form of the closure is constrained by the RANS formalism, which has shown some discrepancies with respect to the LES reference solutions in the recirculation zone near the leeward wall of the obstacle.

– The ability of the EMUL-RANS-TE hybrid approach to emulate mean tracer concentration fields was demonstrated through a spatial analysis of the Q^2-score. We showed that EMUL-RANS-TE provides accurate prediction in regions dominated by mean advection

and shear near emission sources, and eliminates the artefacts observed in the purely data-driven reduced-order model (DIRECT-ROM-HF) predictions in Chapter IV. However, the performance of EMUL-RANS-TE is reduced in the downstream recirculation area due to the turbulent tracer flux closure deficiencies.

– With the development of the EMUL-RANS-TE hybrid approach, models with different levels of fidelity (LES versus hybrid LES/RANS approach) are available. They were mixed in a multi-fidelity reduced-order model framework (DIRECT-ROM-MF) to benefit from their complementarity, i.e. the accurate prediction of the LES data-driven model downstream and of EMUL-RANS-TE upstream of the obstacle. A second advantage is that a very large training database (made of 450 EMUL-RANS-TE snapshots and 50 LES snapshots) can be generated since integrating the EMUL-RANS-TE hybrid approach is much cheaper (by a factor of 10 to 100) than directly integrating a LES model. The multi-fidelity approach relies on a convolutional autoencoders trained through transfer learning for dimension reduction, and on Gaussian processes based on co-kriging for metamodelling. The multi-fidelity approach obtained very satisfactory reconstruction and prediction results, and even outperformed those achieved with the LES data-driven reduced-order model with an equal budget of LES simulations.

As a result of this PhD thesis work, multi-fidelity appears as a promising approach to take advantage of high-fidelity LES solutions in order to build an accurate and efficient reduced-order model for microscale dispersion.

Perspectives

Methodological perspectives

From a methodological standpoint, future work both includes *(i)* improving the design of reduced-order models in terms of dimension reduction and regression, and *(ii)* fully exploit the bayesian framework of Gaussian processes to enable accurate uncertainty analyses.

When dealing with dimension reduction, the explored algorithms minimised the mean-squared error (MSE) of the reconstruction with respect to the original fields (both for centred POD and autoencoders). We first applied a standard formulation of the centred POD, matching the diagonalisation of the covariance matrix and equivalent to the diagonalisation of the centred snapshot matrix, as the baseline algorithm for dimension reduction. This is equivalent as minimising the average MSE over all features (the mesh nodes). This formulation is suitable for modelling regions of high ensemble variance. However, it is not particularly adapted to modelling low concentration regions, often related to low ensemble variance regions. Moreover, this formulation does not incorporate any kind of physical constraints. An interesting perspective would be to tune the loss function to improve the properties of the reconstructed fields. For instance, altering the weights in the feature space for POD has not been investigated in this work. This can be done by reducing the snapshot matrix (i.e. unit variance features), the reduced and centred POD then matches the diagonalisation of the correlation matrix. Ruan et al.

[2006] showed that areas of low ensemble variance are better recovered with this formulation, but at the expense of regions of high ensemble variance. As for autoencoders, better tuning the loss function could improve robustness, physics consistency or interpretability. This is a very active area of research, where additional penalisation terms and constraints may be applied on the latent space, on the output fields, or on the network weights [Champion et al., 2019; Kingma et al., 2019]. For instance, Champion et al. [2019] added sparsity and regularity constraints to the loss function to promote interpretability and neural network generalisation. As dimension reduction algorithms are primarily used to produce a latent space representation that regression algorithms will leverage as output, statistical constraints can be applied on the latent space. For instance, variational autoencoders [Kingma et al., 2019] enforce a multi-dimensional Gaussian distribution on the latent space. From the regression algorithm point of view, constraints may also be added to improve physics-consistency and robustness, e.g. recent developments on constrained Gaussian regression processes [Swiler et al., 2020] may help better define the output manifold in the latent space.

The reduced-order models we used in this work involved dimension reduction and regression models that were trained independently. First, we designed one regression model per mode of the latent space in single-output prediction manner. Such methodology does not account for correlations between modes. In further work, added value of joint optimisation may be evaluated for multi-output regression models [Borchani et al., 2015]. It would also be of interest to combine dimension reduction and regression learning at the same time. This could be done by combining an autoencoder with a multilayer perceptron network. In the long run, it may be worthwhile to investigate if gradient propagation may enable integrated learning of Gaussian processes, decision trees, or other regression tools with autoencoders, although this will surely raise overfitting issues.

The classes of algorithms investigated in this PhD thesis are rather diverse. Nonetheless, we did not carry out a thorough comparison for transfer learning. The work presented in Chapter V is rather preliminary and only explored baseline algorithms. In particular for the regression part, it could be of interest to evaluate the performance of more advanced Gaussian process frameworks such as deep Gaussian processes, which have been developed in recent years [Le Gratiet, 2013; Raissi and Karniadakis, 2016; Moreno-Muñoz et al., 2021].

Finally, GPR served as a cornerstone in all presented reduced-order modelling frameworks as its robust interpolation capabilities were extensively employed for accurate estimation of mean scenarios. However, the thesis investigations did not explore the Gaussian process' full potential for uncertainty quantification. For instance, the Gaussian underlying distribution can help in the design of confidence intervals to exhibit scenarios at risk.

Application perspectives

From an application standpoint, future work includes applying the reduced-order model to more realistic atmospheric boundary-layer flows. The next step is to extend the approach to a three-dimensional field-scale case with multiple obstacles representative of an urban canopy. The MUST experiment [Biltoft, 2001] is a good candidate as it corresponds to a microscale

dispersion experiment through an idealised urban canopy made of a regular array of containers, for which 10 to 100 LES simulations can be carried out (as an indication, a LES simulation of a neutral MUST trial with the AVBP solver has a cost of about 20,000 CPU hours, which is about 25 times more than the two-dimensional case studied in this work). In this context, the purely LES data-driven reduced-order modelling approach could be tested and the applicability of the multi-fidelity approach could be explored (the performance of the EMUL-RANS-TE hybrid model needs to be assessed beforehand to verify its applicability and identify its limitations on the MUST trial). A first main issue with the MUST application lies in the capability of the dimensionality reduction component to deal with the high dimension of the field statistics (there is a jump of two orders of magnitude in the number of grid points, from 10^5 in the present case study to 10^7 grid points in the neutral MUST trial). A second main issue is the more complex physics that is represented in the LES snapshots for the neutral MUST trial. For instance, fundamental mechanisms like vortex stretching are absent in two-dimensional flow simulations. A more complex physics could induce stronger nonlinearities in the mapping between the field statistics and the uncertain input parameters. This study on the MUST experiment would be an interesting step to demonstrate the added value of machine-learning-based reduced-order model for microscale dispersion applications to produce ensemble forecasts and assess human exposure to toxic air pollutants in the event of an accident.

In the longer term, to continue the development of machine and deep learning approaches for micrometeorology, it would be of interest to extend the reduced-order modelling approach to different atmospheric stability conditions to go beyond neutral conditions. Stable conditions are known to be critical for air quality issues as they can favour pollutant accumulation in the lowest part of the atmospheric boundary-layer [Sabatier et al., 2021]. It is therefore of primary importance to evaluate the performance of the reduced-order modelling approach in this context. Unstable conditions are also of interest, both from an application viewpoint and from a methodological viewpoint. The physical processes become more complex as vertical convection becomes significant. This raises the question of the applicability of the reduced-order modelling approach proposed in this work. This also raises the treatment of the time dimension in the reduced-order model as unstable conditions induce unsteady processes and field statistics may no longer be the most relevant quantities of interest. Furthermore, unstable conditions are closely linked with extreme events such as large-scale wildfires, which induce the development of thermo-convective plumes and which can severely degrade air quality. This could be an interesting application to target in future work [Costes et al., 2022].

Conclusions et perspectives

L'idée centrale de ce travail de thèse était d'explorer comment les approches d'apprentissage automatique et profond peuvent être exploitées pour émuler des approches de mécanique des fluides numérique, précises mais coûteuses afin de prévoir l'écoulement et la dispersion de panache à la micro-échelle urbaine dans un contexte paramétrique, c'est-à-dire dans un contexte où des incertitudes importantes sur les conditions aux limites de l'écoulement et sur la localisation de la source d'émission sont présentes. L'accent a été mis sur l'apprentissage de la variabilité spatiale des champs statistiques d'intérêt (concentration moyenne du traceur, composantes moyennes de la vitesse d'écoulement, énergie cinétique turbulente, échelle de temps de l'écoulement turbulent) à partir de données de SGE, et de sa dépendance aux paramètres incertains d'écoulement et de source. Nous avons réalisé une comparaison détaillée de plusieurs modèles réduits sur une configuration canonique bidimensionnelle d'écoulement avec dispersion, avec l'idée de définir des recommandations sur les méthodes d'apprentissage statistique pour préparer les futures applications de dispersion à grande échelle.

Synthèse des principaux résultats

Afin d'identifier l'approche d'émulation la plus adaptée au contexte de dispersion micro-échelle, une comparaison détaillée des approches de réduction de dimension (décomposition orthogonale aux valeurs propres ou POD, autoencodeur convolutif) et des modèles de régression (décomposition en chaos polynomial, gradient de boosting, processus gaussiens) a été réalisée. Ceci a été rendu possible par l'étude d'un cas d'étude canonique bidimensionnel correspondant à un écoulement atmosphérique turbulent autour d'un obstacle isolé. Malgré les limitations sur la représentation de la turbulence, cette étude de cas présente des structures complexes d'écoulement. En amont de l'obstacle, la dispersion du panache est bimodale : le traceur est soit piégé dans une première région de recirculation le long de la face amont de l'obstacle, soit advecté en aval de l'obstacle. En aval de l'obstacle, l'écoulement est guidé par une association entre le délestage tourbillonnaire quasi-périodique induit par l'interaction écoulement-obstacle et par la turbulence grande échelle qui se propage depuis l'entrée du domaine. De plus, un écoulement inverse se produit près du sol, associé à une seconde région de recirculation, transportant le traceur vers l'obstacle en amont. La coût de simulation d'un tel cas à l'aide de SGE dans un contexte de requêtes multiples est envisageable de manière hors-ligne. Ainsi , une très grande base de données de SGE composée de 750 solutions correspondant à différents échantillons de paramètres d'entrée incertains (vitesse d'écoulement de référence en entrée du domaine, longueur de rugosité, position horizontale et hauteur de la source d'émission) a été générée. Elle comprend une diversité de topologies d'écoulement et de panache. Cette diversité rend le processus d'émulation complexe en raison de la réponse non-linéaire des statistiques du champ aux variations des paramètres incertains. Cette problématique devient d'autant plus importante que les statistiques du champ sont de très grande dimension (de l'ordre de 10^5 points de grille pour cette étude de cas). Pour éviter d'introduire des biais dans l'évaluation du processus d'émulation, l'ensemble a été divisé

en sous-ensembles d'entraînement, de calibration/validation et de test.

Premier objectif : mettre en œuvre un modèle réduit orienté données pour émuler les statistiques de champs SGE

Le Chapitre IV a démontré la capacité d'un modèle réduit basé sur des outils d'apprentissage à émuler les champs moyens de concentration de traceur obtenus à partir d'un ensemble de simulations SGE paramétriques. Le processus d'émulation permet de reproduire avec précision les résultats LES pour un grand ensemble de données d'entraînement (composé d'au moins 100 solutions de SGE), tout en réduisant de plusieurs ordres de grandeur le coût de calcul pour prévoir la réponse à un nouveau jeu de paramètres incertains.

- La POD a été utilisée comme approche de référence pour compresser les statistiques de champ de grande dimension. Une étude détaillée des modes POD a mis en évidence les difficultés de la POD à traiter la position incertaine de la source d'émission, qui induit par définition une variabilité spatiale du sillage de concentration de traceur en amont de l'obstacle. Ce lien étroit entre l'espace d'incertitude et le domaine spatial est la raison principale de ces difficultés. Il nécessite un très grand nombre de modes (environ 100 modes pour la base de données d'entraînement de 450 solutions de SGE) pour représenter la grande variabilité spatiale de la concentration du traceur près des sources d'émission. Les premiers modes POD suffisent pour capturer la variabilité spatiale de la concentration du traceur dans les zones où la diffusion est importante, par exemple dans la zone de recirculation derrière l'obstacle. A l'inverse, il est nécessaire d'inclure des modes d'ordre élevé dans la base POD pour capturer les structures fines du panache dans les régions proches des sources d'émission en amont de l'obstacle, où l'advection domine.

- La meilleure approche de métamodélisation pour représenter la réponse des coefficients réduits POD aux changements des paramètres incertains s'est avérée être la régression par processus gaussiens, avec une optimisation mode par mode des hyperparamètres pour s'adapter à la large gamme d'échelles spatiales présente dans les modes POD. Cette approche a conduit à un très bon score global Q^2 (supérieur à 95%) pour une grande base de données d'entraînement de type SGE. Cette meilleure performance des processus gaussiens par rapport aux autres modèles de régression peut s'expliquer par le faible bruit des données d'entraînement et la présence de fortes non-linéarités dans la relation entre les paramètres incertains et les coefficients réduits POD. Les modèles d'interpolation comme les processus gaussiens excellent dans ces conditions.
  Nous avons analysé plus en détails le comportement du modèle de régression par processus gaussiens à l'aide d'un score Q^2 en chaque nœud de maillage pour comprendre la variabilité spatiale de la performance de prévision. Nous avons montré que la concentration moyenne du traceur dans les zones de recirculation est bien prédite, tandis qu'une diminution de la performance apparaît proche des sources d'émission. La réponse de la concentration moyenne du traceur y est portée par des modes POD d'ordre élevé. Ces modes d'ordre élevé présentent des motifs fins et localisés, associés aux emplacements de

source d'émission perturbés et difficiles à prédire car caractérisés par un bruit fort. Nous avons également montré que la difficulté à prévoir les modes d'ordre élevé augmente lorsque l'ensemble de données d'entraînement est réduit, mais la performance de prévision reste acceptable lorsque l'on considère au moins 100 solutions de type SGE dans la base de données d'entraînement.
En outre, le processus d'optimisation des hyperparamètres du processus gaussien effectué pour chaque coefficient réduit POD a été amélioré pour gagner en coût de calcul. Nous avons montré qu'une distribution a priori satisfaisante pour les hyperparamètres peut être obtenue à partir des modes POD et peut être utilisée pour informer une procédure d'optimisation MAP. Nous avons constaté que la procédure MAP fournit des résultats similaires à l'approche plus standard de la maximisation MLL à N répétitions (en anglais – *restarts*). MAP n'utilise qu'une seule descente de gradient bien choisie, fournissant ainsi un cadre de d'entraînement à la fois efficace et fiable.

– La capacité de l'autoencodeur convolutif à mieux compresser les statistiques des champs dans le modèle réduit basé sur les processus gaussiens a été étudiée pour pallier les limitations de la POD, en s'inspirant des travaux de Fukami et al. [2020]. Nous avons montré que pour la base de données d'entraînement complète (composée de 450 solutions de type SGE), il est possible de réduire la taille de l'espace latent d'un facteur 10 par rapport à la POD (10 variables latentes contre 100 modes POD) sans perte d'information ; il y a même une amélioration de la précision de la prévision. Nous avons également mis en évidence que l'entraînement de l'autoencodeur devient difficile lorsque la base de données d'entraînement est réduite. La cohérence des résultats n'est pas garantie et le comportement de l'autoencodeur est difficile à anticiper en raison de sa nature de type boîte noire.

Deuxième objectif principal : informer une équation de transport de scalaire RANS à partir de statistiques d'écoulement LES pour mettre en œuvre un modèle réduit hybride

Le modèle réduit purement orienté données nécessite une grande base de données d'entraînement de type SGE pour obtenir une prévision précise sans artefacts numériques, ce qui est hors de portée pour des cas d'application réalistes tridimensionnels. La nécessité d'un grand ensemble de données d'entraînement est principalement due aux structures fines du panache dans les régions proches de la source causées par l'incertitude sur la position de la source d'émission. En réduisant le nombre de solutions disponibles pour l'entraînement à moins de 100, une perte de cohérence avec les lois de la physique a été observée : par exemple, des structures bruitées non physiques apparaissent dans les régions sans traceur. En effet, le modèle réduit est construit uniquement sur un paradigme statistique et n'est pas contraint par les lois de la physique. Pour pallier ces limitations, nous avons mis en œuvre une approche alternative de modélisation hybride RANS/modèle réduit (EMUL-RANS-TE) dans le Chapitre V, qui est basée sur l'idée-clé d'injecter des informations détaillées sur l'écoulement provenant de SGE dans une équation de transport de traceur de moindre fidélité dans le formalisme RANS.

– Le modèle réduit proposé EMUL-RANS-TE repose sur l'émulation des statistiques de

l'écoulement d'air (composantes du champ d'écoulement moyen, énergie cinétique turbulente, échelle de temps de l'écoulement turbulent) à partir de la SGE en utilisant un modèle réduit spécifique à chaque quantité. Ce modèle réduit combine la POD et la régression par processus gaussiens comme dans le Chapitre IV, à la différence que le nombre de modes POD à retenir dans la base réduite est maintenant très faible (inférieur ou égal à 10) car le problème d'émulation est simplifié avec seulement les conditions aux limites d'écoulement comme source d'incertitude. Dans un deuxième temps, les statistiques des SGE émulées sont intégrées dans l'équation de transport de scalaire RANS pour fermer le flux massique de traceur turbulent moyenné. Le modèle hybride peut ensuite être intégrer pour faire varier la position horizontale et la hauteur de la source d'émission afin d'obtenir des prévisions du champ de concentration moyen. La principale limitation provient du modèle de fermeture RANS sur le flux de masse, qui pénalise les prévisions vis-à-vis des solutions de références des SGE dans la zone de recirculation dans le sillage de l'obstacle.

– La capacité de l'approche hybride EMUL-RANS-TE à émuler les champs moyens de concentration a été démontrée à travers une analyse spatiale du Q^2. Nous avons montré qu'EMUL-RANS-TE fournit une prévision précise dans les régions dominées par l'advection moyenne et le cisaillement près des sources d'émission, et élimine les artefacts observés dans les prévisions du modèle réduit orienté données (DIRECT-ROM-HF) discuté au "Chapitre IV. Cependant, la performance d'EMUL-RANS-TE est réduite dans la zone de recirculation en aval en raison des insuffisances du modèle de fermeture RANS du flux massique turbulent.

– Avec le développement de l'approche hybride EMUL-RANS-TE, des modèles avec différents niveaux de fidélité (SGE, approche hybride SGE/RANS) sont disponibles. Ils ont été intégrés dans un cadre de modèle réduit multi-fidélité (DIRECT-ROM-MF) afin de bénéficier de leurs points forts, c'est-à-dire la prévision précise du modèle réduit orienté données en aval et celle du modèle EMUL-RANS-TE en amont de l'obstacle. Un deuxième avantage est qu'une très grande base de données d'entraînement (composée de 450 solutions EMUL-RANS-TE et de 50 solutions SGE) peut être générée puisque l'intégration de l'approche hybride EMUL-RANS-TE est beaucoup moins chère (d'un facteur 10 à 100) que l'intégration directe d'une simulation SGE. L'approche multi-fidélité s'appuie sur des autoencodeurs convolutifs entraînés via apprentissage par transfert pour la réduction de dimension, et sur la régession par processus gaussiens via le co-krigeage pour l'interpolation. L'approche multi-fidélité a obtenu des résultats très satisfaisants en matière de reconstruction et de prévision, et a même surpassé ceux obtenus avec le modèle réduit orienté données SGE à budget équivalent.

À la suite de ce travail de thèse, la multi-fidélité apparaît comme une approche prometteuse pour tirer profit des solutions de SGE haute-fidélité afin de construire à moindre coût un modèle réduit précis pour la dispersion à micro-échelle.

Perspectives

Perspectives méthodologiques

D'un point de vue méthodologique, les futurs travaux viseront à la fois à *(i)* à améliorer l'approche par modélisation réduite en termes de réduction de dimension et de régression, et *(ii)* à pleinement exploiter l'expression bayésienne des processus gaussiens pour la régression.

En ce qui concerne la réduction de dimension, les algorithmes testés ont minimisé l'erreur quadratique moyenne (MSE pour *mean-squared error* en anglais) de la reconstruction par rapport aux champs d'origine (tant pour la POD centrée que pour les autoencodeurs). Nous avons d'abord appliqué une formulation standard de la POD centrée, correspondant à la diagonalisation de la matrice de covariance et équivalente à la diagonalisation de la matrice des snapshots centrée, comme algorithme de base pour la réduction de dimension. Ceci équivaut à minimiser l'erreur MSE moyenne sur toutes les caractéristiques (les nœds du maillage). Cette formulation est adaptée à la modélisation des régions à forte variance d'ensemble. Cependant, elle n'est pas très adaptée à la modélisation des régions à faible concentration, correspondant souvent aux régions à faible variance d'ensemble. De plus, cette formulation n'intègre aucun type de contraintes physiques. Une perspective intéressante serait d'ajuster la fonction objectif pour améliorer les propriétés des champs reconstruits. Par exemple, la modification des poids dans l'espace des caractéristiques pour la POD n'a pas été étudiée dans ce travail. Ceci pourrait être effectué en réduisant la matrice des snapshots (c'est-à-dire en transformant les caractéristiques pour que la variance soit égale à un), la POD réduite et centrée correspond alors à la diagonalisation de la matrice de corrélation. Ruan et al. [2006] ont montré que les régions de faible variance d'ensemble sont mieux capturées avec cette formulation, mais au détriment des régions de forte variance d'ensemble. Comme pour les autoencodeurs, une meilleure configuration de la fonction objectif pourrait améliorer la robustesse, la cohérence physique ou l'interprétabilité. Il s'agit d'un domaine de recherche très actif, où des termes de pénalisation et des contraintes supplémentaires peuvent être appliqués à l'espace latent, aux champs de sortie, ou aux poids du réseau [Champion et al., 2019; Kingma et al., 2019]. Par exemple, Champion et al. [2019] ont ajouté des contraintes de parcimonie et de régularité à la fonction objectif pour favoriser l'interprétabilité et la généralisation du réseau de neurones. Comme les algorithmes de réduction de dimension sont principalement utilisés pour produire une représentation de l'espace latent que les algorithmes de régression seront capables d'exploiter en sortie, des contraintes statistiques peuvent être appliquées sur l'espace latent. Par exemple, les autoencodeurs variationnels [Kingma et al., 2019] imposent une distribution gaussienne multidimensionnelle sur l'espace latent. Du point de vue de l'algorithme de régression, des contraintes peuvent également être ajoutées pour améliorer la cohérence physique et la robustesse. Par exemple, les développements récents sur les processus de régression gaussiens contraints [Swiler et al., 2020] peuvent aider à mieux définir la variété de sortie dans l'espace latent.

Les modèles réduits que nous avons utilisés dans ce travail impliquent réduction de dimension et modèles de régression qui ont été entraînés indépendamment les uns des autres. Tout d'abord,

nous avons mis en œuvre un modèle de régression par mode de l'espace latent donnant une sortie unique. Une telle méthodologie ne tient pas compte des corrélations entre les modes. Dans de futurs travaux, la valeur ajoutée de l'optimisation conjointe pourra être évaluée pour les modèles de régression à sorties multiples [Borchani et al., 2015]. Il serait également intéressant de combiner simultanément réduction de dimension et apprentissage par régression. Ceci pourrait être fait en combinant un autoencodeur avec un réseau perceptron multicouche. À long terme, il pourrait être intéressant d'étudier si la propagation du gradient peut permettre l'apprentissage intégré de processus gaussiens, d'arbres de décision ou d'autres outils de régression avec des autoencodeurs, bien que cette approche induira sans doute des problèmes de surapprentissage. Enfin, les classes d'algorithmes étudiées dans cette thèse sont assez diverses. Néanmoins, nous n'avons pas effectué une comparaison approfondie pour l'apprentissage par transfert. Le travail présenté dans le Chapitre chap:5 est plutôt préliminaire et n'a exploré que des algorithmes de base. En particulier, pour la partie régression, il pourrait être intéressant d'évaluer les performances de processus gaussiens plus avancés tels que les processus gaussiens profonds, qui se sont développés ces dernières années [Le Gratiet, 2013; Raissi and Karniadakis, 2016; Moreno-Muñoz et al., 2021].

Enfin, les GPR ont joué un rôle central dans tous les modèles réduits présentés dans cette thèse. Leur capacité d'interpolation a été largement utilisée pour pour l'estimation de scénarios en moyenne. Néanmoins, les investigations de cette thèse sont restées limitées à cet aspect et n'ont pas exploré pleinement le potentiel du processus gaussien pour la modélisation de l'incertitude, notamment via l'exploitation de la covariance sous-jacente de la distribution gaussienne. En exploitant la distribution complète dérivée des GPR, il devient possible de générer des intervalles de confiance bien conçus, offrant ainsi une perspective intéressante pour quantifier le manque de prévisibilité de nouveaux scénarios.

Perspectives applicatives

Du point de vue de l'application, les futurs travaux prévoient l'application du modèle réduit à des écoulements de couche limite atmosphérique plus réalistes. La prochaine étape vise à étendre l'approche à un cas tridimensionnel, à l'échelle d'un terrain et composé d'obstacles multiples, représentatif d'une canopée urbaine. L'expérience MUST [Biltoft, 2001] est un bon exemple car elle correspond à une expérience de dispersion à micro-échelle à travers une canopée urbaine idéalisée constituée d'une série de conteneurs régulièrement répartis, pour laquelle 10 à 100 simulations de type SGE peuvent être réalisées (à titre indicatif, une simulation SGE d'un essai MUST dans les conditions neutres avec le solveur AVBP a un coût d'environ 20 000 heures CPU, soit environ 25 fois plus que le cas bidimensionnel étudié dans ce travail). Dans ce contexte, l'approche de modélisation réduite orientée données de SGE pourrait être testée et l'applicabilité de l'approche multi-fidélité pourrait être explorée (les performances du modèle hybride EMUL-RANS-TE doivent être évaluées au préalable pour vérifier son applicabilité et identifier ses limites sur le cas MUST). Une première problématique importante avec l'application MUST réside dans la capacité des outils de réduction de dimension à traiter la très grande dimension des statistiques de champ (la dimension du maillage est d'environ deux ordres de grandeur au-dessus du cas

bidimensionnel, de 10^5 dans la présente étude de cas à 10^7 points de grille dans le cas neutre MUST). Une deuxième problématique importante est la physique plus complexe contenue dans les simulations SGE du cas MUST sous conditions atmosphériques neutres. Par exemple, des mécanismes fondamentaux tels que les étirements de tourbillon ne sont pas représentés dans les simulations d'écoulement bidimensionnel. Une physique plus complexe pourrait induire des non-linéarités plus fortes dans la relation entre les paramètres d'entrée incertains et les statistiques de sortie. Cette étude sur l'expérience MUST serait une étape intéressante pour démontrer la valeur ajoutée du modèle réduit basé sur les algorithmes d'apprentissage pour aller vers les applications de dispersion à micro-échelle afin de produire des prévisions d'ensemble et d'évaluer l'exposition sanitaire aux polluants atmosphériques toxiques en cas d'accident.

À plus long terme, pour poursuivre le développement des approches d'apprentissage automatique et profond pour la micrométéorologie, il serait intéressant d'étendre l'approche de modélisation réduite à différentes conditions de stabilité atmosphérique pour aller au-delà des conditions neutres. Les conditions stables sont connues pour être critiques pour la problématique de la qualité de l'air car elles peuvent favoriser l'accumulation de polluants dans la partie la plus basse de la couche limite atmosphérique [Sabatier et al., 2021]. Il est donc de première importance d'évaluer la performance des modèles réduits dans ce contexte. Les conditions instables présentent également un intérêt, tant du point de vue applicatif que méthodologique. Les processus physiques deviennent plus complexes lorsque la convection verticale devient importante. Ceci pose la question de l'applicabilité de l'approche par modèle réduit proposée dans ce travail. Cela soulève également la question du traitement de la dimension temporelle dans le modèle réduit, car les conditions instables induisent des processus instationnaires et les statistiques de champ peuvent ne plus être les quantités d'intérêt les plus pertinentes. De plus, les conditions instables sont étroitement liées à l'occurrence d'événements extrêmes tels que les incendies de forêt à grande échelle, qui induisent le développement de panaches thermo-convectifs et qui peuvent fortement dégrader la qualité de l'air. Ceci pourrait être une application intéressante à cibler dans des travaux futurs [Costes et al., 2022].

Nomenclature

Acronyms

ABL/PBL/UBL Atmospheric/Planetary/Urban boundary-layer

AE Autoencoder

ARD Automatic relevance determination

CART Classification and regression trees

CCP Cost complexity pruning

CDF Cumulative distribution function

CFD Computational fluid dynamics

CFL Courant–Friedrichs–Lewy

CNN Convolutional neural network

CPU Central processing unit

DNS Direct numerical simulation

EVM Eddy viscosity model

GB Gradient Boosting

GGDH Generalised gradient diffusion hypothesis

GPR Gaussian process regression

GPU Graphics processing unit

HOGGDH High-order generalised gradient diffusion hypothesis

kNN k-nearest-neighbours

LES Large-Eddy Simulation

LF/MF/HF Lower/Multi/High-order fidelity

LIC Line integral convolution

MAP Maximum *a posteriori*

MLL Maximum log-likelihood

MLP Multiple-Layer Perceptron

MSE Mean Square Error

MUST Mock urban setting test

NMF/NNMF Non-negative matrix factorisation

PC/PCE Polynomial chaos expansion

PGS Pressure gradient scaling

POD Proper Orthogonal Decomposition

RANS Reynolds-Averaged Navier-Stokes

RBF Radial basis function

ReLU Rectified linear unit

RSM Reynolds stress equation model

SGD Stochastic gradient descent

SGDH Standard gradient diffusion hypothesis

SGS Subgrid scale

Tanh Hyperbolic tangent

TKE Turbulent Kinetic Energy

TTG two-step Taylor-Galerkin

List of Figures

List of Tables

Bibliography

1d convolution, 2022. URL `https://peltarion.com/knowledge-center/documentation/modeling-view/build-an-ai-model/blocks/1d-convolution`. 49, 176

K Abe and K Suga. Towards the development of a Reynolds-averaged algebraic turbulent scalar-flux model. *International Journal of Heat and Fluid Flow*, 22(1):19–29, 2001. 135

E Ajuria Illarramendi, M Bauerheim, and B Cuenot. Performance and accuracy assessments of an incompressible fluid solver coupled with a deep convolutional neural network. *Data-Centric Engineering*, 3:e2, 2022. 36

KJ Allwine and JE Flaherty. Joint Urban 2003: Study overview and instrument locations. Technical report, Pacific Northwest National Laboratory (PNNL), Richland, WA, USA, 2006. 18

KJ Allwine, JH Shinn, GE Streit, KL Clawson, and M Brown. Overview of urban 2000: A multiscale field study of dispersion through an urban environment. *Bulletin of the American Meteorological Society*, 83(4):521–536, 2002. 18

KJ Allwine, MJ Leach, LW Stockham, JS Shinn, RP Hosker, JF Bowers, and JC Pace. Overwiew of Joint Urban 2003 – an atmospheric dispersion study in Oklahoma City. *Symposium on Planning, Nowcasting, and Forecasting in the Urban Zone*, 2004. 18

S Arnold, H ApSimon, J Barlow, S Belcher, M Bell, R Britter, R Colvile, H Cheng, A Dobre, BR Greally, et al. Dispersion of air pollution & penetration into the local environment dapple. *Science of the Total Environment*, 332:139–153, 2004. 18

F Baetke, H Werner, and H Wengle. Numerical simulation of turbulent flow over surface-mounted obstacles with sharp edges and corners. *Journal of Wind Engineering and Industrial Aerodynamics*, 35:129–147, 1990. 26

Z Bai, SL Brunton, BW Brunton, JN Kutz, E Kaiser, A Spohn, and BR Noack. Data-driven methods in fluid dynamics: Sparse classification from experimental data. In *Whither turbulence and big data in the 21st century?*, pages 323–342. Springer, 2017. 28

J-J Baik and J-J Kim. On the escape of pollutants from urban street canyons. *Atmospheric Environment*, 36(3):527–536, 2002. 17

P Baldi and K Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural networks*, 2(1):53–58, 1989. 32

MF Barone, I Kalashnikova, DJ Segalman, and HK Thornquist. Stable galerkin reduced order models for linearized compressible flow. *Journal of Computational Physics*, 228(6):1932–1946, 2009. 34

A Baskaran and T Stathopoulos. Computational evaluation of wind effects on buildings. *Building and Environment*, 24(4):325–333, 1989. 26

A Baskaran and T Stathopoulos. Influence of computational parameters on the evaluation of wind effects on the building envelope. *Building and Environment*, 27(1):39–40, 1992. 26

F Bazdidi-Tehrani, A Ghafouri, and M Jadidi. Grid resolution assessment in large eddy simulation of dispersion around an isolated cubic building. *Journal of Wind Engineering and Industrial Aerodynamics*, 121:1–15, 2013. 71

A Beck, D Flad, and C-D Munz. Deep neural networks for data-driven les closure models. *Journal of Computational Physics*, 398:108910, 2019. 36

SE Belcher. Mixing and transport in urban areas. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 363(1837):2947–2968, 2005. 11

L Belhoucine, M Deville, AR Elazehari, and MO Bensalah. Explicit algebraic reynolds stress model of incompressible turbulent flow in rotating square duct. *Computers & Fluids*, 33(2): 179–199, 2004. 35

G Berkooz, P Holmes, and JL Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 25(1):539–575, 1993. 31, 43

M Berveiller, B Sudret, and M Lemaire. Stochastic finite element: a non intrusive approach by regression. *European Journal of Computational Mechanics/Revue Européenne de Mécanique Numérique*, 15(1-3):81–92, 2006. 59

CA Biltoft. Customer report for Mock Urban Setting Test. *DPG Document*, (8-CO):160–000, 2001. 18, 163, 170

G Blatman. *Adaptive sparse polynomial chaos expansions for uncertainty propagation and sensitivity analysis*. PhD thesis, Université Clermont-Ferrand, 2009. 59

G Blatman and B Sudret. Adaptive sparse polynomial chaos expansion based on least angle regression. *Journal of Computational Physics*, 230(6):2345–2367, 2011. 60

B Blocken. 50 years of computational wind engineering: past, present and future. *Journal of Wind Engineering and Industrial Aerodynamics*, 129:69–102, 2014. 20

B Blocken. LES over RANS in building simulation for outdoor and indoor applications: A foregone conclusion? In *Building Simulation*, volume 11, pages 821–870. Springer, 2018. 2, 8, 20, 24, 26

B Blocken and J Carmeliet. Pedestrian wind conditions at outdoor platforms in a high-rise apartment building: generic sub-configuration validation, wind comfort assessment and uncertainty issues. *Wind and Structures*, 11(1):51–70, 2008. 26

B Blocken, P Moonen, T Stathopoulos, and J Carmeliet. Numerical study on the existence of the venturi effect in passages between perpendicular buildings. *Journal of Engineering Mechanics*, 134(12):1021–1028, 2008a. 16, 19, 26

B Blocken, T Stathopoulos, P Saathoff, and X Wang. Numerical evaluation of pollutant dispersion in the built environment: comparisons between models and experiments. *Journal of Wind Engineering and Industrial Aerodynamics*, 96(10-11):1817–1831, 2008b. 71

B Blocken, T Stathopoulos, J Carmeliet, and JLM Hensen. Application of computational fluid dynamics in building performance simulation for the outdoor environment: an overview. *Journal of Building Performance Simulation*, 4(2):157–184, 2011. 26

B Blocken, T Stathopoulos, and JPAJ Van Beeck. Pedestrian-level wind conditions around buildings: Review of wind-tunnel and CFD techniques and their accuracy for wind comfort assessment. *Building and Environment*, 100:50–81, 2016a. 26

B Blocken, R Vervoort, and T van Hooff. Reduction of outdoor particulate matter concentrations by local removal in semi-enclosed parking garages: a preliminary case study for eindhoven city center. *Journal of Wind Engineering and Industrial Aerodynamics*, 159:80–98, 2016b. 24, 26

J Bluett, C Heydenrych, G Fisher, T Freeman, and J Godfrey. *Good Practice Guide: Atmospheric dispersion modelling*. 2002. 21

Hanen Borchani, Gherardo Varando, Concha Bielza, and Pedro Larranaga. A survey on multi-output regression. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 5(5):216–233, 2015. 163, 170

L Bottou et al. Online learning and stochastic approximations. *On-line learning in neural networks*, 17(9):142, 1998. 52

MA Bouhlel, JT Hwang, N Bartoli, R Lafage, J Morlier, and JRRA Martins. A python surrogate modeling framework with derivatives. *Advances in Engineering Software*, page 102662, 2019. 87

Y-L Boureau, J Ponce, and LeCun Y. A theoretical analysis of feature pooling in visual recognition. In *International Conference on Machine Learning*, 2010. 50

H Bourlard and Y Kamp. Auto-association by multilayer perceptrons and singular value decomposition. *Biological cybernetics*, 59(4):291–294, 1988. 32

L Breiman, JH Friedman, RA Olshen, and CJ Stone. *Classification and regression trees*. Routledge, 2017. 65

L Brevault, M Balesdent, and A Hebbal. Overview of Gaussian process based multi-fidelity techniques with variable relationship between fidelities, application to aerospace systems. *Aerospace Science and Technology*, 107:106339, 2020. 150

RE Britter and SR Hanna. Flow and dispersion in urban areas. *Annual Review of Fluid Mechanics*, 35(1):469–496, 2003. 1, 7, 11

RE Britter, S Di Sabatino, F Caton, KM Cooke, PG Simmonds, and G Nickless. Results from three field tracer experiments on the neighbourhood scale in the city of Birmingham UK. *Water, Air and Soil Pollution: Focus*, 2(5):79–90, 2002. 18

B Brunekreef and ST Holgate. Air pollution and health. *The lancet*, 360(9341):1233–1242, 2002. 1, 7

S Brunton, B Noack, and P Koumoutsakos. Machine learning for fluid mechanics. *arXiv preprint arXiv:1905.11075*, 2019. 27, 30, 31, 33, 175

SL Brunton, JL Proctor, and JN Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, 2016. 28, 33

W Brutsaert. *Evaporation into the atmosphere: theory, history and applications*, volume 1. Springer Science & Business Media, 2013. 74, 80

B Cabral and LC Leedom. Imaging vector fields using line integral convolution. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 263–270, 1993. 77

RH Cameron and WT Martin. The orthogonal development of non-linear functionals in series of fourier-hermite functionals. *Annals of Mathematics*, pages 385–392, 1947. 58

M Carpentieri and AG Robins. Influence of urban morphology on air flow over building arrays. *Journal of Wind Engineering and Industrial Aerodynamics*, 145:61–74, 2015. 17

CEDVAL. CEDVAL at Hamburg University Compilation of Experimental Data for Validation of Microscale Dispersion Models; WebSite provided by the Environmental Wind Tunnel Laboratory (EWTL) of the Meteorological Institute, 2022. URL `https://mi-pub.cen.uni-hamburg.de/index.php?id=432`. 19

K Champion, B Lusch, JN Kutz, and SL Brunton. Data-driven discovery of coordinates and governing equations. *Proceedings of the National Academy of Sciences*, 116(45):22445–22451, 2019. 163, 169

TL Chan, G Dong, CW Leung, CS Cheung, and WT Hung. Validation of a two-dimensional pollutant dispersion model in an isolated street canyon. *Atmospheric Environment*, 36(5): 861–872, 2002. 17

H Choi and P Moin. Grid-point requirements for large eddy simulation: Chapman's estimates revisited. *Physics of Fluids*, 24(1):011702, 2012. 28

F Chollet et al. Keras. `https://keras.io`, 2015. 87

O Colin and M Rudgyard. Development of high-order Taylor–Galerkin schemes for LES. *Journal of Computational Physics*, 162(2):338–371, 2000. doi: 10.1006/jcph.2000.6538. 72

A Costes, Q Rodier, V Masson, C Lac, and MC Rochoux. Effects of high-density gradients on wildland fire behavior in coupled atmosphere-fire simulations. *Journal of Advances in Modeling Earth Systems*, page e2021MS002955, 2022. 164, 171

JC Couillet. Methods for the assessment and prevention of accidental risks (dra-006). ω-12, atmospheric dispersion (mechanisms and calculation tools). 2002. 14, 21

TJ Craft, BE Launder, and K Suga. Development and application of a cubic eddy-viscosity model of turbulence. *International Journal of Heat and Fluid Flow*, 17(2):108–115, 1996. 35

BJ Daly and FH Harlow. Transport equations in turbulence. *The Physics of Fluids*, 13(11): 2634–2649, 1970. 135

T Dauxois, T Peacock, P Bauer, CP Caulfield, C Cenedese, C Gorlé, G Haller, GN Ivey, PF Linden, E Meiburg, N Pinardi, NM Vriend, and AW Woods. Confronting grand challenges in environmental fluid mechanics. *Physical Review Fluids*, 6:020501, 2021. 1, 2, 7, 8, 27, 28, 36, 39

N Deng, BR Noack, M Morzyński, and LR Pastur. Galerkin force model for transient and post-transient dynamics of the fluidic pinball. *Journal of Fluid Mechanics*, 918:A4, 2021. 32

A Dobre, SJ Arnold, RJ Smalley, JWD Boddy, JF Barlow, AS Tomlin, and SE Belcher. Flow field measurements in the proximity of an urban intersection in London, UK. *Atmospheric Environment*, 39(26):4647–4657, 2005. 18

Y Du, B Blocken, and S Pirker. A novel approach to simulate pollutant dispersion in the built environment: Transport-based recurrence CFD. *Building and Environment*, 170:106604, 2020. 130, 131

J Duchi, E Hazan, and Yo Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(7), 2011. 53

K Duraisamy, G Iaccarino, and H Xiao. Turbulence modeling in the age of data. *Annual Review of Fluid Mechanics*, 51(1):357–377, 2019a. 36

K Duraisamy, G Iaccarino, and H Xiao. Turbulence modeling in the age of data. *Annual Review of Fluid Mechanics*, 51:357–377, 2019b. 27, 28

H Eivazi, S Le Clainche, S Hoyas, and R Vinuesa. Towards extraction of orthogonal and parsimonious non-linear modes from turbulent flows. *Expert Systems with Applications*, 202:117038, 2022. 33

S El Garroussi, S Ricci, M De Lozzo, N Goutal, and D Lucor. Tackling random fields nonlinearities with unsupervised clustering of polynomial chaos expansion in latent space: application to global sensitivity analysis of river flooding. *Stochastic Environmental Research and Risk Assessment*, 36:693–718, 2022. 33, 53

A Elbeltagi, N Kumari, JK Dharpure, A Mokhtar, K Alsafadi, M Kumar, B Mehdinejadiani, HR Etedali, Y Brouziyne, ARMT Islam, et al. Prediction of Combined Terrestrial Evapotranspiration Index (CTEI) over large river basin based on machine learning approaches. *Water*, 13(4), 2021. 62

JH Ferziger. Approaches to turbulent flow computation: applications to flow over obstacles. *Journal of Wind Engineering and Industrial Aerodynamics*, 35:1–19, 1990. 24

J Franke, A Hellsten, KH Schlunzen, and B Carissimo. The COST 732 best practice guideline for CFD simulation of flows in the urban environment: A summary. *International Journal of Environment and Pollution*, 44(1–4):419–427, 2011. 1, 7, 74

H Frezat, G Balarac, J Le Sommer, R Fablet, and R Lguensat. Physical invariance in neural networks for subgrid-scale scalar flux modeling. *Physical Review Fluids*, 6(2):024607, 2021. 30

J Friedman. Stochastic gradient boosting. Department of Statistics. Stanford University, 1999. 65

JH Friedman. Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, pages 1189–1232, 2001. 65, 102

K Fukami, T Nakamura, and K Fukagata. Convolutional neural network based hierarchical autoencoder for nonlinear mode decomposition of fluid field data. *Physics of Fluids*, 32(9): 095110, 2020. 33, 50, 120, 126, 161, 167, 176

H Gamel. *Caractérisation expérimentale de l'écoulement et de la dispersion autour d'un obstacle bidimensionnel*. PhD thesis, École Centrale de Lyon, France, 2015. 19, 76, 82, 135, 137

V Garbero. *Pollutant dispersion in urban canopy study of the plume behaviour through an obstacle array*. PhD thesis, École Centrale de Lyon, France, 2008. 17, 175

V Garbero, P Salizzoni, and L Soulhac. Experimental study of pollutant dispersion within a network of streets. *Boundary-Layer Meteorology*, 136(3):457–487, 2010. 19

C García-Sánchez, DA Philips, and C Gorlé. Quantifying inflow uncertainties for CFD simulations of the flow in downtown Oklahoma City. *Building and Environment*, 78:118–129, 2014. 28, 29, 33, 37, 53, 79, 80

C García-Sánchez, G Van Tendeloo, and C Gorlé. Quantifying inflow uncertainties in RANS simulations of urban pollutant dispersion. *Atmospheric Environment*, 161:263–273, 2017. 3, 4, 9, 26, 29, 33, 37, 53, 79

C García-Sanchez, J van Beeck, and C Gorlé. Predictive large eddy simulations for urban flows: Challenges and opportunities. *Building and Environment*, 139:146–156, 2018. 2, 8, 30

J Gardner, G Pleiss, KQ Weinberger, D Bindel, and AG Wilson. Gpytorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration. *Advances in Neural Information Processing Systems*, 31, 2018. 87

JR Garratt. *The Atmospheric Boundary Layer*. Cambridge University Press, Cambridge, UK, 1992. 13

TB Gatski and CG Speziale. On explicit algebraic stress models for complex turbulent flows. *Journal of Fluid Mechanics*, 254:59–78, 1993. 35

M Germano, U Piomelli, P Moin, and WH Cabot. A dynamic subgrid-scale eddy viscosity model. *Physics of Fluids A: Fluid Dynamics*, 3(7):1760–1765, 1991. 25

R Ghanem. Ingredients for a general purpose stochastic finite elements implementation. *Computer Methods in Applied Mechanics and Engineering*, 168(1-4):19–34, 1999. 58

LYM Gicquel, N Gourdain, J-F Boussuge, H Deniau, G Staffelbach, P Wolf, and T Poinsot. High performance parallel computing of flows in complex geometries. *Comptes Rendus Mecanique*, 339(2-3):104–124, 2011. 72

D Gleichauf, C Dollinger, N Balaresque, AD Gardner, M Sorg, and A Fischer. Thermographic flow visualization by means of non-negative matrix factorization. *International Journal of Heat and Fluid Flow*, 82:108528, 2020. 32

X Glorot and Y Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010. 51, 52

X Glorot, A Bordes, and Y Bengio. Deep sparse rectifier neural networks. In G Gordon, D Dunson, and M Dudík, editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 315–323, Fort Lauderdale, FL, USA, 11–13 Apr 2011. 52

R Goldstein. *Fluid Mechanics Measurements*. Routledge, 2017. 27

I Goodfellow, Y Bengio, and A Courville. *Deep Learning*. MIT press, 2016. 48, 49, 50, 57, 149, 150, 176

IJ Goodfellow. Technical report: Multidimensional, downsampled convolution for autoencoders. *Université de Montréal*, 2010. 32

P Gousseau, B Blocken, T Stathopoulos, and GJF Van Heijst. CFD simulation of near-field pollutant dispersion on a high-resolution grid: a case study by LES and RANS for a building group in downtown Montreal. *Atmospheric Environment*, 45(2):428–438, 2011. 24

P Gousseau, B Blocken, and GJF Van Heijst. Large-eddy simulation of pollutant dispersion around a cubical building: Analysis of the turbulent mass transport mechanism by unsteady concentration and velocity statistics. *Environmental Pollution*, 167:47–57, 2012. 71

CSB Grimmond and TR Oke. Aerodynamic properties of urban areas derived from analysis of surface form. *Journal of Applied Meteorology and Climatology*, 38(9):1262–1292, 1999. 16

C Gromke and B Blocken. Influence of avenue-trees on air quality at the urban neighborhood scale. Part II: Traffic pollutant concentrations at pedestrian level. *Environmental Pollution*, 196:176–184, 2015. 24

C Gromke, R Buccolieri, S Di Sabatino, and B Ruck. Dispersion study in a street canyon with tree planting by means of wind tunnel and numerical investigations–evaluation of cfd data with experimental data. *Atmospheric Environment*, 42(37):8640–8650, 2008. 17

T Grylls, CMA Le Cornec, P Salizzoni, L Soulhac, MEJ Stettler, and M van Reeuwijk. Evaluation of an operational air quality model using large-eddy simulation. *Atmospheric Environment*, 3:100041, 2019. 26

Y Guan, SL Brunton, and I Novosselov. Sparse nonlinear models of chaotic electroconvection. *Royal Society Open Science*, 8(8):202367, 2021. 30

M Guo and JS Hesthaven. Reduced order modeling for nonlinear structural analysis using Gaussian process regression. *Computer Methods in Applied Mechanics and Engineering*, 341: 807–826, 2018. 33, 41, 53

JH Halton. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numerische Mathematik*, 2(1):84–90, 1960. 85

K Hanjalić and BE Launder. A Reynolds stress model of turbulence and its application to thin shear flows. *Journal of Fluid Mechanics*, 52(4):609–638, 1972. 35

SR Hanna, MJ Brown, FE Camelli, ST Chan, WJ Coirier, OR Hansen, AH Huber, S Kim, and RM Reynolds. Detailed simulations of atmospheric flow and dispersion in downtown manhattan: An application of five computational fluid dynamics models. *Bulletin of the American Meteorological Society*, 87(12):1713–1726, 2006. 26

T Hastie, R Tibshirani, and JH Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, volume 2. Springer, 2009. 55

TM Heskes and B Kappen. On-line learning processes in artificial neural networks. In *North-Holland Mathematical Library*, volume 51, pages 199–233. Elsevier, 1993. 52

JS Hesthaven and S Ubbiali. Non-intrusive reduced order modeling of nonlinear problems using neural networks. *Journal of Computational Physics*, 363:55–78, 2018. 31, 33

S Hijazi, G Stabile, A Mola, and G Rozza. Data-driven POD-Galerkin reduced order model for turbulent flows. *Journal of Computational Physics*, 416:109513, 2020. 34

GE Hinton and R Zemel. Autoencoders, minimum description length and Helmholtz free energy. *Advances in Neural Information Processing Systems*, 6, 1993. 32

S Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6 (02):107–116, 1998. 51

NS Holmes and L Morawska. A review of dispersion modelling and its application to the dispersion of particles: An overview of different dispersion models available. *Atmospheric Environment*, 40(30):5902–5928, 2006. 20

PJ Huber. Robust statistics. In *International Encyclopedia of Statistical Science*, pages 1248–1251. Springer, 2011. 65

Y Jia, C Huang, and T Darrell. Beyond spatial pyramids: Receptive field learning for pooled image features. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3370–3377. IEEE, 2012. 50

IT Jolliffe. Principal component analysis. *Springer series in statistics*, 29, 2002. 94, 111

WP Jones and BE Launder. The prediction of laminarization with a two-equation model of turbulence. *International Journal of Heat and Mass Transfer*, 15(2):301–314, 1972. 23

L Kang, X Zhou, T van Hooff, B Blocken, and M Gu. CFD simulation of snow transport over flat, uniformly rough, open terrain: Impact of physical and computational parameters. *Journal of Wind Engineering and Industrial Aerodynamics*, 177:213–226, 2018. 24

K Kavukcuoglu, M Ranzato, and Y LeCun. Fast inference in sparse coding algorithms with applications to object recognition. *arXiv preprint arXiv:1010.3467*, 2010. 52

MC Kennedy and A O'Hagan. Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87(1):1–13, 2000. 151

A Kessy, A Lewin, and K Strimmer. Optimal whitening and decorrelation. *The American Statistician*, 72(4):309–314, 2018. 45

J-J Kim and J-J Baik. A numerical study of the effects of ambient wind direction on flow and dispersion in urban street canyons using the RNG k–ε turbulence model. *Atmospheric Environment*, 38(19):3039–3048, 2004. 17

DP Kingma, M Welling, et al. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392, 2019. 163, 169

DP Kingma, JA Ba, and J Adam. A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 106, 2020. 53

RH Kraichnan. Diffusion by a random velocity field. *Physics of Fluids*, 13(1):22–31, 1970. 74

SK Kumar. On weight initialization in deep neural networks. *arXiv preprint arXiv:1704.08863*, 2017. 52

G Lamberti and C Gorlé. A multi-fidelity machine learning framework to predict wind loads on buildings. *Journal of Wind Engineering and Industrial Aerodynamics*, 214:104647, 2021. 33

M Lange, H Suominen, M Kurppa, L Järvi, E Oikarinen, R Savvides, and K Puolamäki. Machine-learning models to replicate large-eddy simulations of air pollutant concentrations along boulevard-type streets. *Geoscientific Model Development*, 14(12):7411–7424, 2021. 29

CJ Lapeyre, A Misdariis, N Cazard, D Veynante, and T Poinsot. Training convolutional neural networks to estimate turbulent sub-grid scale reaction rates. *Combustion and Flame*, 203: 255–264, 2019. 36

BE Launder. Phenomenological modelling: Present.... and future? In *Whither Turbulence? Turbulence at the Crossroads*, pages 439–485. Springer, 1990. 35

ND Lawrence. Gaussian process latent variable models for visualisation of high dimensional data. *Advances in Neural Information Processing Systems*, 16, 2003. 32

ND Lawrence. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. *Journal of Machine Learning Research*, 6:1783–1816, 2005. 32, 181

Y Le Cun and F Fogelman-Soulié. Modèles connexionnistes de l'apprentissage. *Intellectica*, 2 (1):114–143, 1987. 32

L Le Gratiet. *Multi-fidelity Gaussian process regression for computer experiments*. PhD thesis, Université Paris-Diderot-Paris VII, 2013. 150, 151, 152, 163, 170

Y LeCun et al. Generalization and network design strategies. *Connectionism in perspective*, 19 (143-155):18, 1989. 32, 51

YA LeCun, L Bottou, GB Orr, and K-R Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–48. Springer, 2012. 51

Á Leelőssy, F Molnár, F Izsák, Á Havasi, I Lagzi, and R Mészáros. Dispersion modeling of air pollutants in the atmosphere: a review. *Open Geosciences*, 6(3):257–278, 2014. 12

BM Leitl and RN Meroney. Car exhaust dispersion in a street canyon. Numerical critique of a wind tunnel experiment. *Journal of Wind Engineering and Industrial Aerodynamics*, 67: 293–304, 1997. 17

C Lemieux. *Monte Carlo and Quasi-Monte Carlo Sampling*. Springer Series in Statistics. Springer New York, 2009. 84, 85

F Li, J Liu, J Ren, and X Cao. Predicting contaminant dispersion using modified turbulent Schmidt numbers from different vortex structures. *Building and Environment*, 130:120–127, 2018. 24

W-W Li and RN Meroney. Gas dispersion near a cubical model building. Part I. Mean concentration measurements. *Journal of Wind Engineering and Industrial Aerodynamics*, 12(1): 15–33, 1983a. 16, 19, 71, 81

W-W Li and RN Meroney. Gas dispersion near a cubical model building. Part II. Concentration fluctuation measurements. *Journal of Wind Engineering and Industrial Aerodynamics*, 12(1): 35–47, 1983b. 71

Y Li and T Stathopoulos. Numerical evaluation of wind-induced dispersion of pollutants around a building. *Journal of Wind Engineering and Industrial Aerodynamics*, 67:757–766, 1997. 16, 19, 71

DK Lilly. A proposed modification of the Germano subgrid-scale closure method. *Physics of Fluids A: Fluid Dynamics*, 4(3):633–635, 1992. 25

J Ling, R Jones, and J Templeton. Machine learning strategies for systems with invariance properties. *Journal of Computational Physics*, 318:22–35, 2016a. 36

J Ling, A Kurzawski, and J Templeton. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *Journal of Fluid Mechanics*, 807:155–166, 2016b. 36

J Ling, A Ruiz, G Lacaze, and J Oefelein. Uncertainty analysis and data-driven model advances for a jet-in-crossflow. *Journal of Turbomachinery*, 139(2):021008, 2017. 35

J-C Loiseau and SL Brunton. Constrained sparse galerkin regression. *Journal of Fluid Mechanics*, 838:42–67, 2018. 30, 34

P Louka, SE Belcher, and RG Harrison. Coupling between air flow in streets and the well-developed boundary layer aloft. *Atmospheric Environment*, 34(16):2613–2621, 2000. 16

D Lucor, A Agrawal, and A Sergent. Simple computational strategies for more effective physics-informed neural networks modeling of turbulent natural convection. *Journal of Computational Physics*, 456:111022, 2022. doi: 10.1016/j.jcp.2022.111022. 33

JL Lumley. Computational modeling of turbulent flows. *Advances in Applied Mechanics*, 18: 123–176, 1979. 35

D Ma, J Gao, Z Zhang, and H Zhao. Identifying atmospheric pollutant sources using a machine learning dispersion model and Markov chain Monte Carlo methods. *Stochastic Environmental Research and Risk Assessment*, 35(2):271–286, 2021. doi: 10.1007/s00477-021-01973-7. 33

L Margheri and P Sagaut. A hybrid anchored-ANOVA–POD/Kriging method for uncertainty quantification in unsteady high-fidelity CFD simulations. *Journal of Computational Physics*, 324:137–173, 2016. 3, 9, 26, 29, 31, 33, 37, 53, 79

D Martin, CS Price, IR White, G Nickless, A Dobre, and DE Shallcross. A study of pollutant concentration variability in an urban street under low wind speeds. *Atmospheric Science Letters*, 9(3):147–152, 2008. 18

R Martinuzzi and C Tropea. The flow around surface-mounted, prismatic obstacles placed in a fully developed channel flow (data bank contribution). *ASME Journal of Fluids Engineering*, 115(1):85–92, 1993. 16, 77

R Maulik, O San, A Rasheed, and P Vedula. Subgrid modelling for two-dimensional turbulence using neural networks. *Journal of Fluid Mechanics*, 858:122–144, 2019. 36

I Mavroidis and RF Griffiths. Investigation of building—influenced atmospheric dispersion using a dual source technique. In *Urban Air Quality: Measurement, Modelling and Management*, pages 239–247. Springer, 2000. 81

I Mavroidis, RF Griffiths, and DJ Hall. Field and wind tunnel investigations of plume dispersion around single surface obstacles. *Atmospheric Environment*, 37(21):2903–2918, 2003. 82

M Mendil, S Leirens, P Armand, and C Duchenne. Hazardous atmospheric dispersion in urban areas: A Deep Learning approach for emergency pollution forecast. *Environmental Modelling & Software*, 152:105387, 2022. 29

RN Meroney. Ten questions concerning hybrid computational/physical model simulation of wind flow in the built environment. *Building and Environment*, 96:12–21, 2016. 22

RN Meroney, BM Leitl, S Rafailidis, and M Schatzmann. Wind-tunnel and numerical modeling of flow and dispersion about several building shapes. *Journal of Wind Engineering and Industrial Aerodynamics*, 81(1-3):333–345, 1999. 16, 19

M Milano and P Koumoutsakos. Neural network modeling for near wall turbulent flow. *Journal of Computational Physics*, 182(1):1–26, 2002. 32

M Milliez and B Carissimo. Numerical simulations of pollutant dispersion in an idealized urban area, for different meteorological conditions. *Boundary-Layer Meteorology*, 122(2):321–342, 2007. doi: 10.1007/s10546-006-9110-4. 17, 35

AA Mishra and G Iaccarino. Uncertainty estimation for Reynolds-Averaged Navier-Stokes predictions of high-speed aircraft nozzle jets. *AIAA Journal*, 55(11):3999–4004, 2017. 36

V Mnih, K Kavukcuoglu, D Silver, A Graves, I Antonoglou, D Wierstra, and M Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013. 31

A Mochida, Y Tominaga, S Murakami, R Yoshie, T Ishihara, and R Ooka. Comparison of various k-ε models and DSM applied to flow around a high-rise building-report on AIJ cooperative project for CFD prediction of wind environment. *Wind and Structures*, 5(2_3_4):227–244, 2002. 26

Pablo Moreno-Muñoz, Antonio Artés, and Mauricio Álvarez. Modular gaussian processes for transfer learning. *Advances in Neural Information Processing Systems*, 34:24730–24740, 2021. 163, 170

PC Mukesh Kumar and R Kavitha. Prediction of nanofluid viscosity using multilayer perceptron and Gaussian process regression. *Journal of Thermal Analysis and Calorimetry*, 144(4):1151–1160, 2021. 62

S Murakami. Comparison of various turbulence models applied to a bluff body. In *Computational Wind Engineering*, pages 21–36. Elsevier, 1993. 19, 26

S Murakami. Current status and future trends in computational wind engineering. *Journal of Wind Engineering and Industrial Aerodynamics*, 67:3–34, 1997. 24, 26

S Murakami. Overview of turbulence models applied in CWE–1997. *Journal of Wind Engineering and Industrial Aerodynamics*, 74:1–24, 1998. 22, 24

S Murakami and A Mochida. Three-dimensional numerical simulation of turbulent flow around buildings using the k- ε turbulence model. *Building and Environment*, 24(1):51–64, 1989. 26

S Murakami, A Mochida, and Y Hayashi. Examining the $\kappa - \epsilon$ model by means of a wind tunnel test and large-eddy simulation of the turbulence structure around a cube. *Journal of Wind Engineering and Industrial Aerodynamics*, 35:87–100, 1990a. 26

S Murakami, A Mochida, Y Hayashi, and K Hibi. Numerical simulation of velocity field and diffusion field in an urban area. *Energy and Buildings*, 15(3-4):345–356, 1990b. 26

S Murakami, A Mochida, Y Hayashi, and S Sakamoto. Numerical study on velocity-pressure field and wind forces for bluff bodies by κ-ϵ, ASM and LES. *Journal of Wind Engineering and Industrial Aerodynamics*, 44(1-3):2841–2852, 1992. 26

T Murata, K Fukami, and K Fukagata. Nonlinear mode decomposition with convolutional neural networks for fluid dynamics. *Journal of Fluid Mechanics*, 882, 2020. 32, 33, 120, 126, 128

YE Nesterov. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. In *Dokl. akad. nauk Sssr*, volume 269, pages 543–547, 1983. 53

PV Nielsen, F Allard, HB Awbi, L Davidson, and A Schälin. Computational fluid dynamics in ventilation design REHVA guidebook No 10, 2007. 24

BX Nony, MC Rochoux, T Jaravel, and D Lucor. Reduced-order modeling for parameterized large-eddy simulations of atmospheric pollutant dispersion, 2022. URL `https://arxiv.org/abs/2208.01518`. 128

AM Obukhov. Turbulence in an atmosphere with a non-uniform temperature. *Boundary-layer meteorology*, 2(1):7–29, 1971. 15

H Ogura. Orthogonal functionals of the Poisson process. *IEEE Transactions on Information Theory*, 18(4):473–481, 1972. 58

TR Oke. Street design and urban canopy layer climate. *Energy and Buildings*, 11(1-3):103–113, 1988. 17

TR Oke. Urban environments. *The Surface Climates of Canada*, pages 303–327, 1997. 14, 175

TR Oke. *Boundary Layer Climates*. Routledge, 2002. 13, 181

TR Oke, G Mills, A Christen, and JA Voogt. *Urban Climates*. Cambridge University Press, 2017. 13

GB Orr. *Dynamics and algorithms for stochastic learning*. PhD thesis, Department of Computer Science and Engineering, Oregon Graduate Institute of Science & Technology, 1995. 52

R Paoli, A Poubeau, and D Cariolle. Large-eddy simulations of a reactive solid rocket motor plume. *AIAA Journal*, 58(4):1639–1656, 2020. 72

EJ Parish and K Duraisamy. A paradigm for data-driven predictive modeling using field inversion and machine learning. *Journal of Computational Physics*, 305:758–774, 2016. 35, 134

F Pasquill. The estimation of the dispersion of windborne material. *Meteorology Magazine*, 90: 33, 1961. 15

T Passot and A Pouquet. Numerical simulation of compressible homogeneous flows in the turbulent regime. *Journal of Fluid Mechanics*, 181:441–466, 1987. 75

DA Paterson and CJ Apelt. Simulation of flow past a cube in a turbulent boundary layer. *Journal of Wind Engineering and Industrial Aerodynamics*, 35:149–176, 1990. 26

F Pedregosa, G Varoquaux, A Gramfort, V Michel, B Thirion, O Grisel, M Blondel, P Prettenhofer, R Weiss, V Dubourg, J Vanderplas, A Passos, D Cournapeau, M Brucher, M Perrot, and E Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. 87

DA Philips. *Modeling scalar dispersion in urban environments*. Stanford University, 2012. 18, 21, 78, 175

DA Philips, R Rossi, and G Iaccarino. Large-eddy simulation of passive scalar dispersion in an urban-like canopy. *Journal of Fluid Mechanics*, 723:404–428, 2013. 12, 17, 26, 30, 175

A Pollard, TJ Hacker, and S Dyke. Whither turbulence and big data for the twenty-first century. In *Whither Turbulence and Big Data in the 21st Century?*, pages 551–574. Springer, 2017. 27

BT Polyak. Some methods of speeding up the convergence of iteration methods. *USSR computational mathematics and mathematical physics*, 4(5):1–17, 1964. 53

SB Pope. *Turbulent Flows*. Cambridge University Press, 2000. 21, 25, 35, 73

A Poubeau, R Paoli, and D Cariolle. Evaluation of afterburning chemistry in solid-rocket motor jets using an off-line model. *Journal of Spacecraft and Rockets*, 53(2):380–388, 2016. 72

M Raissi and G Karniadakis. Deep multi-fidelity gaussian processes. *arXiv preprint arXiv:1604.07484*, 2016. 163, 170

JD Ramshaw, PJ O'Rourke, and LR Stein. Pressure gradient scaling method for fluid flow with nearly uniform pressure. *Journal of Computational Physics*, 58(3):361–376, 1985. 72

CE Rasmussen and CK Williams. *Gaussian Processes for Machine Learning*, volume 2. MIT press Cambridge, MA, USA, 2006. 60, 61, 63, 64

S Rezaeiravesh, R Vinuesa, and P Schlatter. On numerical uncertainties in scale-resolving simulations of canonical wall turbulence. *Computers & Fluids*, 227:105024, 2021. 36

PJ Richards and RP Hoxey. Appropriate boundary conditions for computational wind engineering models using the k-ϵ turbulence model. *Journal of Wind Engineering and Industrial Aerodynamics*, 46-47:145–153, 1993. 75

LF Richardson. I. Some measurements of atmospheric turbulence. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 221(582-593):1–28, 1921. 15

M Riedmiller and H Braun. A direct adaptive method for faster backpropagation learning: The RPROP algorithm. In *IEEE international conference on neural networks*, pages 586–591. IEEE, 1993. 53

S Rifai, P Vincent, X Muller, X Glorot, and Y Bengio. Contractive auto-encoders: Explicit invariance during feature extraction. In L Getoor and T Scheffer, editors, *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 833–840, New York, NY, USA, June 2011. ACM. 52

MK Rivera, WB Daniel, SY Chen, and RE Ecke. Energy and enstrophy transfer in decaying two-dimensional turbulence. *Physical Review Letters*, 90:104502, 2003. 73

MC Rochoux, S Ricci, D Lucor, B Cuenot, and A Trouvé. Towards predictive data-driven simulations of wildfire spread – Part I: Reduced-cost Ensemble Kalman Filter based on a Polynomial Chaos surrogate model for parameter estimation. *Natural Hazards and Earth System Sciences*, 14(11):2951–2973, 2014. 33

W Rodi. Comparison of les and rans calculations of the flow around bluff bodies. *Journal of wind engineering and industrial aerodynamics*, 69:55–75, 1997. 134

RS Rogallo and P Moin. Numerical simulation of turbulent flows. *Annual Review of Fluid Mechanics*, 16(1):99–137, 1984. 27

R Rossi. A numerical study of algebraic flux models for heat and mass transport simulation in complex flows. *International Journal of Heat and Mass Transfer*, 53(21):4511–4524, 2010. 135

R Rossi, DA Philips, and G Iaccarino. A numerical study of scalar dispersion downstream of a wall-mounted cube using direct simulations and algebraic flux models. *International Journal of Heat and Fluid Flow*, 31(5):805–819, 2010. 135

MW Rotach, S-E Gryning, E Batchvarova, A Christen, and R Vogt. Pollutant dispersion close to an urban surface–the bubble tracer experiment. *Meteorology and Atmospheric Physics*, 87 (1):39–56, 2004. 18

D Ruan, H He, DA Castañón, and KC Mehta. Normalized proper orthogonal decomposition (npod) for building pressure data compression. *Journal of wind engineering and industrial aerodynamics*, 94(6):447–461, 2006. 162, 169

PJ Saathof, T Stathopoulos, and M Dobrescu. Effects of model scale in estimating pollutant dispersion near buildings. *Journal of Wind Engineering and Industrial Aerodynamics*, 54: 549–559, 1995. 19

P Saathoff, T Stathopoulos, and H Wu. The influence of freestream turbulence on nearfield dilution of exhaust from building vents. *Journal of Wind Engineering and Industrial Aerodynamics*, 77:741–752, 1998. 19

T Sabatier, C Sarrat, S Aubry, and T Chaboud. Quantification of the airport-related pollution under wintertime anticyclonic conditions from idealized large-eddy simulations. *Atmospheric Environment*, 262:118619, 2021. 164, 171

R Saegusa, H Sakano, and S Hashimoto. Nonlinear principal component analysis to preserve the order of principal components. *Neurocomputing*, 61:57–70, 2004. 33, 126

I Scherl, B Strom, JK Shang, O Williams, BL Polagye, and SL Brunton. Robust principal component analysis for modal decomposition of corrupt fluid flows. *Physical Review Fluids*, 5 (5):054401, 2020. 32

PJ Schmid. Dynamic mode decomposition and its variants. *Annual Review of Fluid Mechanics*, 54:225–254, 2022. 32

B Schölkopf, A Smola, and K-R Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998. 32

T Schönfeld and M Rudgyard. Steady and unsteady flow simulations using the hybrid flow solver avbp. *AIAA Journal*, 37(11):1378–1385, 1999. 72

JH Seinfeld. Atmospheric chemistry and physics of air pollution. *Environmental Science & Technology*, 20(9):863–863, 1986. 15, 175

L Sirovich. Turbulence and the dynamics of coherent structures. I - Coherent structures. II - Symmetries and transformations. III - Dynamics and scaling. *Quarterly of Applied Mathematics*, 45:561–571, 1987. 31, 43

J Smagorinsky. General circulation experiments with the primitive equations: I. The basic experiment. *Monthly Weather Review*, 91(3):99–164, 1963. 25

H Sompolinsky. On-line learning of dichotomies: algorithms and learning curves. *Neural Networks: The Statistical Mechanics Perspective*, pages 105–130, 1995. 52

J Sousa and C Gorlé. Computational urban flow predictions with bayesian inference: Validation with field data. *Building and Environment*, 154:13–22, 2019. 37, 74

P Spalart and S Allmaras. *A one-equation turbulence model for aerodynamic flows*. 1994. 23, 35, 133

CG Speziale. Analytical methods for the development of Reynolds-stress closures in turbulence. *Annual Review of Fluid Mechanics*, 23(1):107–157, 1991. 35

N Srivastava, G Hinton, A Krizhevsky, I Sutskever, and R Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15 (1):1929–1958, 2014. 52

T Stathopoulos. The numerical wind tunnel for industrial aerodynamics: Real or virtual in the new millennium? *Wind and Structures*, 5(2_3_4):193–208, 2002. 19

T Stathopoulos and A Baskaran. Boundary treatment for the computation of three-dimensional wind flow conditions around a building. *Journal of Wind Engineering and Industrial Aerodynamics*, 35:177–200, 1990. 26

T Stathopoulos and A Baskaran. Computer simulation of wind environmental conditions around buildings. *Engineering Structures*, 18(11):876–885, 1996. 26

Julia Steiner, Axelle Viré, and Richard P Dwight. Classifying regions of high model error within a data-driven rans closure: Application to wind turbine wakes. *Flow, Turbulence and Combustion*, 109(3):545–570, 2022. 134

JM Stockie. The mathematics of atmospheric dispersion modeling. *SIAM Review*, 53(2):349–372, 2011. 20

I Sutskever, J Martens, G Dahl, and G Hinton. On the importance of initialization and momentum in deep learning. In *International Conference on Machine Learning*, pages 1139–1147, 2013. 53

OG Sutton. The problem of diffusion in the lower atmosphere. *Quarterly Journal of the Royal Meteorological Society*, 73(317-318):257–281, 1947. 20

RS Sutton. Adapting bias by gradient descent: An incremental version of delta-bar-delta. In *AAAI*, pages 171–176. San Jose, CA, USA, 1992. 52

RS Sutton and AG Barto. *Reinforcement Learning: An Introduction*. MIT press, 2018. 31

Laura P Swiler, Mamikon Gulian, Ari L Frankel, Cosmin Safta, and John D Jakeman. A survey of constrained gaussian process regression: Approaches and implementation challenges. *Journal of Machine Learning for Modeling and Computing*, 1(2), 2020. 163, 169

R Swischuk, L Mainini, B Peherstorfer, and K Willcox. Projection-based model reduction: Formulations for physics-based machine learning. *Computers & Fluids*, 179:704–717, 2019. 31, 33, 34

K Taira, SL Brunton, STM Dawson, CW Rowley, T Colonius, BJ McKeon, OT Schmidt, S Gordeyev, V Theofilis, and LS Ukeiley. Modal analysis of fluid flows: An overview. *AIAA Journal*, 55(12):4013–4041, 2017. 30

K Taira, MS Hemati, SL Brunton, Y Sun, K Duraisamy, S Bagheri, STM Dawson, and C-A Yeh. Modal analysis of fluid flows: Applications and outlook. *AIAA Journal*, 58(3):998–1022, 2020. 30

R Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996. 52, 60

T Tieleman, G Hinton, et al. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, 4(2):26–31, 2012. 53

M Titsias and ND Lawrence. Bayesian Gaussian process latent variable model. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 844–851. JMLR Workshop and Conference Proceedings, 2010. 32

F Toja-Silva, J Chen, S Hachinger, and F Hase. CFD simulation of CO2 dispersion from urban thermal power plant: Analysis of turbulent Schmidt number and comparison with Gaussian plume model and measurements. *Journal of Wind Engineering and Industrial Aerodynamics*, 169:177–193, 2017. 24

Y Tominaga and T Stathopoulos. Turbulent Schmidt numbers for CFD analysis with various types of flowfield. *Atmospheric Environment*, 41(37):8091–8099, 2007. 24

Y Tominaga and T Stathopoulos. Numerical simulation of plume dispersion around an isolated cubic buildings: comparisons between RANS and LES computations. In *BBAA VI International Colloquium on: Bluff Bodies Aerodynamics & Applications*, pages 20–24, 2008. 16, 19

Y Tominaga and T Stathopoulos. Numerical simulation of dispersion around an isolated cubic building: comparison of various types of k-ϵ models. *Atmospheric Environment*, 43(20):3200–3210, 2009. 24, 71

Y Tominaga and T Stathopoulos. Numerical simulation of dispersion around an isolated cubic building: model evaluation of RANS and LES. *Building and Environment*, 45(10):2231–2239, 2010. 24, 26, 71

Y Tominaga and T Stathopoulos. CFD modeling of pollution dispersion in building array: evaluation of turbulent scalar flux modeling in RANS model using LES results. *Journal of Wind Engineering and Industrial Aerodynamics*, 104:484–491, 2012. 26

Y Tominaga and T Stathopoulos. CFD simulation of near-field pollutant dispersion in the urban environment: A review of current modeling techniques. *Atmospheric Environment*, 79: 716–730, 2013. 1, 7, 24

Y Tominaga, S Murakami, and A Mochida. CFD prediction of gaseous diffusion around a cubic model using a dynamic mixed SGS model based on composite grid technique. *Journal of Wind Engineering and Industrial Aerodynamics*, 67:827–841, 1997. 16, 19

B Tracey, K Duraisamy, and J Alonso. Application of supervised learning to quantify uncertainties in turbulence and combustion modeling. In *51st AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition*, page 259, 2013. 35

BD Tracey, K Duraisamy, and JJ Alonso. A machine learning strategy to assist turbulence model development. In *53rd AIAA aerospace sciences meeting*, page 1287, 2015. 35

P Tseng. An incremental gradient (-projection) method with momentum term and adaptive stepsize rule. *SIAM Journal on Optimization*, 8(2):506–531, 1998. 53

G Turbelin. *Modélisation de la turbulence atmosphérique en vue de l'étude du chargement aérodynamique des structures soumises aux effets du vent*. Theses, Université d'Evry Val d'Essonne, 2000. 16, 175

DB Turner. *Workbook of Atmospheric Dispersion Estimates: an Introduction to Dispersion Modeling*. CRC press, 2020. 20

VN Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999. 55

L Vervecken, J Camps, and J Meyers. Accounting for wind-direction fluctuations in Reynolds-averaged simulation of near-range atmospheric dispersion. *Atmospheric Environment*, 72: 142–150, 2013. 26, 29

L Vervecken, J Camps, and J Meyers. Dynamic dose assessment by Large Eddy Simulation of the near-range atmospheric dispersion. *Journal of Radiological Protection*, 35(1):165–178, jan 2015a. 26

L Vervecken, J Camps, and J Meyers. Stable reduced-order models for pollutant dispersion in the built environment. *Building and Environment*, 92:360–367, 2015b. 29

R Vinuesa and SL Brunton. Enhancing computational fluid dynamics with machine learning. *Nature Computational Science*, 2:358–366, 2022. 28, 29, 34, 175

J-Y Vinçont, S Simoënsn, M Ayrault, and JM Wallace. Passive scalar dispersion in a turbulent boundary layer from a line source at the wall and downstream of an obstacle. *Journal of Fluid Mechanics*, 424:127–167, 2000. 19, 76, 77, 78, 137

D Wackerly, W Mendenhall, and RL Scheaffer. *Mathematical Statistics with Applications*. Cengage Learning, 2014. 91

S Wallin and AV Johansson. An explicit algebraic Reynolds stress model for incompressible and compressible turbulent flows. *Journal of Fluid Mechanics*, 403:89–132, 2000. 35

R Wang, K Kashinath, M Mustafa, A Albert, and R Yu. Towards physics-informed deep learning for turbulent flow prediction. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1457–1466, 2020. 30

Y-X Wang and Y-J Zhang. Nonnegative matrix factorization: A comprehensive review. *IEEE Transactions on Knowledge and Data Engineering*, 25(6):1336–1353, 2012. 32

N Wiener. The homogeneous chaos. *American Journal of Mathematics*, 60(4):897–936, 1938. 57

J Wiernga. Representative roughness parameters for homogeneous terrain. *Boundary-Layer Meteorology*, 63(4):323–363, 1993. 16

DC Wilcox. Formulation of the k-ω turbulence model revisited. *AIAA Journal*, 46(11):2823–2838, 2008. 23, 35

S Xiang, X Fu, J Zhou, Y Wang, Y Zhang, X Hu, J Xu, H Liu, J Liu, J Ma, et al. Non-intrusive reduced order model of urban airflow with dynamic boundary conditions. *Building and Environment*, 187:107397, 2021. 33, 53

D Xiao, CE Heaney, F Fang, L Mottet, R Hu, DA Bistrian, E Aristodemou, IM Navon, and CC Pain. A domain decomposition non-intrusive reduced order model for turbulent flows. *Computers & Fluids*, 182:15–27, 2019. 31, 33, 53

H Xiao and P Cinnella. Quantification of model uncertainty in RANS simulations: A review. *Progress in Aerospace Sciences*, 108:1–31, 2019. 22, 175

D Xiu and GE Karniadakis. The Wiener–Askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, 24(2):619–644, 2002. 58

D Xiu and GE Karniadakis. Modeling uncertainty in flow simulations via generalized polynomial chaos. *Journal of Computational Physics*, 187(1):137–167, 2003. 58, 181

R Yoshie, A Mochida, Y Tominaga, H Kataoka, K Harimoto, T Nozu, and T Shirasawa. Co-operative project for CFD prediction of pedestrian wind environment in the Architectural Institute of Japan. *Journal of Wind Engineering and Industrial Aerodynamics*, 95(9-11): 1551–1578, 2007. 26

A Yoshizawa, H Abe, Y Matsuo, H Fujiwara, and Y Mizobuchi. A Reynolds-averaged turbulence modeling approach using three transport equations for the turbulent viscosity, kinetic energy, and dissipation rate. *Physics of Fluids*, 24(7):075109, 2012. 133, 135, 140