N°d'ordre NNT : 2022LYSEI007

THESE de DOCTORAT DE L'UNIVERSITE DE LYON
opérée au sein de
L'INSA de Lyon

École Doctorale ED512
InfoMaths

Spécialité / discipline de doctorat :
Informatique

Soutenue publiquement le 04/02/2022, par :
Mohamed Anis Fekih

Low-cost Wireless Sensor Networks in Participatory Air Quality Monitoring

---

Devant le jury composé de :

MITTON, Nathalie	Directrice de recherche, INRIA	Rapportrice
MONTAVONT, Nicolas	Professeur des universités, IMT Atlantique	Rapporteur
GUITTON, Alexandre	Professeur des universités, Univ Clermont Auvergne	Examinateur
REDON, Nathalie	Maître de conférences, IMT Nord Europe	Examinatrice
RIVANO, Hervé	Professeur des universités, INSA-Lyon	Directeur de thèse
BECHKIT, Walid	Maître de conférences, INSA-Lyon	Co-directeur de thèse

Département FEDORA – INSA Lyon - Ecoles Doctorales

SIGLE	ECOLE DOCTORALE	NOM ET COORDONNEES DU RESPONSABLE
CHIMIE	**CHIMIE DE LYON** **https://www.edchimie-lyon.fr** Sec. : Renée EL MELHEM Bât. Blaise PASCAL, 3e étage secretariat@edchimie-lyon.fr	**M. Stéphane DANIELE** C2P2-CPE LYON-UMR 5265 Bâtiment F308, BP 2077 43 Boulevard du 11 novembre 1918 69616 Villeurbanne **directeur@edchimie-lyon.fr**
E.E.A.	**ÉLECTRONIQUE, ÉLECTROTECHNIQUE, AUTOMATIQUE** **https://edeea.universite-lyon.fr** Sec. : Stéphanie CAUVIN Bâtiment Direction INSA Lyon Tél : 04.72.43.71.70 secretariat.edeea@insa-lyon.fr	**M. Philippe DELACHARTRE** INSA LYON Laboratoire CREATIS Bâtiment Blaise Pascal, 7 avenue Jean Capelle 69621 Villeurbanne CEDEX Tél : 04.72.43.88.63 **philippe.delachartre@insa-lyon.fr**
E2M2	**ÉVOLUTION, ÉCOSYSTÈME, MICROBIOLOGIE, MODÉLISATION** **http://e2m2.universite-lyon.fr** Sec. : Sylvie ROBERJOT Bât. Atrium, UCB Lyon 1 Tél : 04.72.44.83.62 secretariat.e2m2@univ-lyon1.fr	**M. Philippe NORMAND** Université Claude Bernard Lyon 1 UMR 5557 Lab. d'Ecologie Microbienne Bâtiment Mendel 43, boulevard du 11 Novembre 1918 69 622 Villeurbanne CEDEX **philippe.normand@univ-lyon1.fr**
EDISS	**INTERDISCIPLINAIRE SCIENCES-SANTÉ** **http://ediss.universite-lyon.fr** Sec. : Sylvie ROBERJOT Bât. Atrium, UCB Lyon 1 Tél : 04.72.44.83.62 secretariat.ediss@univ-lyon1.fr	**Mme Sylvie RICARD-BLUM** Institut de Chimie et Biochimie Moléculaires et Supramoléculaires (ICBMS) - UMR 5246 CNRS - Université Lyon 1 Bâtiment Raulin - 2ème étage Nord 43 Boulevard du 11 novembre 1918 69622 Villeurbanne Cedex Tél : +33(0)4 72 44 82 32 **sylvie.ricard-blum@univ-lyon1.fr**
INFOMATHS	**INFORMATIQUE ET MATHÉMATIQUES** **http://edinfomaths.universite-lyon.fr** Sec. : Renée EL MELHEM Bât. Blaise PASCAL, 3e étage Tél : 04.72.43.80.46 infomaths@univ-lyon1.fr	**M. Hamamache KHEDDOUCI** Université Claude Bernard Lyon 1 Bât. Nautibus 43, Boulevard du 11 novembre 1918 69 622 Villeurbanne Cedex France Tél : 04.72.44.83.69 **hamamache.kheddouci@univ-lyon1.fr**
Matériaux	**MATÉRIAUX DE LYON** **http://ed34.universite-lyon.fr** Sec. : Yann DE ORDENANA Tél : 04.72.18.62.44 yann.de-ordenana@ec-lyon.fr	**M. Stéphane BENAYOUN** Ecole Centrale de Lyon Laboratoire LTDS 36 avenue Guy de Collongue 69134 Ecully CEDEX Tél : 04.72.18.64.37 **stephane.benayoun@ec-lyon.fr**
MEGA	**MÉCANIQUE, ÉNERGÉTIQUE, GÉNIE CIVIL, ACOUSTIQUE** **http://edmega.universite-lyon.fr** Sec. : Stéphanie CAUVIN Tél : 04.72.43.71.70 Bâtiment Direction INSA Lyon mega@insa-lyon.fr	**M. Jocelyn BONJOUR** INSA Lyon Laboratoire CETHIL Bâtiment Sadi-Carnot 9, rue de la Physique 69621 Villeurbanne CEDEX **jocelyn.bonjour@insa-lyon.fr**
ScSo	**ScSo*** **https://edsciencessociales.universite-lyon.fr** Sec. : Mélina FAVETON INSA : J.Y. TOUSSAINT Tél : 04.78.69.77.79 melina.faveton@univ-lyon2.fr	**M. Christian MONTES** Université Lumière Lyon 2 86 Rue Pasteur 69365 Lyon CEDEX 07 **christian.montes@univ-lyon2.fr**

*ScSo : Histoire, Géographie, Aménagement, Urbanisme, Archéologie, Science politique, Sociologie, Anthropologie

Acknowledgments

This Ph.D. thesis is the core of the 3M'Air multidisciplinary project funded by the "LabEx IMU (Intelligences des Mondes Urbains)" (ANR- 10-LABX-0088) of the University of Lyon, within the program "Investissements d'Avenir" (ANR-11-IDEX-0007) operated by the French National Research Agency (ANR). This thesis was supervised by Professors Walid Bechkit and Hervé Rivano within the AGORA team of the CITI-lab of INSA-Lyon.

Abstract

Mobile crowdsensing is an emerging and promising paradigm that has attracted much attention in recent years, especially for environmental monitoring. Coupled with the power of low-cost wireless sensor networks (WSN), it leverages population density to collect extensive data in many applications, such as air pollution and urban heat islands (UHI) monitoring. In fact, air pollution and UHI are one of the main problems that still suffer from a lack of characterization due to the limitations of traditional assessment methods employed in terms of cost, network size, and flexibility. Mobile crowdsensing and WSN aim at filling this gap by enabling large-scale deployments to improve the local knowledge of these phenomena on the one hand, while simultaneously involving the citizens in the process on the other hand.

In this thesis, we mainly consider the air quality monitoring application with a mobile crowdsensing approach, while focusing on three main parts: 1) the design of low-cost participatory air quality monitoring systems; 2) the analysis of dense data from low-cost WSNs and their contribution to the fine-grained mapping of air quality; 3) the selection of the participants' paths in order to improve the knowledge of the phenomenon while taking into account the constraints of travel distance and sensor errors. Through this work, we aim to show the potential of using low-cost WSN coupled with participatory sensing for air quality monitoring. In this vein, we carry out substantial experimental work on the design of a participatory air quality monitoring system from scratch. We provide engineering guidelines regarding the design of low-cost participatory environmental monitoring platforms. Moreover, we conduct extensive validation tests to evaluate the performance of our sensor nodes. In addition, we perform analysis on our sensors' data and propose a general framework that allows the comparison of different regression and data assimilation strategies, based on numerical simulations and an adequate estimation of simulation and sensing error covariances. We also explore the impact of the sensing rate on the energy consumption and the mapping error. Furthermore, we tackle the problem of route selection in participatory sensing and propose two new approaches that take into account the participants' constraints and the characteristics of air quality monitoring using low-cost WSN.

Résumé

La mesure mobile par la foule (aussi appelé *mobile crowdsensing*) est un paradigme émergent et prometteur qui a attiré beaucoup d'attention ces dernières années, notamment dans le domaine de la surveillance de l'environnement. Couplé à la puissance des réseaux de capteurs sans fil (RCSF) à bas coût, il permet de tirer parti de la densité de la population pour collecter de nombreuses données dans de nombreuses applications, telles que la surveillance de la pollution de l'air et des îlots de chaleur urbains (ICU). En effet, la pollution de l'air et les ICUs sont parmi les principaux problèmes qui souffrent encore d'un manque de caractérisation en raison des limites des méthodes d'évaluation traditionnelles en termes de coût, de taille de réseau et de flexibilité. Le *mobile crowdsensing* et les RCSFs visent à combler cette lacune en permettant des déploiements à grande échelle afin d'améliorer la connaissance locale du phénomène, tout en impliquant les citoyens dans le processus de suivi de celui-ci.

Dans cette thèse, nous considérons l'application de surveillance de la qualité de l'air avec une approche de crowdsensing mobile, tout en nous concentrant sur trois axes principaux : 1) la conception de systèmes de surveillance de la qualité de l'air participatifs et à faible coût ; 2) l'analyse de données denses issues de micro-capteurs à bas coût et leur apport à la cartographie fine de la qualité de l'air ; 3) la sélection des chemins des participants afin d'améliorer la connaissance du phénomène tout en prenant en compte différentes contraintes. À travers ce travail, nous souhaitons montrer le potentiel de l'utilisation de RCSF à faible coût couplé à la mesure participative dans la surveillance de la qualité de l'air. Dans cette optique, nous réalisons un travail expérimental poussé sur la conception d'un système participatif de surveillance de la qualité de l'air. Nous fournissons des recommandations d'ingénierie concernant la conception de plateformes de surveillance environnementale participative à faible coût. En outre, nous effectuons des tests de validation approfondis pour évaluer les performances de nos nœuds de capteurs. De plus, nous analysons les données collectées par nos capteurs et proposons un framework général qui permet de comparer différentes stratégies de régression et d'assimilation de données, à l'aide de simulations numériques et d'une estimation adéquate des covariances des erreurs de simulation et de mesure. Nous explorons également l'impact de la fréquence de mesure sur la consommation d'énergie et l'erreur de cartographie. Enfin, nous nous intéressons au problème de sélection de routes dans le cadre de la mesure participative et proposons deux nouvelles approches qui prennent en compte les contraintes des participants et les caractéristiques de la surveillance de la qualité de l'air à l'aide de RCSF à faible coût.

Résumé Long

La pollution de l'air représente une préoccupation majeure dans plusieurs villes du monde. En effet, les concentrations de polluants dépassent les limites sanitaires standards dans de nombreuses villes plusieurs fois par an. La pollution de l'air est formée de produits chimiques et de particules rejetés dans l'atmosphère qui constituent de graves menaces pour la santé et l'environnement. Les effets néfastes de ce phénomène ont été largement rapportés dans de nombreuses études. Les polluants sont suffisamment petits pour franchir facilement les barrières protectrices du corps humain. Par conséquent, l'exposition à de fortes concentrations de polluants pendant une période prolongée peut entraîner des maladies cardiaques, une réduction de la fonction pulmonaire, des infections respiratoires, ainsi que des diabètes, ce qui se traduit par une augmentation de la mortalité dans les villes polluées. Selon les estimations de l'Organisation Mondiale de la Santé (OMS), des millions de décès par an sont causés par la pollution de l'air intérieure et extérieure. En outre, les conséquences nocives de la pollution de l'air ne se limitent pas à la santé humaine, mais affectent également l'environnement. Des niveaux excessifs d'oxydes d'azote, par exemple, contribuent à l'acidification des sols, modifient le taux de croissance des plantes et, par conséquent, affectent la biodiversité.

De plus, l'urbanisation continue des villes à travers le monde contribue à la suppression des surfaces de végétation au profit des bâtiments et des routes pavées. Par conséquent, la chaleur reste piégée entre les bâtiments et entraîne une différence de température significative entre les zones urbaines et les zones rurales environnantes, créant un phénomène connu sous le nom d'îlot de chaleur urbain (ICU). Les îlots de chaleur urbains sont influencés par plusieurs facteurs, tels que la taille et la structure des bâtiments d'une ville. Ils peuvent poser de graves problèmes de santé, car ils augmentent considérablement le risque d'allergies, de problèmes respiratoires et cardiovasculaires et peuvent entraîner une surmortalité chez les personnes vulnérables pendant les vagues de chaleur intense.

Afin d'atténuer l'impact de la pollution de l'air et des îlots de chaleur urbains, les autorités régionales et gouvernementales ont fait des efforts considérables pour mettre en place des méthodes de surveillance de la qualité de l'air. Des stations de surveillance de la qualité de l'air ont été installées dans plusieurs villes dans le but de mesurer et suivre avec une grande précision de nombreux polluants tels que le monoxyde de carbone (CO), le dioxyde d'azote (NO_2), l'ozone (O_3) et les particules fines (PM_1, $PM_{2.5}$, PM_{10}), ainsi que les paramètres météorologiques tels que la température, l'humidité, la vitesse du vent, etc. Bien qu'elles soient très précises, ces stations traditionnelles sont déployées en nombre très réduit en raison de leur grande taille, de leur manque de flexibilité et de leur coût d'entretien très élevé. Outre ces stations, d'autres méthodes de suivi de la qualité de l'air ont été envisagées, comme les modèles physicochimiques qui modélisent la propagation de la pollution de l'air au moyen de simulations numériques afin de générer une carte de la pollution. Cepen-

dant, il est difficile de parvenir à une connaissance fine en utilisant uniquement ces modèles, car les données d'entrée sont généralement moyennées dans l'espace et dans le temps.

Les limitations des méthodes traditionnelles de suivi en termes de coût, de taille, de flexibilité et de granularité spatiale ont conduit à l'émergence de petits capteurs environnementaux à faible coût. La connexion sans fil de ces capteurs, formant un réseau de capteurs sans fil (RCSF) à faible coût, présente de nombreux avantages par rapport aux solutions traditionnelles de surveillance de la qualité de l'air. En effet, leur rapport coût-efficacité permet des déploiements denses et améliore donc la granularité spatiale et temporelle. De plus, les considérations de taille et de coût offrent une plus grande flexibilité de déploiement. Cependant, les RCSFs à faible coût présentent certains obstacles et défis à relever. Le plus important d'entre eux est la faible précision des sondes de mesure. Par exemple, la plupart des sondes sont très sensibles à la température et à l'humidité. De plus, certains types de sondes présentent une réactivité croisée avec des molécules similaires. Les sondes optiques de comptage de particules, généralement utilisées comme capteurs de particules à faible coût, sont également moins précises que les stations au sol. En effet, le nombre de particules donné est sensible à de nombreux paramètres tels que la forme, la couleur, la densité, l'humidité et l'indice de réfraction des particules. Par ailleurs, la conversion du nombre de particules en masse est basée sur des modèles théoriques.

Mesure participative de la qualité de l'air

Dans cette thèse, nous considérons trois éléments clés concernant la mesure participative tout en nous focalisant sur la surveillance de la qualité de l'air : 1) la conception d'un système de surveillance participative de la qualité de l'air basé sur des capteurs à faible coût et une communication sans fil à longue portée ; 2) l'évaluation du potentiel des données des capteurs à faible coût dans l'interpolation spatiale et l'assimilation des données ; 3) la sélection de chemins pour les participants aux campagnes de mesures afin d'améliorer la connaissance du phénomène étudié.

Systèmes de surveillance de la qualité de l'air à faible coût

La conception d'un système efficace de surveillance de la qualité de l'air à l'aide de capteurs alimentés par batterie et peu coûteux est un défi majeur dans le domaine de la mesure mobile. Elle nécessite la prise en compte de multiples aspects liés à la plateforme, à son utilisation, à son coût et à sa maintenance.

Dans le cadre de la mesure mobile de la qualité de l'air, les nœuds de mesure sont en général portés par des participants, ce qui implique de placer différents composants électroniques dans un espace réduit sans affecter la qualité des mesures. Le système doit offrir la possibilité de collecter des données sur un ou plusieurs polluants avec une résolution acceptable, tout en tenant compte de la mobilité du nœud. Les données doivent être horodatées et géolocalisées afin de relier la mesure à l'heure et au lieu correspondants. Les relevés des capteurs sont envoyés à des serveurs distants pour être traités et reconstruits en vue de leur visualisation. Cela nécessite une technologie de communication adaptée qui permet d'atteindre de longues distances tout en étant économe en énergie afin d'assurer la connectivité du réseau. En outre, les participants peuvent ne pas avoir de connaissances techniques spécifiques, ce qui pourrait les empêcher d'utiliser la plateforme, d'où l'importance de prêter attention à la conception

des nœuds et à la manière dont les participants les utilisent en cachant la complexité derrière leur conception.

Un autre aspect important est la relation entre la fréquence de mesure et sa qualité. L'objectif d'un système d'évaluation de la qualité de l'air est de construire des cartes à haute résolution spatio-temporelle qui décrivent les phénomènes étudiés. Cela implique un réseau dense de nœuds avec un taux d'échantillonnage significativement élevé. Cependant, une telle configuration n'est pas pratique pour les projets à petit budget, les petits capteurs alimentés par batterie ou les protocoles de communication à faible consommation. En effet, l'une des principales limites rencontrées avec ce type de capteurs est leur autonomie. Un taux d'échantillonnage élevé consomme beaucoup d'énergie et réduit considérablement l'autonomie des nœuds de capteurs.

Exploitation de données denses et peu précises

En mesure participative, les capteurs collectent et envoient généralement une quantité importante de données de manière périodique, ce qui nécessite des algorithmes adaptés pour traiter de grands volumes de données, afin d'en extraire des informations utiles. Dans le cas de la surveillance participative de la qualité de l'air, l'élaboration de cartes de pollution repose généralement sur des techniques d'interpolation spatiale ou d'assimilation de données, qui utilisent des mesures collectées à l'aide de capteurs sujets à des erreurs. De plus, les capteurs peuvent être hétérogènes, ce qui peut impliquer des niveaux de précision différents. Par conséquent, l'analyse, le nettoyage et la prédiction des données manquantes pour générer des cartes de concentration de polluants peuvent être très difficiles. Dans ce contexte, il est courant de combiner les données de ces capteurs avec des informations supplémentaires relativement plus précises. Ces données peuvent améliorer remarquablement la qualité de la prédiction finale en fournissant des détails sur d'autres paramètres (par exemple, des détails sur les conditions de mesure) qui influencent le phénomène étudié. En outre, la caractérisation de l'incertitude des mesures recueillies peut être très utile pour décider quelle source d'information est la plus fiable.

Le routage des participants dans la mesure participative mobile

La qualité des cartes de pollution générées à l'aide de la mesure participative est fortement liée au déplacement et à la distribution des citoyens impliqués. Les informations fournies par les itinéraires des participants sont en effet importantes, car les participants qui passent par les mêmes itinéraires ou des itinéraires similaires apporteront peu d'informations supplémentaires à la cartographie globale. Ne pas tenir compte de cet aspect peut donner lieu à des zones très échantillonnées d'un côté, et d'autres beaucoup moins de l'autre côté. Il y a ainsi toujours un compromis à trouver entre l'étendue de la zone d'intérêt et la granularité des informations collectées.

La planification d'itinéraire ou la sélection d'itinéraire sont d'excellents outils qui peuvent être mis en œuvre afin d'influencer le déplacement des participants sans trop impacter leur distance parcourue. La planification d'itinéraire consiste à construire le chemin d'un participant sur la base d'un ensemble de points d'intérêt, tout en tenant compte de ses contraintes en termes de distance et de durée du voyage. Dans la sélection d'itinéraire, le participant dispose d'un ensemble de chemins préférés et le rôle du système est de choisir celui qui répond le mieux aux besoins de la tâche. Bien que le fait d'avoir des itinéraires candidats qui satisfont déjà les contraintes de temps et de distance réduise l'espace de recherche, cette approche est cependant plus

limitée que la planification d'itinéraire. En effet, la sélection d'itinéraire n'offre pas un contrôle total sur les chemins et peut rencontrer un ensemble d'itinéraires qui ont peu ou aucune valeur ajoutée.

En plus de la distance et de la durée du trajet, les solutions susmentionnées doivent prendre en compte les erreurs de mesure, ce qui constitue un défi puisque les capteurs à faible coût peuvent présenter des erreurs différentes même s'ils sont du même type ou proviennent du même fabricant. La redondance des données doit également être prise en compte, car les itinéraires des participants peuvent se chevaucher sur une certaine distance.

Contributions de la thèse

Dans cette thèse, nous abordons le problème de la mesure participative mobile pour l'évaluation de la qualité de l'air avec des RCSFs à faible coût, tout en adoptant une approche scientifique à la fois expérimentale et théorique. Nous avons conçu, testé et validé un système participatif mobile de surveillance de la qualité de l'air à base de capteurs à faible coût. Nous étudions également l'impact de la fréquence de mesure sur la consommation d'énergie des capteurs et sur la performance des modèles d'estimation. De plus, nous évaluons les performances des modèles de régression et d'assimilation de données pour la cartographie de la qualité de l'air dans un déploiement à grande échelle. Enfin, nous explorons le problème de la sélection de chemins dans le contexte de la mesure participative tout en tenant compte de la faible précision des capteurs. Il convient de mentionner que, bien que les contributions suivantes soient principalement axées sur la surveillance de la pollution de l'air, elles peuvent viser d'autres applications environnementales telles que le suivi des îlots de chaleur urbains.

Conception d'une plateforme participative de surveillance de la qualité de l'air : Au cours de cette thèse et dans le cadre du projet 3M'Air, nous avons conçu un système de mesure participatif mobile à faible coût à base de petits nœuds portables. Les nœuds capteurs présentés sont alimentés par des batteries et équipés de trois sondes de mesure environnementale qui mesurent la température, l'humidité relative, le dioxyde d'azote (NO_2) et les particules fines (PM_1, $PM_{2.5}$ et PM_{10}). En outre, chaque nœud est équipé d'un abri anti-radiation solaire qui protège les sondes tout en permettant une ventilation naturelle. Les mesures sont effectuées périodiquement, et sont horodatées et géolocalisées grâce au récepteur GPS intégré. Les données collectées sont téléchargées vers un serveur distant pour un stockage et un traitement centralisés. En outre, les nœuds sont équipés d'une carte microSD pour le stockage local, afin d'augmenter la robustesse en cas de perte de communication. La transmission des données repose sur LoRaWAN, une technologie LPWAN (*low-power wide-area network*) qui permet des communications à longues distances avec une faible consommation d'énergie. Afin de visualiser les données collectées, nous avons conçu une interface graphique accessible à l'aide d'un navigateur web.

Le travail expérimental a abouti à 16 nœuds capteurs qui ont été testés et comparés à des capteurs de référence pour validation. Sur la base de ces comparaisons, nous montrons qu'à l'aide d'une formule de calibration simple, les mesures de nos capteurs peuvent être corrigées. Les nœuds ont été impliqués dans de multiples campagnes de mesure que nous avons coorganisées à Lyon, de juin à octobre 2019.

Évaluation des performances et du potentiel des capteurs à faible coût pour la cartographie de la qualité de l'air : Dans cette partie, nous avons analysé les données collectées lors des campagnes de mesure de 3M'Air. L'objectif était d'explorer l'impact de la fréquence de mesure sur la consommation d'énergie et la qualité de la cartographie de la pollution. Nous avons évalué la consommation d'énergie des nœuds avec différentes configurations et identifié les éléments à forte consommation d'énergie. Par la suite, nous avons considéré une métrique de cycle de mesure et défini un modèle énergétique qui prend en compte ce cycle pour évaluer l'impact de la fréquence de mesure sur la consommation d'énergie. Afin d'évaluer la performance de la cartographie de la pollution avec différentes fréquences de mesure, nous avons combiné les concentrations de $PM_{2.5}$ collectées avec un ensemble de variables explicatives qui fournissent des informations sur la zone étudiée. Ensuite, nous avons appliqué des modèles d'interpolation spatiale pour construire des cartes de pollution et comparer les erreurs d'estimation en utilisant chaque fréquence de mesure. Il est intéressant de noter que les résultats révèlent qu'une légère réduction de la fréquence de mesure ne diminue pas de manière significative la qualité de la cartographie, mais qu'en revanche, elle peut considérablement prolonger le temps de fonctionnement du nœud.

Régression Vs assimilation de données pour la cartographie de la qualité de l'air à l'aide de données de capteurs à faible coût : Dans la cartographie de la qualité de l'air à l'aide de RCSFs à faible coût, la qualité de la carte dépend de multiples facteurs, tels que la précision des capteurs, la densité du réseau et le modèle d'estimation sélectionné. Dans cette étude, notre objectif était d'examiner l'impact des caractéristiques intrinsèques du déploiement dense de RCSF à faible coût pour la surveillance de la qualité de l'air (taille du réseau, erreurs de modèle numérique et erreurs de mesure) sur la performance des modèles de régression et de l'assimilation de données qui combine des simulations numériques et des mesures collectées à partir de capteurs déployés. À cet égard, nous avons modélisé les erreurs de mesure et caractérisé l'erreur de simulation d'un modèle numérique largement utilisé à l'aide de mesures provenant d'un réseau de stations de surveillance dans la région de Lyon. Nous avons proposé un framework général qui permet la comparaison de différentes stratégies basées sur des simulations numériques et une estimation adéquate des covariances des erreurs de simulation ainsi que des covariances des erreurs de mesure. Les résultats indiquent que l'assimilation de données surpasse toutes les méthodes de régression considérées dans tous les scénarios envisagés.

Sélection de routes dans la mesure participative de la qualité de l'air : Dans cette partie, nous avons abordé le problème de la sélection de routes dans un contexte de mesure participative de la qualité de l'air. L'objectif était de suggérer pour chaque participant un itinéraire qui contribue le plus à la réduction de l'erreur d'estimation, tout en tenant compte des contraintes du participant en termes de distance parcourue et de la relation entre les itinéraires. Nous avons proposé deux algorithmes de sélection de route qui prennent en compte la faible précision des capteurs à faible coût afin de trouver les itinéraires les plus informatifs. Le premier algorithme est basé sur une métrique de similarité et vise à maximiser la couverture spatiale en réduisant les chevauchements entre les itinéraires des participants. Le deuxième algorithme tire avantage du clustering hiérarchique pour construire des groupes de points similaires sur la carte en fonction des variables explicatives, afin de maximiser la di-

versité de l'information collectée. Nous avons comparé les performances des solutions proposées aux algorithmes de sélection d'itinéraire de base, en termes de distance parcourue et d'erreur d'estimation globale. Nos solutions ont mieux réussi à réduire l'erreur globale de cartographie par rapport aux autres algorithmes, tout en étant efficaces en ce qui concerne la distance du trajet.

Contents

1 Introduction **15**
- 1.1 Context of the thesis . . . 15
  - 1.1.1 Air pollution and urban heat islands: a major threat . . . 15
  - 1.1.2 Air quality conventional monitoring methods . . . 16
  - 1.1.3 Low-cost environmental sensors . . . 17
  - 1.1.4 Wireless Sensor Networks . . . 18
  - 1.1.5 Mobile crowdsensing . . . 19
  - 1.1.6 Participatory air quality and UHI monitoring using low-cost WSNs . . . 20
- 1.2 Thesis challenges . . . 21
  - 1.2.1 Low-cost environmental monitoring systems . . . 21
  - 1.2.2 Exploiting dense and inaccurate data on air quality . . . 22
  - 1.2.3 Citizens routing in mobile participatory sensing . . . 22
- 1.3 Contributions . . . 23
- 1.4 Organization of the following chapters . . . 25
- 1.5 List of publications . . . 25

2 Air quality assessment: from measurements to spatial mapping **26**
- 2.1 Emergence of low-cost sensors for air quality and urban heat islands assessment . . . 26
- 2.2 Related low-cost WSN-based platforms . . . 27
  - 2.2.1 OpenSense . . . 27
  - 2.2.2 UrPolSens . . . 27
  - 2.2.3 City Scanner . . . 27
  - 2.2.4 Zigbee-based network for temperature and humidity assessment 28
  - 2.2.5 BLE-based monitoring platform . . . 28
  - 2.2.6 ZigBee-based VOCs monitoring system . . . 28
  - 2.2.7 ESP-NOW-based air pollution sensing platform . . . 29
  - 2.2.8 Citi-Sense-MOB . . . 29
  - 2.2.9 Comparison and discussion . . . 31
- 2.3 Measurement-based air quality spatial mapping . . . 31
- 2.4 Background on Kriging . . . 33
- 2.5 Background on machine learning-based spatial interpolation . . . 35
  - 2.5.1 K-Nearest Neighbors . . . 37
  - 2.5.2 Multiple Linear Legression . . . 38
  - 2.5.3 Random Forest . . . 38
  - 2.5.4 Gradient Boost . . . 39
  - 2.5.5 eXtreme Gradient Boosting . . . 40
  - 2.5.6 Multilayer perceptron . . . 40
- 2.6 Air quality mapping using spatial interpolation . . . 42

2.7 Background on data assimilation . . . 44
2.8 Air quality mapping using data assimilation . . . 45
2.9 Conclusion . . . 47

3 Design of low-cost sensor-based air quality monitoring systems 48
3.1 Objective and design guidelines . . . 48
3.2 System architecture . . . 50
3.2.1 Sensor nodes . . . 51
3.2.2 Transmission layer . . . 52
3.2.3 Storage and processing . . . 54
3.2.4 Visualization . . . 54
3.3 Platform validation . . . 55
3.3.1 Comparison to reference sensors . . . 56
3.3.2 Cross comparison . . . 59
3.3.3 Energy consumption . . . 60
3.4 Conclusion . . . 62

4 Data analysis of WSN-based air quality monitoring systems 64
4.1 Area of interest and datasets . . . 64
4.1.1 Area of interest . . . 64
4.1.2 3M'Air platform data . . . 65
4.1.3 Explanatory variables . . . 65
4.1.4 Numerical simulations . . . 66
4.2 3M'Air platform's data Analysis . . . 67
4.2.1 Comparison of regression models . . . 68
4.2.2 Sensor data reconstruction . . . 70
4.2.3 Sensing rate VS energy consumption . . . 71
4.3 Air quality mapping using dense sensor networks . . . 73
4.3.1 Framework overview . . . 73
4.3.2 Characterization of the variance of simulation errors . . . 75
4.3.3 Ground truth and measurements generation . . . 75
4.3.4 Evaluation of regression approaches . . . 76
4.3.5 Evaluation of data assimilation . . . 77
4.3.6 Regression vs data assimilation . . . 79
4.4 Conclusion . . . 80

5 Route selection in participatory mobile sensing 82
5.1 Route selection : a paradigm of great interest in mobile crowdsensing 83
5.2 Problem statement . . . 85
5.2.1 Scenario Description . . . 85
5.2.2 Objective . . . 85
5.2.3 Mathematical notation . . . 86
5.3 Traditional route selection algorithms . . . 86
5.4 Similarity-based route selection . . . 87
5.5 Cluster-based route selection . . . 88
5.6 Validation . . . 90
5.6.1 Methodology . . . 90
5.6.2 Study area and reference map . . . 90
5.6.3 Participant routes and sensor measurements generation . . . 91
5.6.4 Computing the similarity . . . 91

5.6.5 Clusters construction 92
5.7 Performance evaluation and discussion 92
5.7.1 Comparison of the performance of MLR, KNN, and XGBoost 93
5.7.2 Comparison of route selection algorithms 93
5.8 Conclusion 95

6 Conclusion and perspectives **97**
6.1 Main conclusion 97
6.2 Extensions and future work 98
6.2.1 Future extensions of our participatory sensing platform 98
6.2.2 Expanding the network's lifetime 98
6.2.3 Extensions of our route selection solutions 99

List of Tables

2.1 Comparison of different environmental monitoring platforms 30

3.1 RMSE and correlation coefficient of PM_1, $PM_{2.5}$ and PM_{10} measurements taken by the designed node next to a reference device 58
3.2 RMSE and correlation coefficient for measurements of temperature, relative humidity and particulate matter concentrations 61
3.3 Comparison of different operating configurations 62

4.1 Main categories of dependent variables 67
4.2 Average execution time of 1 iteration for KNN, MLR, XGBoost, and MLP 69
4.3 An example of Pearson's coefficient of correlation of $PM_{2.56}$ for sensors concentrations 70
4.4 3M'Air node's operating time in function of the sampling rate 72
4.5 MAE vs k value of KNN regression ($\sigma_m = 2\ \mu g/m^3$, Fraction of deployed sensors = 0.3) 76

5.1 MAE vs the number of clusters with 40 participants 92
5.2 Average execution time of 1 iteration For KNN, multiple linear regression, and XGBoost 93

List of Figures

1.1 Air quality assessment methods 17
1.2 Architecture of a wireless sensor network 18

2.1 Illustration of spatial interpolation 32
2.2 A spherical variogram model 34
2.3 Architecture of a multilayer perceptron network 41
2.4 Illustration of the influence of weights and activation functions on the error of an MLP 42

3.1 System architecture overview 50
3.2 Design of the 3M'Air sensor node 52
3.3 The node PCB with the different modules embedded 52
3.4 The web dashboard of the platform 55
3.5 The web interface showing the measurements of the nodes on a map . 55
3.6 Our deployed node on the field of "Météo-France" in Lyon, France . . 56
3.7 3M'Air node vs reference station (a) temperature; (b) relative humidity 57
3.8 3M'Air node vs reference station (a) raw measurements; (b) calibrated measurements 59
3.9 Measurements from our designed nodes for (a) temperature (b) $PM_{2.5}$ concentrations 60

4.1 An example of a pre-defined sensing path 66
4.2 Annual averages of NO_2 concentrations for the city of Lyon in 2008 . 67
4.3 Assignment of explanatory variables to points measured by the sensors 68
4.4 MAE of $PM_{2.5}$ concentrations estimation for each sensing campaign . 69
4.5 MAE of $PM_{2.5}$ concentrations estimation in function of the training set fraction 69
4.6 Example of $PM_{2.5}$ concentrations measured by our sensing nodes during the campaign of (a) October, 5th 2019 (b) October, 12th 2019 70
4.7 $PM_{2.5}$ concentration estimations MAE for cross-validation 71
4.8 MAE of $PM_{2.5}$ concentrations estimation with three different sampling rates using 80% of data for training 73
4.9 Overview of the methodological flowchart used for the framework . . 74
4.10 The zone of interest (a) map of the center of Lyon and its immediate vicinity (b) annual averages of NO_2 concentrations in 2008 (simulated data) 74
4.11 Error standard deviation vs Simulation Values 75
4.12 MAE vs percentage of sampled points, with σ_m= 1 $\mu g/m^3$ 76
4.13 (a) MAE vs standard error deviation, Fraction of deployed sensors = 0.3; (b) MAE of XGBoost method vs percentage of deployed sensors, $\sigma_m \in [1, 3, 7]$ 77

4.14 MAE of data assimilation in function of the sensing error and the percentage of covered points $\sigma_m = 3\ \mu g/m^3$ 78
4.15 MAE of data assimilation in function of the sensing error and the percentage of deployed sensors, α=0.05 79
4.16 MAE of all regression and data assimilation methods in function of α, with $\sigma_m = 1\ \mu g/m^3$ and 30% of points sampled 79
4.17 NO_2 concentrations heatmap for the area of interest, (a) a realization of the ground truth (b) random forest estimation (b) BLUE estimation 80

5.1 An example of overlapping segments 87
5.2 An example of two possible paths activating different clusters 89
5.3 The general methodology used in validation 90
5.4 (a) Area of interest (b) reference heatmap of NO_2 concentrations (simulated data) 91
5.5 An example of two routes not overlapping but close enough 92
5.6 MAE vs number of participants 93
5.7 MAE vs number of participants 94
5.8 Total travel distance for the different route selection algorithms . . . 95
5.9 Heatmap of NO_2 concentrations, estimated with 40 users and the similarity-based algorithm 95

Chapter 1

Introduction

In this introductory chapter, we first present an overview of the air pollution and urban heat islands (UHI) phenomena, mentioning their effects on human health and the environment. We will provide an overview on conventional methods and approaches used for air quality monitoring. The advantages and disadvantages of low-cost sensors will be pointed. Moreover, we will highlight the benefits of mobile crowdsensing and its added value in air quality monitoring.

1.1 Context of the thesis

1.1.1 Air pollution and urban heat islands: a major threat

Air pollution represents a major concern in many cities worldwide. Indeed, pollutant concentrations exceed standard health limits in many cities several times a year. Air pollution consists of chemicals and particles released into the atmosphere that pose serious threats to health and the environment. According to the World Health Organization (WHO), both indoor and outdoor air pollution caused 7 million deaths in 2016 [1] of which nearly 600,000 were children. The sources of air pollution can be natural, such as volcanic eruptions or wildfires. However, these sources are responsible for a small amount of pollution, as most air pollution is caused by human activities mainly related to massive industrialization and high traffic density [2]. Air pollutants are classified into primary and secondary pollutants. Primary pollutants are released directly into the air, such as nitrogen oxides from car exhausts. Secondary pollutants are a result of chemical reactions between other pollutants such as ground-level ozone (O_3) which is created when nitrogen oxides (NO_x) and volatile organic compounds (VOC) react in the presence of sunlight [2, 3]. In Europe, air pollution remains high despite European Union (EU) policies. According to the European Environment Agency (EEA), air quality standards are exceeded in 130 cities in Europe, leading to about 400,000 premature deaths each year [4].

The adverse effects of air pollution have been widely reported in numerous studies. In fact, air pollutants are small enough to easily break through the human body's protective barriers. Therefore, exposure to high concentrations of pollutants over an extended period of time can lead to heart disease, reduced lung function and respiratory infections which results in increased mortality in polluted cities [4, 3]. The nefarious impact of air pollution is not limited to human health, but also affects the environment. Excessive levels of nitrogen oxides, for instance, contribute to the acidification of soil, modify the growth rate of plants, and consequently affect biodiversity

[2, 4]. In addition, the economy in turn is negatively impacted by air pollution. Studies established the relationship between high concentrations of $PM_{2.5}$ (particulate matter with an aerodynamic diameter of 2.5 μm or less), increased health care costs, and reduced gross domestic product (GDP) [5, 6].

Furthermore, continuous urbanization of cities across the world contributes to the suppression of vegetated surfaces for buildings and paved roads. Therefore, heat gets trapped between building and leads to a significant temperature difference between urban areas and their surrounding rural ones, creating a phenomenon known as an urban heat island (UHI) [7]. Urban heat islands are impacted by several factors, such as the size and building patterns of a city. They can have serious health problems as they greatly increase the risk of allergies, respiratory and cardiovascular problems, and can lead to excess mortality among vulnerable people during harsh heatwaves [8].

Air pollution and urban heat islands are strongly linked. Indeed, heatwaves significantly increase energy demand, which leads to burning more fossil fuels. This releases greenhouse gases that trap sun heat in the atmosphere, resulting in higher global temperatures, which in turn contributes to the degradation of air quality by increasing the concentration of certain pollutants [8, 9].

All these harmful consequences of both air pollution and urban heat islands have motivated remarkable research efforts to help cut these effects. The first step in this endeavor is to have a fine-grained knowledge of air quality and urban heat islands. To this end, multiple assessment methods have been put in place, in particular for air quality.

1.1.2 Air quality conventional monitoring methods

In order to better characterize and then mitigate the impact of air pollution, regional and governmental authorities have made considerable efforts in the implementation of air quality monitoring methods. In France, multiple air quality monitoring stations (See Figure 1.1a) have been installed by government-certified regional associations such as Atmo-AURA [1] which is responsible for air quality monitoring in the region of "Auvergne-Rhône-Alpes". These monitoring stations measure with high accuracy numerous pollutants such as monoxide carbon (CO), nitrogen dioxide (NO_2), Ozone (O_3), and particulate matters (PM_1, $PM_{2.5}$, PM_{10}) in addition to meteorological conditions (e.g., temperature, relative humidity, wind speed, etc.). Although being highly accurate, these stations are deployed in small numbers due to their large size, inflexibility, and high maintenance cost. Therefore, it is impractical to achieve high spatial resolution using only monitoring stations.

To cope with the aforementioned limitations, other air quality assessment methods have been considered to assist traditional monitoring stations. For instance, physical models are often adopted in air quality estimation [10, 11]. They calculate the spread of air pollution by means of numerical simulations in order to generate pollution maps (see Figure 1.1b). To achieve such result, physical models are supplied with meteorological conditions, geographical characteristics of the zone of interest, and the locations of pollution emission sources. However, it is challenging to achieve fine-grained knowledge using only these models, as the input data are usually averaged in both space and time.

Satellite remote sensing is another approach that can be used in air quality and

[1] https://www.atmo-auvergnerhonealpes.fr/

urban heat islands monitoring (see Figure 1.1c). In fact, advanced instruments can be mounted on board satellites, allowing them to measure reflected sunlight and map various trace gases such as nitrogen dioxide, ozone, and carbon monoxide [12]. Although this method provide a spatial resolution of around 10 meters, it is expensive and not easily accessible. In addition, it is highly dependent on climatic conditions (e.g., measurements can be obstructed by dense clouds) and the temporal resolution of its measurements is low due to the time it takes for such low earth orbit satellites to cover the earth.

(a) (b) (c) (d)

Figure 1.1: Air quality assessment methods (a) ENVEA's air quality monitoring station in "Rhône-Alpes", France, (b) output of a physical model, (c) surface temperature image taken from a satellite, (d) low-cost environmental sensors

1.1.3 Low-cost environmental sensors

All the accumulated limitations from traditional air pollution assessment methods, along with recent developments in environmental sensing instruments, have led to the emergence of small and low-cost environmental sensing probes (see Figure 1.1d). These probes offer a lower sensing quality compared to official monitoring stations. Nonetheless, their cost-effectiveness and flexibility allow for large-scale deployments and access to a wider audience.

Today, there is a plethora of environmental sensors available on the market. Their manufacturing processes differ from one manufacturer to another, but they can be grouped in categories based on their sensing technology. For example, electrochemical sensors measure the amount of current generated from the chemical reaction between the target gas and the electrode of the sensor. This type of sensors have a good sensitivity, however, they present cross-reactivity problems (confusing similar molecule types to the target ones) and high sensitivity to meteorological changes such as temperature and humidity [13, 14].

Another widely used type in air pollution monitoring is the optical particle counters, which calculates the number of particulate matter using a light scattering technique. When a particle sample enters the detection chamber of the sensor, a powerful light source illuminates the particles, and the scattered light data is captured to give an estimation of the particles number. These sensors can achieve a good estimation of particle counts, but their quality may depend on the density of the particles, their shape, or their refraction index [13, 14]. These sensing probes are usually integrated into communicating sensor nodes to enable data storage and transfer to remote servers. However, in order to achieve long-term and large-scale deployment with a relatively low budget, the hosting sensor nodes are often limited in terms of energy, size, computation, and memory resources.

1.1.4 Wireless Sensor Networks

The recent advances in the field of low-power wireless communication technologies allow the interconnection of the abovementioned sensor nodes while offering more flexibility, autonomy, and data accessibility. As a result, an appealing type of network has emerged, known as wireless sensor networks (WSNs). WSNs are a deployment of numerous nodes equipped with sensors, communicating and performing a collaborative sensing task. These devices are often battery-powered and arranged in a variety of topological network configurations (e.g., star, mesh, etc.). WSNs offer a great advantage over traditional wired networks, whether in terms of deployment costs, maintenance complexity, or mobility [15].

WSNs enable smart city projects and large spatial coverage with budget constraints and little to no human assistance. Indeed, numerous applications are nowadays powered by WSNs in different fields. In fact, they are used in various applications such as agriculture [16], traffic light control [17], air pollution monitoring [18], smart parking [19], health care [20], structural health monitoring [21], etc. The main components of WSNs are the sensing nodes which perform the measurements and send them to a gathering device known as the gateway (also called sink or base station) using generally a low-power communication technology. The gateway often has more resources than the sensing nodes and can be either battery-powered or power-plugged. In terms of communication, it is usually equipped with an internet connection, through which the data are sent to a remote server for storage and processing (see Figure 1.2).

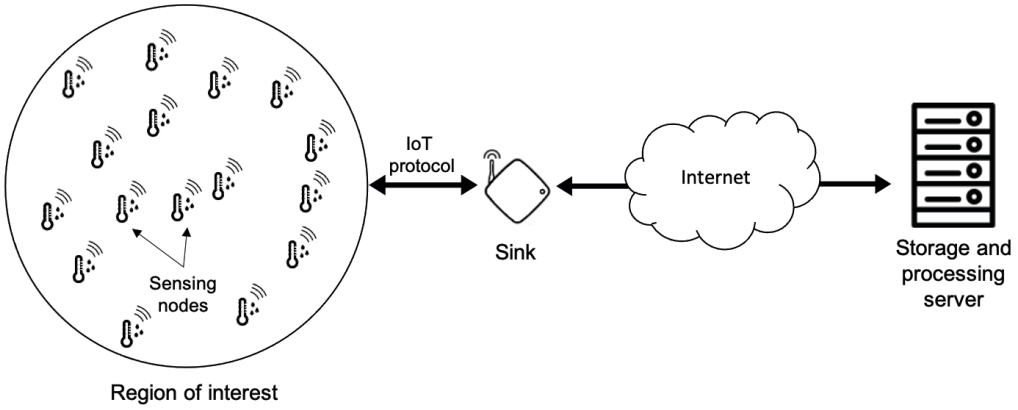


Figure 1.2: Architecture of a wireless sensor network

Numerous WSN-based solutions have been powered by wireless communication technologies, such as Bluetooth Low Energy (BLE), Wi-Fi, and ZigBee, which offer different features and capabilities. BLE (IEEE 802.15.1) is a short-range communication technology that has been designed to fit the low-power constraints in WSN. It offers a theoretical communication range of up to 240 meters and a maximum data rate of up to 2Mbps [22]. ZigBee is another short-range and low-power communication technology that is based on the IEEE 802.15.4 standard, and that has been widely adopted in WSN projects [23]. It enables wireless communication between devices at a maximum distance of 100 meters and a data rate up to 250 kbps [22]. Wi-Fi (IEEE 802.11) allows devices to access local networks with a larger communication range and much higher data rates (e.g., up to 600Mbps in 802.11n [22]) compared to BLE and Zigbee. However, Wi-Fi has a relatively higher power consumption than BLE and ZigBee, which limits its adoption in applications with strict power limitations [23].

Although BLE, ZigBee, and Wi-Fi have been powering a significant number of WSN-based solutions, they are not suitable for applications with dense WSN (e.g., a maximum of 255 devices for ZigBee [22]), long-range communication requirements, or ultra low-power constraints [22, 24]. This led to the rise of a new solution known as LPWAN (low-power wide-area network) communication technologies. LPWAN solutions such as SigFox, NB-IoT, and LoRa/LoRaWAN are designed to take into consideration the characteristics of highly demanding WSN applications, such as large-scale deployments, low cost, small data packets, low data rates, limited resources, and low-power consumption. For instance, SigFox is a popular low-power communication technology that uses the unlicensed industrial, scientific, and medical (ISM) band [22, 24]. It allows sending packets at up to 100bps while achieving wide communication ranges (between 10 km in urban areas and 50 km outside) [22, 24], which is remarkably larger than what BLE, ZigBee, or even Wi-Fi can offer. LoRaWAN is another promising ISM-based LPWAN technology available on the market. It is built upon LoRa's physical layer and supports data transmission over up to 15 km [25] with a data rate between 0.3 and 50 kbps [22, 24]. Besides the low data rates compared to short-range communications, SigFox and LoRaWAN technologies have other limitations that translate into a low duty cycle, limited data packets (e.g., 12 bytes in UpLink for SigFox), etc. [22, 24]. These aspects are to be considered in the design of a new solution using LPWAN technologies in addition to the intrinsic challenges of WSNs, including limited resources, energy constraints, data recovery, security, etc. [26].

1.1.5 Mobile crowdsensing

Mobile crowdsensing (MCS) is an emerging paradigm that has attracted great attention in the recent years, owing to the result of the rapid evolution of WSNs. MCS leverages population density in a world full of connected devices and involves citizens in the collection of large volumes of data in multiple fields. In MCS-based applications, participants use sensors embedded on their smart devices (smartphones, smartwatches, etc.) or autonomous nodes equipped with more or less accurate sensors to accomplish sensing tasks [27]. The mobile nature of the population allows MCS-based applications to achieve high spatio-temporal coverage with relatively less budget. The large amount of data collected is transmitted to the cloud for storage and analysis.

Mobile crowdsensing comes in two forms, "opportunistic sensing" in which sensing tasks are performed with less to no user involvement. In this type, the devices are autonomous and able to decide whether to perform measurements or not. The second type is known as "participatory sensing" in which participants have more freedom on when/where to start sensing and also which data to share [28]. It is to be noted that the difference between these terms remains unclear due to the absence of a clear definition of "opportunistic sensing". In fact, a significant number of research work is referred to as mobile crowdsensing [27]. Therefore, we will use the terms "mobile crowdsensing" and "participatory sensing" interchangeably in this manuscript.

Besides its very promising potential, participatory sensing comes with multiple challenges related to device heterogeneity, connectivity, power consumption, etc. Indeed, several surveys have identified multiple challenges and issues regarding the MCS framework [27, 28, 29].

Limited resources

Devices involved in mobile crowdsensing application are generally small and battery-powered in order to meet different requirements related to size, budget, and usability. This design translates into devices with limited resources in terms of computing power, connectivity, and storage. Therefore, equipping the sensors with the right components in order to achieve the expected performance with a reasonable power consumption represents a challenge for solution creators.

Connectivity

Connectivity is of a great importance since MCS enables large-scale deployment, which involves dense networks of communicating sensing devices. Due to their limitations, devices can not use communication technologies with high-power consumption. In most cases, MCS devices are equipped with BLE to send their data to the user's smartphone, which will use its internet connection to transfer the data to the cloud via Wi-Fi or cellular connection.

Data integrity and accessibility

Data quality is an important aspect of MCS, considering that data are collected by non-experts and with no particular supervision in general. Besides the low accuracy of the sensing probes, the quality of their readings also depends on the sampling conditions. In air quality monitoring, for instance, the mobility pattern and speed can change the air flow, which influences the sensing probe and its readings. In addition, some individuals could have bad intentions and use their devices to send erroneous data, which could negatively affect the performance of the system [28]. On the other hand, because data are generally transmitted to the cloud for analysis, a user might need to wait for the data he collected to be processed before being able to access it [29]. Thus, it is essential to find the right trade-off to ensure data accessibility for users without waiting, while still not giving them full control to ensure the integrity of the data.

Privacy

Ensuring privacy is always challenging when it comes to collecting massive data with the help of the population, especially in mobile sensing. In fact, mobile devices are generally equipped with a GPS receiver to geotag the data, and hence create a link between an information and its corresponding place or zone. These GPS readings could be used by malicious individuals to collect private information about the users, such as their home address, work address, and habits [28, 29].

1.1.6 Participatory air quality and UHI monitoring using low-cost WSNs

To achieve fine-grained knowledge of air quality and urban heat islands, there is a need for extensive data on multiple pollutants and meteorological parameters with high spatial and temporal resolutions. This implies deploying a dense network of sensors over the area of interest. However, investing in a large number of sensing nodes can be extremely costly and impractical, especially for low-budget and small projects. To cope with that, multiple assessment projects have relied on public transport lines as

a mobile platform on which sensors are mounted [30, 31, 32]. Though, this solution highly depends on the size of the public transport network (number of lines, vehicles, etc.) and its configuration. In addition, air quality monitoring applications cannot have any control over the path a bus or tramway will follow.

Participatory sensing is an excellent tool in many areas, including air quality assessment. It provides a broad deployment platform for air quality monitoring applications, thanks to the large number of volunteers involved in the process, that collectively perform measurements using portable sensor nodes. Moreover, citizens are more flexible than vehicles and can therefore access small zones and narrow roads, hence delivering better coverage. In addition, involving citizens in air quality assessment raises their awareness of climate issues, helps them to understand phenomena such as urban heat islands, and consequently contributes to changing poor practices that can worsen air quality and amplify the UHI effect.

In several use cases, the focus is solely on data collected by the participants at specific locations, although it is often interesting to have a broader spatial estimation of the monitored phenomenon. However, offering a complete coverage of the study zone is challenging, regardless of which sensing paradigm or technology to use. This is due to multiple factors that can be associated with the deployment, such as harsh deployment conditions, inaccessible zones (even for pedestrians), or with the sensing nodes being used (e.g., faulty sensing probes, communication loss, etc.). To cope with that, one of the solutions that has been widely and commonly used is the exploitation of these data with spatial interpolation models to achieve the expected coverage of the area of interest. These methods are considered to estimate pollutant concentrations at a given point where no sensing data has been collected, using the available measurements from other sensors [33].

1.2 Thesis challenges

In this thesis, we consider three key components regarding participatory sensing, while focusing on air quality monitoring: 1) the design of a participatory air quality and UHI monitoring system using low-cost sensors and long-range wireless communication [2]; 2) the assessment of the potential of low-cost sensors' air quality data for spatial interpolation and data assimilation; 3) the selection of paths for participants in mobile participatory sensing to improve the knowledge on atmospheric pollution.

1.2.1 Low-cost environmental monitoring systems

Designing an effective air quality and urban heat islands monitoring system using battery-powered, low-cost sensors is a major challenge in mobile sensing. It requires taking into consideration multiple aspects related to the platform, its usage, cost, and maintenance.

In mobile crowdsensing, the nodes are in general carried by participants, this means fitting different electronic components in a small space without impacting the quality of the measurements. The system has to offer the possibility of collecting data on one or more pollutants, temperature, and humidity, with an acceptable resolution, while taking into consideration the mobility of the node. Data should be

[2]for the sake of multidisciplinarity, our system has been designed to also collect temperature and relative humidity, in addition to pollutant concentrations. However, the assessment of UHI has been conducted by another team of the 3M'Air project

time-stamped and geotagged to link the measurements to their corresponding time and locations. Sensor readings are sent to remote servers to be processed and reconstructed for visualization. This requires an adapted communication technology that allows reaching long distances while being power-efficient, in order to ensure network connectivity. In addition, participants may have no specific technical knowledge, which could prevent them from using the platform, hence the importance to give attention to the design of the nodes and the way the participants interact with them, by hiding the complexity behind their design.

Another important aspect is the relationship between the sensing rate and the sensing quality. The goal behind an air quality assessment system is to build high spatio-temporal resolution maps that describe the studied phenomena. This implies a dense network of nodes with a significantly high sampling rate. However, such configuration is not practical for small budget projects, small battery-powered sensors, or low-power communication protocols. Indeed, one of the main limitations encountered with this type of sensors is their battery life. A high sampling rate is power consuming and considerably reduces the autonomy of the sensor nodes.

1.2.2 Exploiting dense and inaccurate data on air quality

Devices in MCS are generally collecting and sending a significant amount of data periodically, which requires adapted algorithms for processing large volumes of data, in order to extract useful information.

In low-cost air quality monitoring with MCS, building pollution maps usually relies on spatial interpolation or data assimilation techniques. Moreover, sensors can be heterogeneous, and hence, have different accuracy levels. Therefore, analyzing, cleaning, and predicting missing data to generate pollutant concentration maps can be very challenging. In air quality assessment, it is a common practice to combine MCS data with relatively more accurate additional information. These data can remarkably improve the quality of the final prediction by providing details on other parameters (e.g., details about the sampling conditions) that influence the studied phenomenon. In addition, characterizing the uncertainty in the collected measurements can be very helpful to decide which source of information is more trustworthy.

1.2.3 Citizens routing in mobile participatory sensing

The quality of pollution maps generated using participatory sensing is strongly related to the movement and distribution of the citizens involved in the sensing task. The information provided by the routes is in fact important because participants who pass by the same or similar routes will bring little additional information to the overall mapping. Not taking this into account may result in highly sampled zones on the one hand, and poorly sampled ones on the other hand. Thus, there is always a compromise to be found between the extent of the area of interest and the granularity of the collected information.

Route planning or route selection are great tools that can be implemented in order to influence the movement of the participants without impacting their traveled distance too much. Route planning consists of building a participant's path based on a set of points of interest, while considering his constraints in terms of travelling distance and duration. In route selection, the participant has a pool of preferred paths, and the role of the system is to choose the one that satisfies the best the needs of the task. Although having candidate routes that already satisfy time and distance

constraints reduces the search space, this approach is in fact more limited compared to route planning. Indeed, route selection does not offer full control over the paths and may encounter a set of routes that have little or no added value.

In addition to travelling distance and duration, the aforementioned solutions have to consider sensing errors, which is challenging given that low-cost sensors can have different errors, even if they are of the same type or share the same manufacturer. Data redundancy is also to be considered, as participants' routes may overlap along a certain distance.

1.3 Contributions

In this thesis, we tackle the problem of mobile participatory sensing for air quality assessment with low-cost WSNs, while adopting both an experimental and theoretical scientific approach. We designed, tested, and validated a mobile participatory air quality and urban heat islands monitoring system using low-cost sensors. We also study the impact of the sensing frequency on the energy consumption of low-cost sensors and on the performance of estimation models. Moreover, we evaluate the performance of regression models and data assimilation for air quality mapping in a large-scale deployment. Furthermore, we explore the route selection problem in the context of participatory sensing while taking into account the low accuracy property of low-cost sensors. It is worth mentioning that although the following contributions mainly focus on air pollution monitoring, they can target other environmental applications such as assessment of urban heat islands.

Contribution 1: Design of an end-to-end participatory air quality monitoring platform featuring low-cost sensors

During this thesis and as a part of the 3M'Air project, we designed a low-cost mobile participatory sensing system that is powered by small and portable nodes. The featured sensing nodes are battery-powered and equipped with three environmental sensing probes that measure temperature, relative humidity, nitrogen dioxide (NO_2), and particulate matters (PM_1, $PM_{2.5}$, and PM_{10}). In addition, each node is equipped with a front shield to protect the environmental sensing probes from solar radiations and supply them with natural ventilation. Measurements are performed periodically, and are time-stamped and geotagged, thanks to the embedded GPS receiver. The collected are data uploaded to a remote server for centralized storage and processing. In addition, the nodes are equipped with a microSD card for local storage, in order to increase robustness in case of communication loss. Data transmission relies on LoRaWAN which is a LPWAN technology that allows reaching long distances with low-power consumption. In order to visualize the collected data, we designed a graphical user interface that is accessible using any web browser. The experimental work resulted in 16 sensor nodes that have been tested and compared to reference sensors for validation. Based on these comparisons, we show that using a simple calibration formula, the measurements of our sensors can be corrected. The nodes powered multiple sensing campaigns we co-organized in the Lyon metropolitan area, from June to October 2019.

Contribution 2: Performance evaluation and assessment of the potential of low-cost sensors for air quality mapping

In this part, we carried out analysis of the data collected during the 3M'Air sensing campaigns. The objective was to explore the impact of the sensing rate on the energy consumption and the quality of the pollution mapping. We evaluated the power consumption of the nodes with different configurations and identified the elements with high-power consumption. Subsequently, we considered a sensing duty cycle metric and defined an energy model that takes into account the duty cycle, to assess the impact of the sensing rate on the energy consumption. In order to evaluate the performance of the pollution mapping with different sensing rates, we combined the collected $PM_{2.5}$ concentrations with a dataset of explanatory variables that provide information on the studied area (e.g., number of routes, vegetation, and population density). Then, we applied regression models to build pollution maps and compare the estimation errors using each sensing rate. Interestingly, results reveal that a slight reduction of the sensing rate does not significantly decrease the quality of the mapping, but on the other hand, it can extend the operating time of the sensor node.

Contribution 3: Regression Vs data assimilation for air quality mapping using low-cost sensors' data

In air quality mapping using low-cost WSN, the quality of the map depends on multiple factors, such as the accuracy of the sensors, the density of the network, and the selected estimation model. Through this study, our goal was to investigate the impact of the intrinsic characteristics of dense deployment of low-cost WSN for air quality monitoring (network size, numerical model errors and sensing errors) on the performance of regression models and data assimilation. In this vein, we modeled sensing errors and characterized the simulation error of a widely used numerical model with the help of measurements from a network of monitoring stations within the region of Lyon. We proposed a general framework that allows the comparison of different strategies based on numerical simulations and an adequate estimation of the simulation error covariances as well as the sensing errors covariances. The results indicate that data assimilation outperforms all considered regression methods in all the studied scenarios.

Contribution 4: Route selection in participatory sensing for air quality assessment

In this part, we addressed the problem of route selection in a participatory air quality sensing context. The objective was to suggest for each participant a route that contributes the most to the reduction of the estimation error, while taking into consideration the participant's constraints in terms of travelling distance and the relationship between routes. We proposed two route selection algorithms that take into consideration the low accuracy characteristic of low-cost sensors, in order to find the most informative routes. The similarity-based route selection algorithm aims to maximize spatial coverage by reducing overlaps between participant routes. The cluster-based route selection takes advantage of hierarchical clustering to build groups of similar points of the map according to the explanatory variables, in order to increase the diversity of the collected information. We compared the performance of the proposed

solutions to baseline route selection algorithms in terms of travelling distance and overall estimation error. Our solutions performed better in reducing the overall mapping error compared to the other algorithms, while being efficient regarding the travel distance.

1.4 Organization of the following chapters

Following this general introduction, we explore in Chapter 2 the assessment of air quality, from measurements to spatial mapping. We first review and compare some of the main WSN-based platforms for air quality sensing using low-cost sensors. In the second part, we provide a background on spatial analysis for air quality. We explain spatial interpolation and data assimilation and their respective most commonly used methods, while providing a review of the most related research work to our subject. In Chapter 3, we introduce our lab-designed participatory air quality platform and provide details on its different elements and their design. The first part of Chapter 4 investigates the impact of the sensing rate on the energy consumption and the mapping quality. The second part compares the performance of regression models and data assimilation in a dense network of low-cost WSN. Subsequently, we discuss in Chapter 5 the problem of route selection and propose two algorithms that take into consideration the relationship between routes and the sensing errors. Finally, we conclude this manuscript and provide some future directions in Chapter 6

1.5 List of publications

In Journals

[J1] Mohamed Anis Fekih, Walid Bechkit, Hervé Rivano, Manoël Dahan, Florent Renard, Lucille Alonso and Florent Pineau. Participatory Air Quality and Urban Heat Islands Monitoring System. In IEEE Transactions on Instrumentation and Measurement, 2020 [34].

[J2] Mohamed Anis Fekih, Walid Bechkit and Hervé Rivano. Route Selection for Low-cost Air Quality Monitoring in Mobile Participatory Sensing. (to be submitted)

In International Conference Proceedings

[C1] Mohamed Anis Fekih, Ichrak Mokhtari, Walid Bechkit and Hervé Rivano. On The Regression and Assimilation for Air Quality Mapping Using Dense Low-cost WSN. In the 34th International Conference on Advanced Information Networking and Applications (AINA 2020), Caserta, Italy [35].

[C2] Mohamed Anis Fekih, Walid Bechkit and Hervé Rivano. On the Data Analysis of Participatory Air Pollution Monitoring Using Low-cost Sensors. In the 26th IEEE Symposium on Computers and Communications (ISCC 2021), Athens, Greece [36].

Chapter 2

Air quality assessment: from measurements to spatial mapping

A fine-grained knowledge of air quality is closely related to the source of data, its accuracy, and availability. Background stations provide high-quality data, which makes them a very reliable source of information. However, their limited number significantly reduces their contribution to the assessment of the local exposure to pollutants. Other sources of information, such as remote sensing or physical models also have their own limitations in terms of accessibility, and spatio-temporal resolution. As seen in the previous chapter, the use of low-cost WSNs for air quality monitoring is a promising solution that can fill the gap left by traditional approaches.

Another important aspect of air quality assessment is the processing of the collected data, in order to extract meaningful information and generate concentration maps that can be easily interpreted. This task often involves combining multiple sources of information, dealing with inaccurate measurements, and predicting missing data, using a set of computational techniques.

In this chapter, we first provide a literature review of a selection of air quality and urban heat islands monitoring systems using low-cost sensors. We present their main features, architecture, and the main results of their deployment. In addition, we present a comparison of these monitoring platforms. Subsequently, we overview some of the most commonly used spatial interpolation techniques for air quality assessment, with a focus on the methods used during this thesis. Furthermore, we present the data assimilation approach while highlighting the main differences with the spatial interpolation. Finally, we review some existing research work in air quality assessment using either spatial interpolation or data assimilation.

2.1 Emergence of low-cost sensors for air quality and urban heat islands assessment

The effectiveness of the actions carried out by public spatial planning policies to mitigate the adverse effects of air pollution and urban heat islands is strongly linked to the fine knowledge of temperature gradients and air quality at local scales. Hence, the first step in producing high-resolution maps, is to establish an effective monitoring system with the primary objective of measuring temperature and air pollution concentrations.

During the recent years, low-cost sensors have acquired an important place in air quality monitoring and environmental applications in general. They are often consid-

ered as a good alternative source of information, owing to their cheap cost, small size, flexibility, and the fact that many of them do not require advanced knowledge to be utilized. Moreover, their large-scale deployments and the high spatio-temporal resolution of their data represent a great advantage over traditional approaches, despite their low accuracy. In addition, online or offline calibration techniques can also be used to correct their readings [37, 38]. These characteristics have not only motivated the raising of numerous individual projects, but have also attracted many certified associations, academics and industrials. [39, 40].

2.2 Related low-cost WSN-based platforms

Multiple environmental monitoring projects based on low-cost sensors have emerged over the past decade. In this section, we present a selection of related low-cost WSN-based research projects for air quality and urban heat islands assessment.

2.2.1 OpenSense

The OpenSense project [31] studies the performance of low-cost sensing probes by designing and deploying mobile sensor nodes on trams and buses. The nodes measure several environmental parameters such as temperature, humidity, O_3, CO, NO_2 and PM. In addition, they are equipped with a GPS and an accelerometer to geolocate the measurements. Each node incorporates a Linux-based core component which stores pollution data locally and streams it to a cloud server using either Wi-Fi or cellular connections. The platform offers a visualization tool that displays pollution concentrations on top of the map of the region. The project was deployed in Zurich and Lausanne in Switzerland, and mounted on top of 10 trams and 10 buses. Through this work, the researchers point out the challenge of calibrating low-cost sensors and propose techniques to reduce the calibration error [41].

2.2.2 UrPolSens

The UrPolSens project [42] proposes a low-cost energy-efficient air quality monitoring platform that uses fixed sensor nodes powered by solar panels and batteries. The nodes measure temperature, relative humidity, NO_2 and can be adapted to measure PM or VOC. Measurements are sampled at one second time intervals, and then averaged per minute and stored on a short-term EEPROM. After every 10 minutes, the average measurements are aggregated into one record, stored on a local SD card, and sent using LoRa communication to a gateway that forwards the packets via cellular network to a central cloud server for storage, filtering and processing. A web interface was developed to visualize real-time air pollution concentrations and weather conditions. The solution was deployed and tested for 3 months in Lyon, France in an urban street surrounded by two reference stations. The results show that the system is energy-efficient while keeping an acceptable degree of accuracy.

2.2.3 City Scanner

City Scanner [32] is a general-purpose mobile sensing platform that offers multiple configurations in terms of sensing probes. The solution does not require a specific

power supply and therefore can be mounted on top of any urban vehicle without impacting its operations. Each deployed node consists of a group of sensing modules sending their measurements via a short-range Wi-Fi network to a core unit which ensures power management, data storage and streaming to the cloud using open Wi-Fi hotspots. The proposed solution was deployed in the city of Cambridge, USA. Environmental probes measuring temperature, humidity and particulate matter were mounted on trash trucks along with other sensors like thermal cameras and Wi-Fi scanners. The main challenges encountered with the deployment were the data transfer reliability, power consumption and data fidelity.

2.2.4 Zigbee-based network for temperature and humidity assessment

A generic environmental monitoring platform based on fixed indoor sensors is presented in [43]. It uses open-source hardware platforms as Arduino and Raspberry Pi, and ZigBee as a communication protocol. The system is composed of sensor nodes measuring temperature and relative humidity, a web interface, and a base station that hosts a gateway node and a backend server. Sampling is performed each 35 min and measurements are sent using ZigBee to the base station that gathers and stores the data for future visualization on a web browser using the internet or a local network. The solution was tested indoor by deploying a base station and 3 sensor nodes measuring both temperature and relative humidity. The researcher identified the need for an important number of ZigBee routers for large-scale deployments.

2.2.5 BLE-based monitoring platform

An implementation of a flexible environmental monitoring system is described in [44]. The solution is based on a set of small sensor nodes and transceivers communicating in Bluetooth Low Energy (BLE) and a cloud-based backend. The sensors are equipped with a microcontroller, a low-power 2.4-GHz transceiver, a Real-Time-Clock (RTC), a temperature and relative humidity sensor, and a local storage that can store up to 125 000 measurements. These components are soldered along with a lithium battery on a small Printed Circuit Board (PCB). Measurements are logged at intervals of 15 minutes on the local memory and sent to a Raspberry Pi board that pushes data to the cloud. Two possible ways of visualizing data are offered, on a smartphone by connecting it to the node using BLE or on a web interface using a web browser. This solution was evaluated indoor in a heritage building in the north Italy for more than two months. The results point out the challenge of environmental condition variations from a position to another, even at small distances.

2.2.6 ZigBee-based VOCs monitoring system

Another cloud-based monitoring system is proposed in [45]. The system is composed of a network of sensors, a cloud system and an end-user layer. Sensor nodes incorporate a microcontroller, up to four gas sensors measuring a set of volatile organic compounds (VOCs) (benzene, toluene, ethylbenzene, and xylene). The nodes are also equipped with a ZigBee module for communicating with the gateway, and a power management controller that can switch between a battery and a solar panel to power

the node. The gateway used in this solution integrates a ZigBee module that receives measurement data at the end of each sampling cycle (10 minutes) and forwards them to the cloud system, where data are stored and processed for delivery to the end-user represented by a web application. The researchers evaluated the performance of the solution in the laboratory by generating different compounds at different concentrations. They also propose pattern recognition techniques to efficiently detect VOCs and show that they achieve good results in their quantification.

2.2.7 ESP-NOW-based air pollution sensing platform

The sensing platform presented in [46] is based on co-located fixed sensing units. In order to increase the availability, reliability and autonomy of the network, the project relies on a redundant configuration. Each sensing unit comprises four identical and independent sensing nodes that communicates using the ESP-NOW protocol, which allows low-power peer-to-peer 2.4GHz wireless communication. Each sensor node has its own microcontroller and is powered by two battery cells and a solar panel. Regarding the sensing capabilities, each node is equipped with a four sensing probes measuring $PM_{2.5}$, PM_{10}, air pressure, temperature, relative humidity, and soil moisture. At each 15-minute interval, all four nodes on a sensing unit perform sampling, but only one of them transmits its data to the gateway, which transfers the data to a processing server using a Wi-Fi connection. A node selection algorithm is proposed to switch between the nodes of a sensing unit based on their energy state. According to the authors, this configuration was chosen to limit data loss and allows the system to operate for an extended period. A total of 15 sensing units were deployed between November 2019 and May 2020 to monitor air particulate matter emissions of a construction site in a suburban area of Sidney, Australia. The deployed sensors were validated through a comparison to official monitoring stations, which showed good agreement between both systems.

2.2.8 Citi-Sense-MOB

The Citi-Sense-MOB project [47] aims to support green growth and sustainable development through a mobile environmental monitoring system formed by sensor nodes based on electrochemical sensors measuring NO_2, NO, CO, SO_2, O_3, CO_2, temperature, and relative humidity. Two sensing platforms are proposed, one mounted on buses and the other mounted on electrical bicycles. The first platform performs measurements by pumping the air inside a chamber in contact with the sensors and collects position, speed and use of brakes directly from the bus computer and sends the data to a central database. The other platform is powered by the bicycle battery and has no need for a pump as it is in direct contact with the air. Measurements gathering begins as a person starts riding the bicycle and turns off shortly after the bicycle stops. Collected measurements from both platforms are sent to the same database. Citizens can visualize nodes' measurements and positions gathered from the sensor nodes on a map using a web interface or a mobile app. Through this work, the authors aim at increasing the awareness of citizens and involving them in the adoption of new habits that contribute to the reduction of air pollution.

Projects	Fixed / mobile	Measured parameters	Sampling rate	Local storage	Communication technology	Energy consumption
OpenSense [31]	Mobile	Temperature, humidity, O_3, CO, NO_2 and particulate matter	/	Yes	Wi-Fi	Not specified
UrPolSens [42]	Fixed	Temperature, humidity and NO_2	1 sec	Yes	LoRa	18 mA
City Scanner [32]	Mobile	Temperature, humidity and particulate matter	/	Yes	Wi-Fi	Not specified
ZigBee-based and Arduino-powered system [43]	Fixed	Temperature and relative humidity	35 min	No	ZigBee	Not specified
BLE-based platform [44]	Fixed	Temperature and relative humidity	15 min	Yes	BLE	20 mA
ZigBee-based VOCs sensing system [45]	Fixed	VOCs	10 min	No	ZigBee	104 to 270 mA
ESP-NON-based dust sensing platform [46]	Fixed	$PM_{2.5}$, PM_{10}, relative humidity, soil moisture, temperature	15 min	No	ESP-NOW	Not specified
Citi-Sense-MOB [47]	Mobile	NO_2, NO, CO, SO_2, O_3, CO_2, temperature and relative humidity	30 sec	No	Bluetooth	Not specified

Table 2.1: Comparison of different environmental monitoring platforms

2.2.9 Comparison and discussion

Table 2.1 summarizes the main characteristics of the aforementioned low-cost environmental monitoring platforms: type of deployment (i.e., fixed or mobile), measured environmental parameters, sampling rate, capacity to store data locally, communication technology, and energy consumption when mentioned.

Temperature and humidity monitoring is present in most of the platforms, proving the importance and the impact of heat islands on other pollutants and air quality in general. Besides, most of the presented platforms use ZigBee, Wi-Fi or BLE, which are quite adapted to indoor or small-scale outdoor monitoring, because of their short or medium communication ranges. In the context of crowdsensing, BLE could be useful to send measurement data to users' smartphones, while transmitting data from the smartphones to the cloud requires a cellular connection or Wi-Fi. However, this operation, whether it requires a human intervention or not, may consume a significant amount of the smartphone's energy and mobile data plans. As an alternative, low-power long-range communication technologies could cover the whole network of sensors with a few gateways.

2.3 Measurement-based air quality spatial mapping

In practice, it is impossible to collect measurements at every single point of the study area, but only at certain observation locations. Therefore, the area of interest is often spatially divided into a grid of cells of equal size. It is then assumed that all points within the same cell have the same properties, hence the same level of pollutant concentration. As in an image, the number and size of these cells indicate the spatial resolution of the resulting map. However, even with such spatial discretization, it is still complicated to have measurements at each cell while keeping a decent spatial resolution. To cope with that, sensor measurements are in general combined with some techniques in order to estimate missing data.

Estimating air pollution concentrations at an unmeasured cell is often a difficult task due to the complexity of the phenomena and the influence of multiple parameters on pollutant dispersion and concentrations. To cope with that, some solutions use the available observations to predict the pollutant concentrations within the target cell. These techniques fall under the name of spatial interpolation, which is a mathematical technique that can be used to study a phenomenon that spreads over a region. Another solution that can be considered is known as data assimilation, which consists of combining sensor measurements with prior estimations (usually resulting from a physicochemical model), in order to provide a better prediction.

Spatial interpolation

The logic behind interpolation methods is to estimate the value of the target variable at an unmeasured point as a weighted combination of the available data [33]. In the example depicted in Figure 2.1, the black dots represent deployed sensors, while no sensor is deployed at the red point. Spatial interpolation uses the available knowledge on the black dots to predict the value of the red dot. Additionally, spatial interpolation methods often rely on extra information regarding the black and red dots. These extra parameters are referred to as explanatory (or independent) variables, and they could be any source of information that has an influence over the studied variable. In the

context of air quality assessment, explanatory variables could include concentrations of other pollutants, meteorological conditions, emission source locations, etc.

Spatial interpolation methods can differ in various aspects, such as the relationship between observations of the studied variable or between the studied variable and the explanatory ones, the nature of their prediction, spatial operating range, etc. Hence, spatial interpolation methods are usually classified in the literature according to their natures and assumptions.

- **Deterministic versus stochastic**: Deterministic approaches do not involve any randomness in the process, i.e., they assume certainty regarding the data and will always provide the same output given the same initial conditions. Conversely, stochastic methods consider that the target signal includes uncertainty due to random variations. The input data is then considered as a realization of a stochastic process. Thus, running the model with the same input is likely to get to a different result [48].

- **Global versus local**: Global techniques use all the available data in order to make predictions. On the other hand, local approaches use local information from either neighbors within a certain range around the unsampled point or simply k closest neighbors. The distance between data points can be spatial or computed on the basis of similarity between data points in terms of explanatory variables [49].

- **Univariate versus multivariate**: Univariate spatial interpolation methods do not involve secondary features and uses only the available samples of the dependent variable. On the other hand, in multivariate methods, the prediction is performed in function of a number of explanatory variables [50]. Note that in what follows, the focus will be on multivariate methods, since air pollution is a complex phenomenon that requires a significant number of variables to be explained.

- **Linear versus non-linear**: Linear methods assume a linear relationship between the target variable and the explanatory ones, while non-linear methods model the phenomenon using a more complex combination of the function parameters [50].

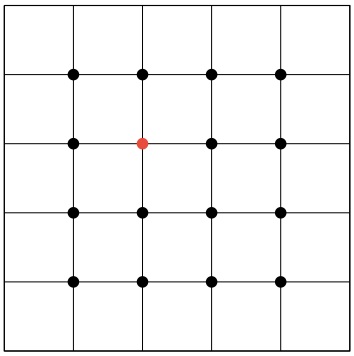


Figure 2.1: Illustration of spatial interpolation

In practice, adopting the suitable spatial interpolation technique for predicting values at unsampled points can be a difficult task, given the complexity of the phenomenon under study, its relationship with explanatory variables, the large size of

the datasets, their accuracy, and their spatial distribution [50]. Spatial interpolation methods are usually classified into two main categories, namely geostatistical techniques and non-geostatistical ones. Other methods can fall under a third category referred to as combined methods, which are combinations of spatial interpolation and other statistical methods [50, 51]. Geostatistical approaches assume that the data are spatially dependent. Hence, they take into consideration the spatial structure and variation of the data in order to make predictions [51, 52]. Geostatistical methods include multiple types of kriging [51, 52, 53]. In contrast, the non-geostatistical category does not account for the spatial dependency structure of the data.

Data assimilation

In contrast to measurement-driven spatial interpolation methods, other approaches combine *a priori* estimations (e.g., estimations from physicochemical models) with on-field measurement data in order to provide better prediction. In other words, the measured data are used to correct the *a priori* estimations usually provided by physicochemical models. These approaches fall under the category of data assimilation (DA). Data assimilation is a branch of mathematics that aims to capitalize on a theoretical model (e.g., dispersion model) and on-field observations to predict the estimated state of the studied system. DA originally came from the field of weather prediction and was used for meteorology until the 90s [54]. After that, it has been applied in many other fields, such as atmospheric chemistry and agronomy. The idea behind data assimilation is to use a theoretical model to predict the outcome of the system, then use measurements to correct the prediction. It is therefore important in data assimilation to find the right balance between the model and the collected data. In other words, find which one is more trustworthy.

In what follows, we give a brief theoretical background on kriging in Section 2.4, followed by a background on some interpolation methods using machine learning in Section 2.5. Subsequently, we present in Section 2.6, a selection of research work that used spatial interpolation (kriging and ML-based solutions) for air quality mapping. Thereafter, we provide more details on data assimilation in Section 2.7 along with some of its implementations in related studies in Section 2.8.

2.4 Background on Kriging

Kriging is a widely used geostatistical interpolation model for predicting the outcome of a function at an unsampled point using the available observations and the spatial relationship between them. Multiple types of kriging exist in the literature, such as simple kriging, ordinary kriging, and universal kriging [55]. We consider in the following ordinary kriging [56], which is one of the most common kriging models available. For a given point x_0 where no observations were collected, the predicted value $\hat{y}_0$ is given by a weighted average of the known observations as follows:

$$\hat{y}_0 = \sum_{i=1}^{m} \lambda_i y_i \tag{2.1}$$

where y_i is the observed value at x_i, m the number of points taken into consideration to estimate the value at x_0, and λ_i the weights attributed to the observations at x_i [53]. The basic kriging model is hence very similar to a regression one, but the errors are assumed to be spatially dependent. Here, the observed values y_i are

interpreted as realizations of random variables correlated between themselves. $\hat{y}_0$ is also interpreted as a realization of the unknown random variable at the point x_0. Let's note the error committed while estimating $\hat{y}_0$ at point x_0 as $\epsilon(x_0) = \hat{y}_0 - y_0$ where y_0 is the unknown ground truth at point x_0. The ordinary kriging assumes that $\epsilon(x_0)$ is unbiased (i.e., $E(\epsilon(x_0)) = 0$) which means that the weights sum up to one. i.e., $\sum_{i=1}^{m} \lambda_i = 1$ [56].

It is proved that the optimal estimator of the vector of weights λ, while minimizing the variance of the error at point x_0 $Var(\epsilon(x_0))$, is given as follows:

$$\lambda = A^{-1} \cdot B \tag{2.2}$$

where $\lambda = (\lambda_1, \lambda_2, ..., \lambda_m, \mu)^T$, $B = (\gamma(x_1, x_0), \gamma(x_2, x_0), ..., \gamma(x_m, x_0), 1)^T$, and A the $(m+1) \times (m+1)$ covariance matrix defined as:

$$A = \begin{bmatrix} \gamma(x_1, x_1) & \cdots & \gamma(x_1, x_m) & 1 \\ \vdots & \ddots & \vdots & \vdots \\ \gamma(x_m, x_1) & \dots & \gamma(x_m, x_m) & 1 \\ 1 & \cdots & 1 & 0 \end{bmatrix} \tag{2.3}$$

with μ a Lagrangian multiplier that forces the constraint $\sum_{i=1}^{m} \lambda_i = 1$ i.e., guarantees the unbiasedness condition [56].

As can be seen from Equations 2.2 and 2.3, the estimation of the weights λ_i requires the definition of the γ function. In kriging the γ is given by a so called variogram, which estimates the covariance between any two data points according to a certain distance metric. In other words, it is intended to show how data points become more similar as they get close to each other. The actual variogram has a cloud shape and is hard to calculate since we usually have only one realization of the studied random process. Therefore, a theoretical variogram is fitted to data. There are multiple variogram models (e.g., Gaussian, spherical, etc.) that can be used depending on the data. Figure 2.2 shows one example of a theoretical variogram which reaches a limit (know as the sill) after a certain distance d (known as the range), indicating that observation points that are more than d away will have a similar effect on the target point.

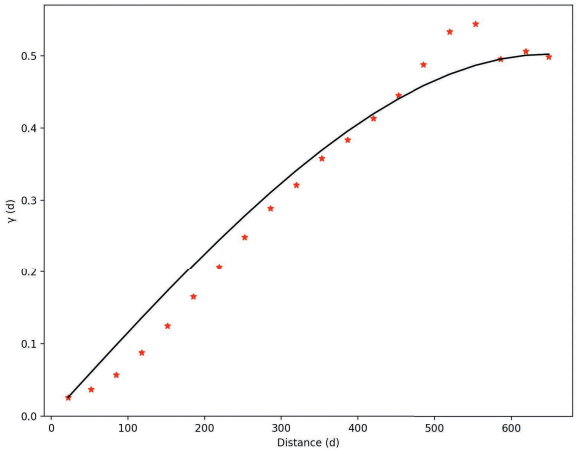


Figure 2.2: A spherical variogram model

In order to apply the kriging, the studied system is assumed to be stationary. This means the mean, the variance, and the variogram should be similar, regardless of the of chunk of data that is being considered. Moreover, the fact that kriging

estimates the weights at each target point separately is very useful to get an insight on where the interpolation is better over the region of interest. However, this leads to a high computational cost because for each target point, kriging needs to calculate the inverse of the matrix A which is large in general. To overcome this, one possible solution is to perform local kriging by limiting the number of observation points x_i to consider for each unsampled point x_0. This is also helpful when the stationarity is not assumed over the whole region of interest. More details on the kriging can be found in [55].

2.5 Background on machine learning-based spatial interpolation

Machine learning (ML) is a branch of artificial intelligence that combines both computer science and statistics in order to build systems capable of automatic learning from data and improving themselves through experience, without being explicitly programmed [57]. The goal of machine learning is to identify patterns in data in order to make decisions in the future. ML has made great progress in the recent years, thanks to the remarkable increase of memory handling for computers and the humongous amount of data generated every day. Machine learning approaches can be categorized into three families:

- **Supervised learning**: Supervised learning models require "labeled" data (input-output pairs) to train on, i.e., they require prior knowledge of what the output of the training data should be [58].
- **Unsupervised learning**: When an ML model is trained without any "labeled" data, it is called unsupervised. The model in this case tries to identify a pattern and classifies the data into groups. These methods are well suited for clustering data and finding anomalies [58].
- **Reinforcement learning**: Reward-based techniques that work on the principle of feedback. They learn which actions to take by combining both the predictions previously made and the feedback they got regarding these predictions, in order to maximize a numerical reward [59].

In this thesis, we mainly focus on supervised learning for two principal reasons. First, we are interested in the most common application in air quality monitoring, that is making spatial predictions. Second, we often have prior knowledge on the studied phenomenon, usually coming from deployed networks of fixed or mobile sensors, or historical predictions performed in the past.

Supervised learning techniques fall under two categories, depending on the type of prediction they are asked to perform. In fact, methods that predict a discrete output (e.g., nominal or binary) are called classification methods. As their name implies, these methods are given a set of predefined output classes, and their mission is to learn how to classify each input into its correct category [60]. One possible application is classifying zones of the map into "highly polluted" and "moderately polluted" zones, based on their pollutant concentrations, vegetation density, distance to highways, factories, etc. On the other hand, when the output value is continuous, the model is performing a regression, and hence, is called a regression model [60]. Regression is a statistical method that has been widely used in several fields, including air pollution

prediction [61, 41]. It is performed to predict the outcome of a continuous dependent variable (e.g., pollutant concentrations), based on collected measurements and/or the available knowledge on the explanatory variables. Regression can also be used to find which of the explanatory variables has the biggest impact on the target variable [33].

Training and validating a Machine Learning model

Different machine learning algorithms have been designed over the years, some are simple while others are quite complex. However, they all share the same key element, which is data. The quality and diversity of the available data highly impact the quality of the learning, and hence, the quality of the prediction an ML system is supposed to make. Indeed, training a ML model with few or redundant data leads to poor performance when the system encounters a new input. Therefore, the more and varied data, the better the model and therefore the greater the accuracy.

The training phase is of great importance in ML. It determines the quality of the prediction of a model, which can be measured by the error the model makes when dealing with a new data, i.e., prediction error (also called the generalization error). Since the prediction error is related to new data, it cannot be measured *a priori* during the model building phase. Therefore, it has to be approximated using a validation technique in order to evaluate the performance of the model. The foundation of validation techniques is to split the initial data into two groups, namely a training set and a testing set. The training set, as explained above, is the group of input-output pairs used to teach an ML algorithm. The testing set, on the other side, is used to assess and validate the performance of the model. These two sets are generally not the same size, with the training set having usually the largest amount of data. In the following, we give a brief overview of some ML validation methods:

- **Train/test split**: usually consists of randomly splitting the data into 70 to 80% for training, and 30 to 20% for testing, respectively. It is the simplest and easiest way to train and validate the model. However, it can lead to sampling bias if one of the training set has only similar data.

- **k-Fold Cross-Validation**: in order to minimize the sampling bias, this technique randomly splits the data into k folds. Then, it picks at each step $k-1$ folds for training, and tests the model on the remaining fold. This is done for all fold combinations, and the output prediction error is the average of the errors for each step. Thus, each sample will be used in both training and testing.

- **Leave-one-out Cross-Validation**: This is a variant of the k-Fold cross-Validation, in which only a single observation is left for testing while all the remaining data are used to train the model. In other words, this is a k-fold cross validation where $k = m$, and m being the number of observations in the dataset. It is clear that this method can be quite costly when the dataset is large, as the model needs to train m times.

In order to assess the prediction error of an ML model, plenty of metrics exist, and the choice of which one to use depends on the type of the model and the performed prediction. Indeed, some metrics are designed for classification models, such as confusion matrices. In regression, evaluation metrics are based on the residuals, which are a measure of how far data points (i.e., real observations) are from the regression line (i.e., predictions). Below, we present two most commonly used metrics for validating regression ML models:

- **Mean Absolute Error (MAE)**: is a well-known and commonly used metric in validation. It is the average of absolute differences between the real and predicted values. In other words, it tells how concentrated the data is around the line of best-fit. Thanks to the absolute value, MAE does not consider the direction of the residuals. In MAE, all residuals have equal weight. Its formula is given by:

$$MAE = \frac{1}{m}\sum_{i=1}^{m}|y_i - \hat{y}_i| \tag{2.4}$$

- **Root Mean Squared Error (RMSE)**: is one of the most commonly-used metrics, along with MAE. It measures the average of squared residuals. Similar to the MAE, the "squared" nature of RMSE prevents positive and negative residuals from cancelling each other. However, squaring residuals before they are averaged gives higher weight to large residuals. This also means that the RMSE is significantly more affected by outliers than MAE, hence the importance of removing them at a prior stage. RMSE is given by:

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2} \tag{2.5}$$

MAE and RMSE are negatively oriented scores, meaning that the lower they are, the better the model is performing.

In what follows, we mainly focus on ML spatial interpolation methods that we have used as a part of this thesis. We present the main logic behind each method, while referring to relevant publications for further details, when necessary.

2.5.1 K-Nearest Neighbors

K-Nearest Neighbors (KNN) is one of the simplest supervised learning algorithms. It can be used for both classification and regression problems [62]. KNN is based on the logic of "similar things tend to be close to each other", or simply similar inputs should have similar outputs. KNN for regression uses 'feature similarity' to predict values of any data points. Similarity between the point to be predicted, and the observation points is defined according to a distance metric applied to independent variables. At the beginning of the algorithm, a number of neighbors k is fixed. Then, for a given point x_0 to be predicted, KNN considers the most k similar data points to x_0 using a distance similarity metric. The predicted value $\hat{y_0}$ is generally the average of the k observations and is given by:

$$\hat{y}_0 = \frac{1}{K}\sum_{i=1}^{K} y_i \tag{2.6}$$

where y_i is the observed value at point x_i. The value of k is of great importance and has to be set wisely. A large value means including more points in the estimation, causing high sensibility to noise. In contrast, fixing a small K reduces the number of points used in the regression, leading to overfitting problems [63].

2.5.2 Multiple Linear Legression

Multiple linear regression (MLR) is an extended version of the linear regression, that seeks to establish a linear relationship between the studied variable, called the dependent variable and the variables likely to explain it, called independent or explanatory variables [64, 65]. In air pollution mapping, MLR assumes that pollutant concentrations at a given location depend on multiple factors, including meteorological conditions (e.g., temperature, humidity, wind speed, etc.) and the surrounding land-use information (e.g., number of routes, distance to highways, population density, number of buildings and their patterns, etc.) [33]. It has a formula of the form:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p \tag{2.7}$$

where Y is the dependent variable, $X_1, ..., X_p$ the explanatory variables, p the number of explanatory variables, and $\beta_0, ..., \beta_p$ the function's coefficient to be estimated. It is clear that these coefficients are the main key to the formula, as poor estimation of these parameters will result in a poor regression model.

In machine learning, a commonly used algorithm to estimate the coefficients of the formula is the gradient descent. The goal of gradient descent is to estimate the coefficients of the function that, when plotted, passes through or near the available observations while minimizing an objective function. To achieve that, gradient descent uses the derivative of the objective function with respect to each parameter. It first starts from an initial guess of the coefficients, and then optimizes them at each step, until reaching an acceptable value. Its logic consists of taking big optimization steps at the beginning and smaller ones when close to the best solution. In order to control the size of the steps, gradient descent uses a parameter called the learning rate, which is related to the slope of the derivative of the objective function to minimize [66].

Generally, objective functions that are minimized are called loss functions. The role of the loss function is to evaluate the performance of the prediction model. It is a scalar value which indicates how good does the curve in terms of predicting the target signal [60]. In general, multiple loss functions exist, and their use depends on a number of factors, such as the type of both the algorithm and the problem itself. The most commonly used loss function in regression is the "Sum of Squared Errors" which is the sum of the squared residuals, i.e., the differences between the real observations and the predicted ones. Its formula is given as follows:

$$L(y_i, F(x)) = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 \tag{2.8}$$

where $F(x)$ the function that gives the predicted values, y_i is the real value observed at a point i, $\hat{y}_i$ the predicted value at the same point, and m the total number of observations [64].

2.5.3 Random Forest

The random forest (RF) algorithm, as the name implies, is a machine learning algorithm made out of multiple decision trees, that can be applied for both classification and regression [67]. Decision trees basically make a series of sequential questions in order to take a decision. They are a simple and easy to interpret tool, especially when fitting a straight line to the data is not useful, due to the non-linear nature

of the studied variable. Decision trees are built from the top down, where the top node is called "the root" while the nodes at the very bottom are called "leaves". In regression, each leaf of a decision tree (also called regression tree) is a numeric value that corresponds to the average of the observations within the leaf. To build the tree, we must decide how to split the observations. For that, we try to find the threshold that minimizes the sum of squared residuals. This threshold becomes the split point and results in two groups. Then we move to the next level, and repeat the splitting process with the resulting two groups. We usually stop splitting observations if the size of the group is less than a predefined value. Although being easy to interpret, decision trees lack stability and struggle to generalize their learning when dealing with new samples (i.e., have high variance) [68]. Indeed, because of their hierarchical nature, the error made in a split at one level affects all the next splits. Random forest takes advantage of the simplicity of decision trees and fills the gap in flexibility by combining multiple randomly-created decisions trees for making a prediction.

Given a dataset D, for each decision tree to create, RF builds a bootstrap dataset B of the same size as D by randomly selecting samples from D, with the possibility of selecting the same sample more than once. The samples that are left out form the "out-of-bag dataset". Then, at each step (or level) of the decision tree creation, it only considers some explanatory variables to figure out how to split samples at each node. Therefore, considering a new bootstrap sample for each decision tree with a new subset of independent variables at each step results in a wide variety of decision trees, hence, making RF more effective than an individual predictor. Finally, to make a prediction for a new sample, RF runs the latter through all of its decision trees and outputs the average value of their predictions. This process of bootstrapping the data and aggregating decisions is called "Bagging" [67].

Usually, multiple random forests are created, with each one using a fixed number of explanatory variables to be considered at each level of a decision tree creation. Subsequently, each random forest will run every sample of the "out-of-bag dataset" through the trees that were not built using that sample. The prediction residual (also called the "out-of-bag error") is then calculated for each sample. Ultimately, the average prediction error is then calculated for each random forest, and then the most accurate model is chosen [67]. For more details, the reader can refer to [68].

2.5.4 Gradient Boost

Gradient Boost is an algorithm that combines predictions from multiple regression trees, similar to random forests [69]. However, each new tree is built based on the error made by the previous one, hence, building the model iteratively. Gradient boost assumes that taking lots of small steps in the right direction, results in better predictions. Thus, it scales the trees by a learning rate (a numeric value between 0 and 1). The role of the learning rate is to reduce the effect of each tree on the final decision in order to improve accuracy on the long-term [69].

Given a dataset, gradient boost, unlike random forests, starts by building a tree composed of one single leaf that represents the average value for all samples as an initial prediction. Then, it computes the residuals of all samples. Following that, it builds a regression tree to predict the residuals instead of the target variable and scales the tree using the learning rate. Therefore, the second tree is built based on the errors made by the previous one, and the new residuals are computed. gradient boost continues building trees using this logic until the maximum number of trees is reached, or no significant improvements are achieved with new trees [70].

2.5.5 eXtreme Gradient Boosting

eXtreme Gradient Boost (XGBoost) is a powerful ensemble learning algorithm that is based on gradient boost and was designed for large, complicated datasets [70]. It first starts with a single leaf representing an initial prediction. Then, it calculates the residuals and fits a regression tree to those residuals. However, the first difference with gradient boost, lies in the construction of the regression trees. For the second tree, XGBoost starts with a single leaf that contains all the residuals. Following that, it calculates for that leaf its output value and similarity score (also called quality score). The latter measures the resemblance between samples within the same leaf. XGBoost does a lot of optimization and approximation to solve for the optimal output value and the similarity score [71], hence the name "extreme". The simplified equations for the output value and the similarity score are respectively given by:

$$output_{jg} = \frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)}{m + r} \tag{2.9}$$

$$Sim_{jg} = \frac{(\sum_{i=1}^{m}(y_i - \hat{y}_i))^2}{m + r} \tag{2.10}$$

where m is the number of samples within the leaf j of the tree g, and r a regularization that prevents overfitting. The next step is to find the best threshold to split residuals into two leaves (i.e., moving from a tree with a single leaf to a tree with a root and two leaves). XGBoost tests multiple thresholds and creates a candidate tree for each configuration, while calculating the similarity score of each leaf. Then, it compares the gains of all candidate trees and keeps the one with the largest gain. The equation of the gain is as follows:

$$Gain = Sim_{left} + Sim_{right} - Sim_{root} \tag{2.11}$$

with Sim_{root} the similarity score of the root, Sim_{left} and Sim_{right} are the similarity scores of the left and right leaves, respectively. XGBoost continues clustering similar residuals (i.e., turning leaves with more than one residual into internal nodes) until it reaches the maximum depth. Another, optimization feature of XGBoost is pruning, which consists of visiting the tree from bottom to top and comparing the gain of each node to a fixed value. If the gain of a branch is less than this value, then the branch is removed (i.e., the node is turned to a single leaf containing the residuals of its previous leaves), and the algorithm continues to the next branch until no pruning is possible, or the tree has been completely removed [71].

2.5.6 Multilayer perceptron

Multilayer perceptron (MLP) is a feed-forward artificial neural network that is mainly designed to solve supervised learning. It consists of a system of fully interconnected layers, each having one or more nodes (or neurons). The first layer is called the input layer, whose only mission is to pass the input vector to the network without modifying it, hence it has as many nodes as there are input variables. The last layer is the output layer that represents the final output of the model, with as many neurons as the dimension of the output vector. In an MLP, each node is connected through weights to every node of the next layer. Between the input and output layers, a multilayer perceptron has one or more internal layers known as hidden layers, in which the output of each neuron is a function of the sum of the weighted inputs to

the neuron, modified by a linear or non-linear differentiable activation function [72]. An example of the architecture of an MLP is illustrated in Figure 2.3.

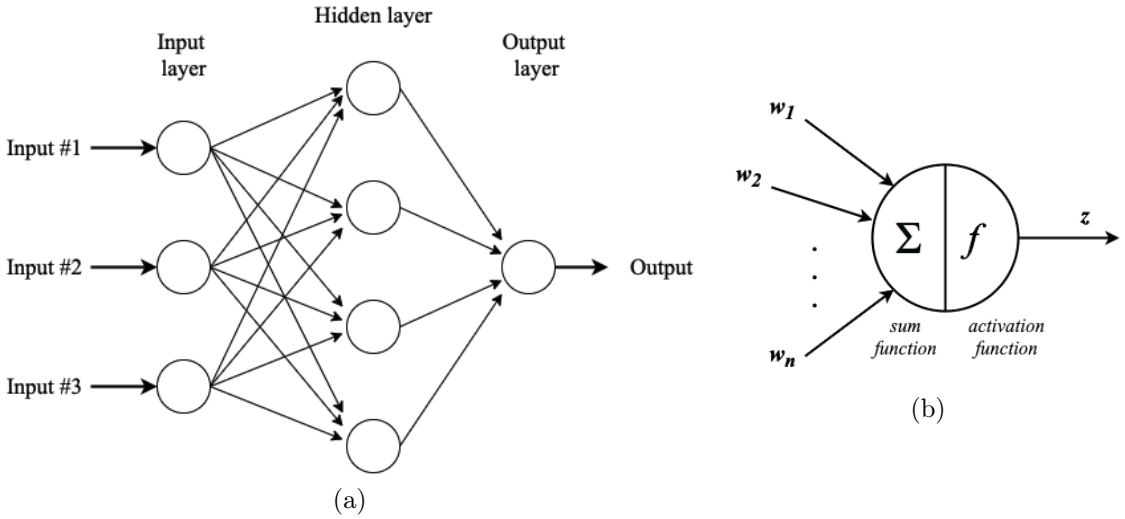


Figure 2.3: (a) A multilayer perceptron network with one hidden layer. (b) architecture of one neuron

Similarly to other supervised learning methods, MLP trains and learns with labeled data. Training the model consists of estimating the values of the weights connecting the neurons, such that the overall error of the model is minimal. The sum of squared residuals given by Equation 2.8 is one of the most commonly used loss functions for training artificial neural networks. A commonly-used algorithm for training an MLP is the back-propagation algorithm combined with gradient descent.

Considering an MLP network with only one node in each layer, let w_L be the weight connecting the last layer L to the previous layer $L-1$, and C the loss function defined by the following expression:

$$C = (a^{(L)} - y)^2 \tag{2.12}$$

where y is the desired output value, and $a^{(L)}$ is the predicted value i.e., the output of the activation function of the last layer, which is defined as:

$$a^{(L)} = f(z^{(L)}) \tag{2.13}$$

and $z^{(L)}$ the input of the neuron at layer L and is given by:

$$z^{(L)} = w_{_L} a^{(L-1)} \tag{2.14}$$

with f an activation function. From Figure 2.4, we can see that the cost function C is directly influenced by the activation function of $a^{(L)}$ which in turn depends on the weight $w_{_L}$ and the activation function $a^{(L-1)}$ of the previous layer (the reader can refer to [73] for details).

The back-propagation algorithm consists of computing, for a single input-output pair, the gradient of the loss function with respect to the weights, starting from the last to the first layer. In other words, it evaluates how sensitive the loss function C is to the weight $w_{_L}$, by calculating the derivative $\frac{\partial C}{\partial w_{_L}}$. Similarly, the influence of the activation function $a^{(L-1)}$ of the previous layer on the error C (i.e., $\frac{\partial C}{\partial a^{(L-1)}}$) is calculated. The algorithm keeps iterating backwards to evaluate the influence of

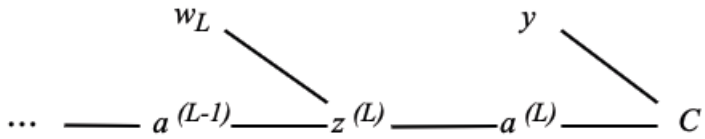


Figure 2.4: Illustration of the influence of weights and activation functions on the error of an MLP

previous weights on the cost function, hence, the name back-propagation. Gradient descent uses the computed gradients for all input-output pairs of the training set to update the weights and minimize the loss function [74, 60].

Other complex and more capable architectures of artificial neural networks, such as convolutional neural networks and recurrent neural networks, have been introduced in the literature for handling complex relationships with a temporal dimension [75, 76]

2.6 Air quality mapping using spatial interpolation

Spatial interpolation methods have been largely used in the literature, and several studies applied machine learning techniques to predict air pollutant concentrations.

Kerckhoffs et al. [61] proposed a multiple linear regression model for estimating ozone O_3 concentrations in the Netherlands, with a spatial resolution of $50m \times 50m$. Ozone concentrations were measured using passive samplers deployed at 90 different sites. All sites were measured simultaneously during four two-week campaigns, spread over the year. Explanatory variables used to build the model included, among others, traffic density, vegetation space, and the background of the sampled site. The measurements collected during the summer, fall, and spring campaigns showed good correlation with the existing continuous sampling network. In contrast, the correlation was poor for the winter campaign due to lower temperatures, according to the authors'. The developed model allowed to reach a determination coefficient of 0.77 for the annual average concentrations. Furthermore, O_3 concentrations showed a high negative correlation with NO_2 and particulate matters. However, the authors point out the need for developing a specific O_3 model, as the existing NO_2 model could only explain 46% of the variability in annual averages of O_3 concentrations.

An ordinary kriging model was employed in [77] for estimating $PM_{2.5}$ concentrations in Taiwan Main Island. $PM_{2.5}$ data were collected between July and October 2018 using remote sensing satellite, 75 monitoring stations, and a network of more than 2,000 deployed low-cost PM sensors. In order to create a single dataset, the three data sources were given a priority level (Monitoring stations, then low-cost sensors, then remote sensing) based on multiple factors, such as their reliability and spatial resolution. Therefore, the most reliable measurement was kept wherever readings from more than one source overlapped at the same location. The remote sensing had the lowest priority because its measurements are highly dependent on weather conditions, topography, etc. This resulted in a large amount of it data to be considered invalid. Cross-validation and RMSE were used to compare the performance of the spatial interpolation using the new dataset to those performed with each data source separately. The results highlight the benefit of combining multiple data sources, as the new dataset helped to capture local variations that were undetected by monitoring stations, thus, achieving a better mapping while maintaining an acceptable overall error.

Marjovi et al. [78], proposed two approaches for estimating the pollution level of UFPs (ultra fine particles) at desired time-location pairs in Lausanne, Switzerland. The first is a log-linear regression built over a virtual dependency graph based on land-use data. The second is a deep learning framework which can capture automatically the relationships between data, based on autoencoders. Different land use, meteorological and traffic data were used to build these models, among them, altitude, density of population, buildings, industries, etc. The two approaches were evaluated against three modeling techniques, namely Basic Log-Linear regression model, Network-based Log-Linear regression, and Basic Log-Linear regression with Land-Use. The results demonstrated the superiority of the proposed approaches and specifically the deep learning model over the other techniques.

A modified version of the Inverse Distance Weighting (IDW) spatial interpolation was employed in [79] for estimating monthly $PM_{2.5}$ concentrations between January 2019 and December 2020 in Taiwan, using a network of 74 background stations. IDW predicts the concentration at an unsampled location as a weighted sum of concentrations at sampled points. The weights are the inverse of the squared distances between the unsampled point and the sampled ones. However, considering all observations may include outliers which negatively impact the prediction. In order to improve the performance of the interpolation, a clustering-based IDW (CIDW) was implemented. In this vein, measurements from a network of more than 11,000 deployed low-cost sensors were utilized to create clusters of similar zones. Subsequently, each monitoring station was affected to its corresponding cluster. Therefore, for each unsampled point, the algorithm selects the cluster to which belong the nearest station. All stations within the selected cluster are then considered by the IDW. The performance of the proposed approach was compared to the original IDW and simple kriging for each of the 24 months. Using the leave-one-out cross-validation, the results revealed that CIDW considerably outperformed both the kriging and IDW for almost all 24 months, with an average RMSE that ranged between 1.17 and 3.86.

A mobile air quality sensing campaign was carried out in [80] across five predefined routes in Seoul, South Korea, using seven AirBeams, a low-cost and smartphone-based $PM_{2.5}$ sensor. A total of 10,177 collected data were combined with land-use information and used to compare the performance of three statistical models: multiple linear regression, random forest, and stacked ensemble. Stacked ensemble is a machine learning approach that combines predictions of an ensemble of multiple ML algorithms. Each ML algorithm of the ensemble is trained individually using the available data, then another ML algorithm (called meta-model) is trained using predictions of the previous ML algorithms in order to make the final prediction of the ensemble. In this work, the ML ensemble comprised 5 algorithms, including random forest, KNN, and recursive partitioning and regression trees. The random forest algorithm was selected again to serve as the meta-model. The results of the 10-fold cross-validation showed good performance across all models, with stacked ensemble achieving the lowest error ($RMSE = 5.22$), outperforming both random forest ($RMSE = 6.2$) and multiple linear regression ($RMSE = 7.01$).

Data collected from a series of four mobile sensing campaigns between 2016 and 2019 were involved in the prediction of temperature over the city of Lyon, France [81]. The primary source of data was two types of low-cost sensors, one measuring temperature and relative humidity, and the other measuring temperature. Remote sensing, meteorological stations, and other sources provided additional information about the influencing parameters, which resulted in a total of 38 explanatory variables. Spatial interpolation was performed by three different models, namely multiple linear

regression, partial least squares regression, and random forest. The outcome of the study revealed the superiority of random forest with a $RMSE = 0.17$ and a coefficient of determination equal to 0.95, compared to the multiple linear regression and partial least squares regression, which had a significantly lower coefficients of determination.

Hasenfratz et al. [41] utilized mobile sensor nodes to collect UFP data. Sensors were installed on top of public transport vehicles in the city of Zurich, Switzerland. Based on the collected measurements and land-use data, a land-use regression model was developed to create pollution maps with a spatial resolution of $100m \times 100m$. Twelve explanatory variables representing the traffic, the population and the city's characteristics were examined to build the air quality model for UFP. The accuracy of the land-use regression model across various time scales was compared. Results showed that low temporal resolution mapping achieves good results, while high temporal resolution estimation presents higher errors. To increase the accuracy of the developped model, past measurements were used in the modeling process, which permitted to reduce the root mean squared error by 26%.

2.7 Background on data assimilation

Given a dynamic system represented by a vector $\mathbf{x}^t$ (t stands for ground truth) of dimension n, the model takes into consideration multiple physics of the system (i.e., variables) in order to provide an *a priori* estimation vector $\mathbf{x}^b$ (b stands for background) of dimension n. However, the complexity of the studied system prevents the model from being perfect. Indeed, the constructed model is often incapable of capturing all the physics of the system, and is just an approximation to its true state. Moreover, we have a vector of observations $\mathbf{y}^m$ of dimension m about the state of the system, constructed from measurements of deployed sensors, for instance. The number of observations is in practice much smaller than the number of points of the model. Taking into consideration the background state and the collected measurements, the goal of data assimilation is to provide a new output that will be as close as possible to the ground truth. This output is called the analysis state $\mathbf{x}^a$ and is given by:

$$\mathbf{x}^a = \mathbf{x}^b + \Delta\mathbf{x} \tag{2.15}$$

where $\Delta\mathbf{x}$ is the correction of the background state $\mathbf{x}^b$. However, since the observations are often not perfect either and present errors, it is difficult to decide which of the model or the observations should be trusted more. To cope with that, data assimilation aims at minimizing the average difference $(\mathbf{x}^a - \mathbf{x}^t)$ between the analysis state and the ground truth state, hence, the need to characterize the errors of the background state and the observations state.

Without loss of generality, let the error vector $\epsilon^b = \mathbf{x}^b - \mathbf{x}^t$ be the error of the theoretical model (i.e., the background state), $\epsilon^m = \mathbf{y}^m - \mathbf{H}\mathbf{x}^t$ the error vector of the observations, and $\epsilon^a = \mathbf{x}^a - \mathbf{x}^t$ the error of the analysis state. The mean values of these error vectors are called the bias and are noted $\overline{\epsilon^b}$, $\overline{\epsilon^m}$, and $\overline{\epsilon^a}$, respectively.

Given the aforementioned vectors, we can build the error covariance matrices:

- $\mathbf{B} = E((\epsilon^b)(\epsilon^b)^T)$: the covariance matrix of the background state error, of dimension $n \times n$
- $\mathbf{R} = E((\epsilon^m)(\epsilon^m)^T)$: the covariance matrix of the observation state error, of dimension $m \times m$ with $m < n$

- $\mathbf{A} = E((\epsilon^a)(\epsilon^a)^T)$: the covariance matrix of the analysis state error, of dimension $n \times n$

where the variances of each vector are represented by the diagonal of its corresponding covariance matrix.

Best Linear Unbiased Estimator

One of the commonly used data assimilation methods is the Best Linear Unbiased Estimator (BLUE), that aims at providing an unbiased analysis vector $\mathbf{x}^a$ as a linear combination of $\mathbf{x}^b$ and $\mathbf{y}^m$, which are assumed to be unbiased too, i.e., $\overline{\epsilon^b} = \overline{\epsilon^m} = 0$ and decorrelated, i.e., $E((\epsilon^b)(\epsilon^m)^T) = 0$ [82, 83]. Based on these elements and assumptions, the analysis vector $\mathbf{x}^a$ is given by:

$$\mathbf{x}^a = \mathbf{x}^b + \mathbf{K}(\mathbf{y}^m - \mathbf{H}\mathbf{x}^b) \tag{2.16}$$

where $\mathbf{K}$ is the gain, the term $(\mathbf{y}^m - \mathbf{H}\mathbf{x}^b)$ is called the innovation, and $\mathbf{H}$ a $m \times n$ matrix that allows mapping the model space to the space of observations.

The mission of BLUE is to find the optimal gain $\mathbf{K}^*$ that minimizes the trace of the covariance matrix of the analysis error vector. Accordingly, $\mathbf{K}^*$ is then given by:

$$\mathbf{K}^* = \mathbf{B}\mathbf{H}^T(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{R})^{-1} \tag{2.17}$$

It is then possible to estimate the analysis vector $\mathbf{x}^a$ using the optimal gain. Hence, the name "Best Linear Unbiased Estimator" [84].

Several other data assimilation methods exist in the literature, such as Kalman filters which propagate the error variance of the analysis state in time, and extended Kalman filters, which are designed for non-linear models [83, 84].

2.8 Air quality mapping using data assimilation

In the field of air quality, several data assimilation approaches have been implemented in the literature. Observations from either highly accurate monitoring stations or dense low-cost sensor networks were assimilated to assist simulation models and provide a better estimation of pollutant concentrations.

Tilloy et al. [85] applied the BLUE method to assimilate observations from 9 fixed monitoring stations into simulations of NO_2 concentrations at urban scale, across the city of Clermont-Ferrand, France. Monitoring stations provided hourly NO_2 concentrations while the simulated concentration were generated every three hours for the whole year of 2008, using a dispersion model called ADAMS Urban. BLUE was applied every three hours, when a new simulation was available, and the performance of the model before and after assimilation was evaluated using the leave-one-out cross-validation. The results showed that data assimilation solved the problem of concentrations overestimation that occurred when using the model alone. The cross-validation revealed that BLUE helped to increase the correlation (up to 0.95) between the monitoring stations and the prediction. Moreover, the RMSE of the prediction was lower for each monitoring station, with an improvement of up to 46%. However, the improvement was considerably low (down to 5%) for monitoring stations that were far from the rest of the network, which might be related to the modeling of the error covariances.

Nguyen et al. [86] compared three data assimilation approaches for correcting daily NO_2 simulated concentrations of the year 2008, using a network of 16 fixed monitoring stations in the city of Lyon, France. The three considered DA methods were BLUE, the Bias Adjustment Technique (BAT) which seeks to eliminate the bias from the background state vector, and the Source Apportionment Least Square (SALS) which assumes that correcting the prediction of the numerical model is mainly related to the correction of the estimation of emission sources. The quality of prediction of the considered DA at locations where no measures are available was evaluated using leave-one-out cross-validation. For each monitoring station S, the three DA approaches used the observations from the other 15 stations to correct the simulated data of the numerical model (SIRANE). The predicted concentrations were then compared against the actual observed data. The evaluation metrics included the bias and the RMSE, and showed similar results for all considered DA methods, with a slight advantage for BLUE. Moreover, the impact of the number of the considered monitoring stations was assessed, and the results of the research suggested that the quality of the prediction is not only influenced by the number of considered stations but also the spatial configuration of the chosen combination.

Measurements from a network of 73 fixed background monitoring stations were merged with a dispersion model's simulations for predicting O_3 and NO_2 concentrations in [87]. Simulation data were generated by the means of an air quality model (AURORA) across Belgium, for a summer and a winter month (June and December, respectively). Contrary to BLUE, the model was considered to be biased. Therefore, a bias-aware optimal interpolation algorithm was implemented for data assimilation. A leave-ten-out cross-validation was adopted for evaluating the performance of data assimilation by keeping observations of 10 background stations out of the assimilation process at each run (70 stations were used for validation over 7 runs). The validation results demonstrated that the data assimilation substantially outperformed the AURORA model in predicting O_3 and NO_2 concentrations for both selected periods, at almost all validation stations. The average RMSE decreased from 27.9 to 12.6 for O_3 and from 17.4 to 11 for NO_2, while the correlation increased from 0.4 to 0.8 and from 0.3 to 0.6, respectively.

An Ensemble Kalman filter data assimilation was performed in [88] for improving $PM_{2.5}$ concentration predictions generated by a chemical transport model. The simulations and observations datasets comprised $PM_{2.5}$ concentrations between February and March 2019 in Aburrá Valley, Colombia. Simulations were obtained from the chemical transport model (LOTOS-EUROS), and observations were collected using a network of 21 monitoring stations and a network of 255 low-cost fixed sensors. The Ensemble Kalman filter consists of generating an ensemble of n forecast states (equivalent to background states for BLUE) given a set of initial conditions. When observations are available, the forecast ensemble is updated into an analysis ensemble, which in turn, will be used as a forecast ensemble at the next time step. The final prediction of the DA is the mean of the forecast of the last ensemble. In this study, the initial forecast ensemble was generated from perturbations to the pollutant emissions. The study compared the performance of data assimilation using either data from the monitoring stations or from the low-cost sensors network, while keeping seven of the 21 monitoring stations for validation. Although both assimilated forecasts improved the forecast over the model alone, the low-cost sensors improved the model forecast further ($RMSE = 18.39$) over the assimilation using the 14 monitoring stations ($RMSE = 20.69$), owing to their high spatial density. Furthermore, a third assimilation experiment was conducted using only a subset of low-cost sensors

(155 sensors) that showed a degree of correlation greater than 0.8 when compared to monitoring stations. The result showed that selecting only high-quality sensors provided better correction of the model with a $RMSE = 17.46$.

In [46], a modified version of the extended Kalman filter (EKF) was utilized for data assimilation of measurements of particulate matter from 15 low-cost sensors, deployed in Sidney, Australia. The modified EKF assimilation method was adopted in order to take into consideration missing data from the deployed sensors in case of a communication failure, for instance. At each instant t, the state vector (equivalent to the background state) is first estimated based on the previous sequences of sensors data and the available information regarding the emissions of the pollutant. Then the analyzed state vector is estimated using the new sequence of observations and the estimated background state. The performance of the modified version of EKF was compared to the standard EKF by removing one sensor and predicting its data using the remaining nodes. The study dataset included $PM_{2.5}$ and PM_{10} concentrations between November 2019 and May 2020. The results obtained indicated that the proposed data assimilation method produces a more accurate estimation with an average $MAE = 1.42$ and Pearson's correlation coefficient equal to 0.99, compared to an average $MAE = 3.88$ and a correlation of 0.97 for the standard EKF.

2.9 Conclusion

In this chapter, we provided a literature review of some air quality monitoring systems that are related to our research. The projects evolved around the use of low-cost sensors and wireless communication to collect extensive data on different parameters. Afterward, we presented an overview of some commonly used techniques for spatial interpolation that we relied on during this thesis, while explaining their fundamentals and logic. We also explained what is data assimilation approach and why it is beneficial in air quality monitoring. Moreover, we presented the main existing research studies that used spatial interpolation methods or data assimilation for air quality assessment. This chapter provides a foundation and helps to understand the contributions that will be presented in the following chapters.

In the next chapter, we present the design, evaluation, and deployment of our air quality monitoring system, which is powered by small and mobile low-cost sensors equipped with a long-range communication technology.

Chapter 3

Design of low-cost sensor-based air quality monitoring systems

In order to bridge the gap between individual exposure and regional measurements, a fine characterization of air quality at local scales is needed. This has motivated the launch of 3M'Air ("Mobile Citizen Measurements and Modeling: Air Quality and Urban Heat Islands"), a three-year multidisciplinary project that aims to explore the potential of participatory sensing to improve the local knowledge of air quality and urban heat islands. The project aims to evaluate the added value of moderately accurate pollution data generated by a non-scientific community, in the fine-grained characterization of air quality and urban heat islands. 3M'Air seeks to present adapted and optimized approaches for participatory data analysis in order to generate fine-grained pollution maps, while taking into account the continuity in space and time of measurements. It also aims to involve the citizens in the scientific process of air quality monitoring. We believe that the interest of these participatory data will be stronger when combined with physical models and accurate reference monitoring stations.

3M'Air leverages several scientific expertises going from electrical engineering, computer science, fluid mechanics, urban climatology and sociology. With this respect, we designed as a part of this thesis a low-cost participatory monitoring platform, featuring lab-designed, modular, portable, and optimized sensor nodes. The designed platform mainly serves the air quality application while remaining generic for other applications such as urban heat islands.

In the first part of this chapter, we introduce our lab-designed participatory system, as part of the 3M'Air project. We first present the main objective along with the design guidelines that we adopted throughout this work. Subsequently, we describe the architecture of our system and provide details on its main components.

The second part is dedicated to the validation of our platform. We evaluate the measurement quality of our platform through a comparison with reference stations and sensors. Furthermore, we evaluate the impact of different configurations on the energy consumption of our nodes.

3.1 Objective and design guidelines

Based on the study and the results of the conducted survey presented in Section 2.2, our first motivation was to design a platform of ergonomic sensors that will be carried by citizens in the context of collective or individual measurement campaigns. The designed system mainly serves participatory air quality monitoring but is expected

to be easily adapted to other environmental applications. To achieve this goal, a number of recommendations had to be respected. Our main design guidelines can be summarized as follows:

Required parameters: In order to serve the application of air quality monitoring while remaining generic for other applications such as urban heat islands assessment (which are also addressed in the 3M'air project), the designed nodes had to measure the main parameters related to the studied phenomena. Therefore, we chose to monitor air temperature, relative humidity, NO_2, PM_1, $PM_{2.5}$ and PM_{10}.

Note that an extra step is sometimes needed to extract meaningful data from sensors, such as conversion from voltage to pollutant concentration for electrochemical sensors, or conversion from particle counts to mass for optical particle counters.

Data gathering: To enable remote monitoring, it was required that the deployed system enables sampling all the identified air and weather parameters, and gathering the collected data on a remote server. In addition, the system was supposed to offer more than one level of storage to cope with communication failures and increase robustness.

Sampling rate: In order to provide a decent spatio-temporal coverage of the area of interest, the developed sensor nodes had to perform measurements at an acceptable sampling rate. This parameter is of great importance and has to be easily configurable to suit the movement modes of the participants (e.g., bicycles, or vehicles, etc.). In our case, the priority was the use by pedestrians, meaning an average speed between and 0.83 and 1.57 m/s [89, 90, 91]. Therefore, by setting the default sampling rate to one every 20 sec, consecutive readings are less than 30 m apart, which gives an acceptable spatio-temporal resolution.

Autonomy and network lifetime: The sensor nodes were intended to be carried by people and without an external power source. With this respect, our goal was to deliver no less than 12 hours of autonomy. This does not just mean integrating a bigger battery, but also optimizing all the system components to offer better efficiency. It was also preferable to equip the sensor with an indicator light to reveal the status of its battery.

Modular design: One of the main guidelines in this work was to build a modular system, in which components can be replaced or extended without highly impacting each other. This can be achieved by opting for a component-based design, where the whole system is composed of multiple hardware and software components that can be removed or extended when needed.

Ease of use: Internet of things projects usually involve a large number of smart and connected devices that are supposed to simplify our life. Thus, their design should be adapted to the target users and the context of use. In this respect, it was essential to build small and lightweight sensor nodes. Furthermore, despite the complexity of these new equipments and the remarkable engineering work behind, they should not require technical or scientific knowledge to be used. It should be relatively simple and as intuitive as possible to turn on the node, to use it and to charge it.

Reliable measurements: One of the main performance indicators in such systems, is the reliability of the information they provide. Consequently, the systems should be adapted to real environmental conditions. Solar radiations have a significant impact on temperature and humidity measurements, but also on other pollutant measurements. Therefore, to protect the sensing probes from the effect of solar radiation, we decided to paint the nodes white and equip them with a solar radiation shield.

3.2 System architecture

The architecture of our monitoring platform is composed of four layers, namely the sensing layer, the transmission layer, the storage layer, and the end-user layer (see Figure 3.1). The motivation behind this architecture is to decouple the system components in order to increase interoperability and scalability. Below, we first give a brief overview of the different layers before providing more details and explanations.

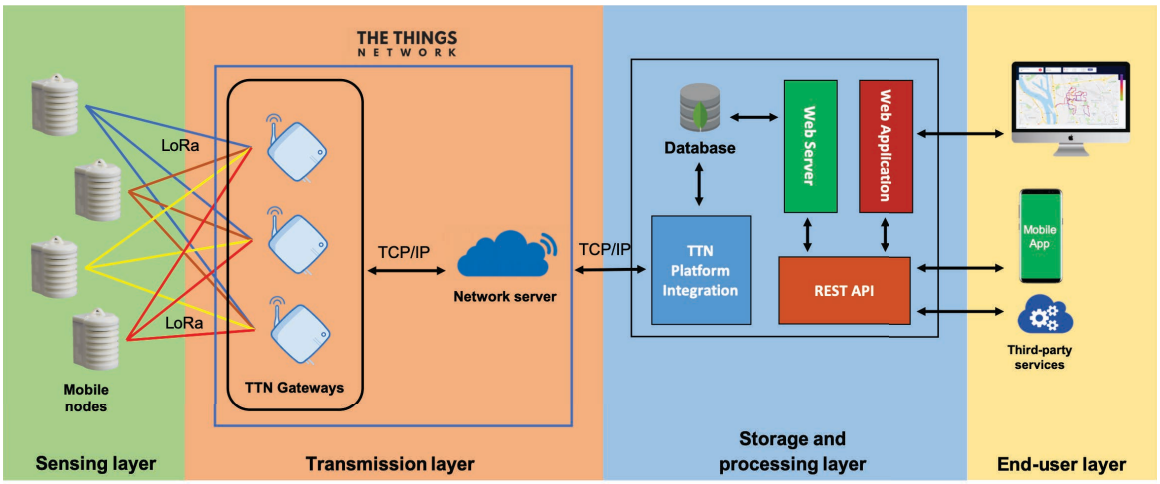


Figure 3.1: System architecture overview

The sensing layer: features low-cost, small-size, battery-powered, and portable wireless nodes equipped with three environmental sensors measuring PM_1, $PM_{2.5}$, PM_{10}, NO_2, temperature, and relative humidity. The nodes also include a power manager, a GPS receiver to geolocate measurements, a LoRa module for data transmission, a microSD card module to ensure data availability in case a node is outside the wireless network coverage, and an analog-to-digital converter (ADC) to enable compatibility with analog sensors. All the peripheral components are managed and orchestrated by a microcontroller.

The communication layer: relies on the LoRaWAN infrastructure and is responsible for data transport and nodes' authentication. LoRaWAN [92] is a communication protocol defined by the LoRa Alliance, which is an organization of more than 500 companies collaborating to promote the LoRaWAN open standard. This standard is one of the most promising LPWAN technologies for the Internet of Things [22, 93].

The storage and processing layer: is implemented on a cloud server. It receives data from the transmission layer and stores them into a NoSQL database for cleaning (i.e., correcting or removing inaccurate or redundant data from the database) and future processing. It offers a REST API through which measurement data can be queried using a web browser, a mobile application, or a third-party service.

The end-user layer: offers the possibility to visualize air pollution concentrations and weather parameters using either a mobile application, a third-party service, or a web interface.

In what follows, we cover each layer of our system, while detailing its functioning and motivating our main choices:

3.2.1 Sensor nodes

Our aim in this project was to design small and portable nodes based on low-cost sensors. Given the fact that these nodes are designed to be mobile and carried by people, it is important to have a suitable design to 1) ensure the node's functioning in mobility, 2) protect the node from solar effects while guaranteeing air flow, 3) maintain a reliable communication, and 4) make the nodes the lightest possible for users. We designed a casing with an integrated solar radiation shield and three separated chambers, as depicted in Figure 3.2. The casing was 3D-printed using a selective laser sintering process with Nylon 12 plastic. This material provides good strength and enough flexibility to withstand a fall without damaging the node. The 16 μm print resolution allows precise placement of the components as well as their attachment directly to the casing.

The first chamber regroups the environmental sensors and was designed to provide natural ventilation while protecting the sensing probes from solar radiations. For the sensors, we employ the *AlphaSense B-43F* sensor, which measures NO_2 concentrations. This low-cost electrochemical probe produces a current from the interaction of the target gas with an electrode [94] and has a sensitivity between -200 to -650 nA/ppm, as per the datas heet [95]. The *Grove HM3301* low-power laser dust sensor measures three sizes of particulate matters (PM_1, $PM_{2.5}$ and PM_{10}). This sensor is equipped with a fan driving the airflow inside a detection chamber, and is based on laser light scattering technology. It has an effective detection range between 1 and 500 $\mu g/m^3$ and a resolution of 1 $\mu g/m^3$ [96]. The last sensor is the *DHT-22*, a small low-power sensor based on a polymer capacitor for measuring temperature and relative humidity with an operating range of -40 to 80 °C and 0 to 100%, respectively, and a resolution of 0.1 °C and 0.1%, respectively [97].

The second chamber holds a two-layer printed circuit board (PCB) that we designed to integrate all necessary components. Our PCB incorporates an Arduino *MKR WAN 1300* built on the *Atmel SAMD21* low-power ARM microcontroller [98], the *Murata CMWX1ZZABZ* LoRa module [99], 8 digital pins, and 7 Analog input pins as depicted in Figure 3.3. An *MKR MEM ASX00008* shield was used to add a microSD card port and 2 MB extra flash memory. For managing analog sensors, a 16-bits *ADS1115* analog digital converter (ADC) [100] was added to have a better precision than the embedded MKR 12-bits ADC. Data geo-referencing is achieved by a low-power GPS receiver based on the MTK3339 chipset. This high-sensitivity module has 66 channels and can track up to 22 simultaneous channels [101]. In addition, the GPS module has a small battery, allowing it to save the current date and time without having to perform a data acquisition from the satellites. This is very helpful in situations where no satellite coverage is available (e.g., passing into a building or entering a metro station). The last component on the PCB is the *PowerBoost 1000C* [102], a small power manager based on a DC/DC boost converter chip that powers the system and is connected to the 5100 mAh LiPo battery placed in the third chamber. It is possible to charge the node by plugging it to an external 2A power source using an easily accessible micro USB port located at the bottom of the node. An additional micro USB port is also available for node programming and debugging purposes.

When the node is turned on, the microcontroller initializes the different modules and establishes an Over The Air Activation connection with the LoRaWAN server. Data are gathered from all environmental sensors and the GPS at 20-second intervals. In order to reduce the energy consumption of data transmission and storage while respecting the maximum payload of LoRaWAN packets, each node accomplishes three

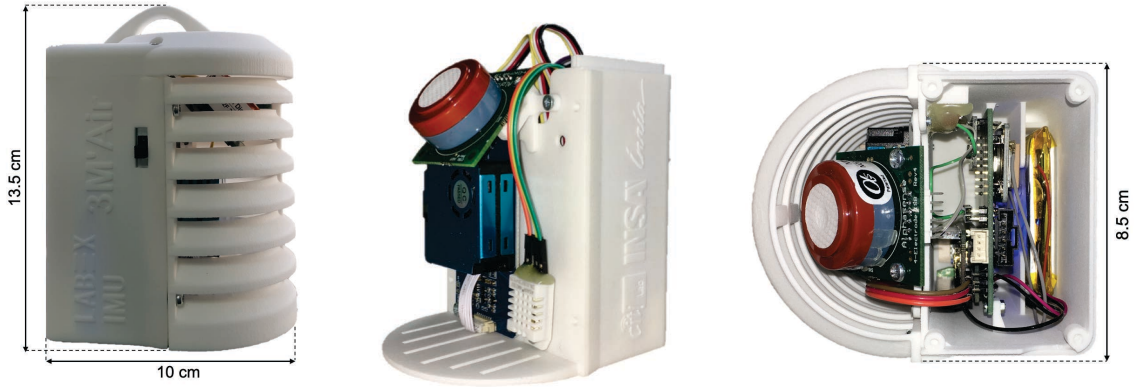


Figure 3.2: Design of the 3M'Air sensor node (a) side view of the designed sensor node; (b) Internal side view (sensor chamber); (c) Internal top view (second and third chambers)

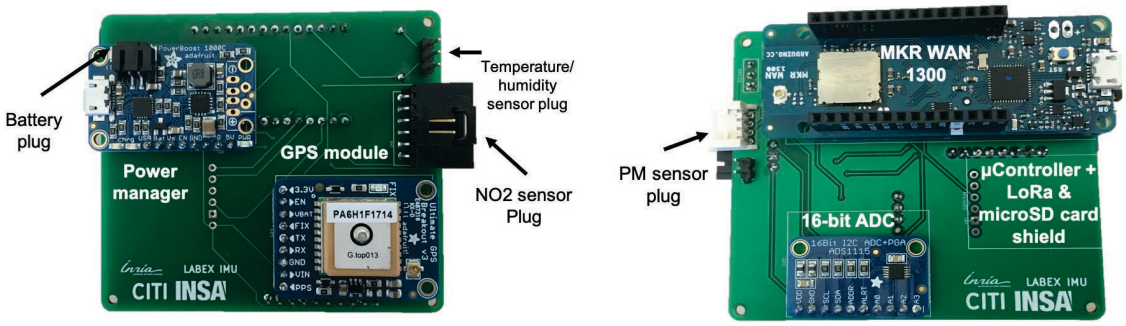


Figure 3.3: The node PCB with the different modules embedded

measuring cycles before storing the data on the micro-SD card and sending them to LoRaWAN gateways. The data record contains the ID of the sensor, the timestamp of the measurement, temperature, relative humidity, NO_2 PM_1, $PM_{2.5}$, PM_{10} values and the corresponding GPS coordinates. In case there is no GPS signal, the GPS coordinates will have the value 0, but we still can get the exact date and time from the GPS.

3.2.2 Transmission layer

The choice of communication technology must take into account the characteristics of our targeted use case, such as communication range, density of the network, transmission rate, energy consumption, etc. Therefore, we have chosen to adopt an LPWAN technology, such as NB-IoT, LoRaWAN, and SigFox which are the most notable technologies in this category. NB-IoT employs licensed spectrum in contrast to SigFox and LoRaWAN which use unlicensed spectrum over the industrial, scientific, and medical (ISM) band [22, 24]. However, the use of ISM bands restricts devices to a duty cycle limitation. In coherence with the choice of open technologies, we opted for LoRaWAN, especially since it respects our design guidelines more than NB-IoT and SigFox.

LoRaWAN relies on LoRa's physical layer, which uses Chirp Spread Spectrum (CSS) as modulation and offers the possibility to use eight configurations (data rates)

on the EU863-870MHz ISM Band, with six possible spreading factor values (SF7 to SF12) [103]. LoRaWAN defines the MAC layer, the architecture of the network, as well as the upper networking layers [22, 24]. The six possible spreading factor configurations use a 125 kHz bandwidth and offer a bit rate varying from 250 bps for SF12 to 5.47 Kbps for SF7. Two more configurations use respectively a 250 kHz bandwidth with SF7 achieving a bit rate of 11 Kbps, and FSK modulation offering 50 Kbps [103]. LoRaWAN specifies three classes of end-devices, related to the downlink (transmission from the server down to the end-device). Class-A must be implemented by all end-devices and consists of opening two receiving windows after each uplink (transmission from the end-device up to the server). Class-B and class-C are extensions of class-A and consist of either opening receiving windows periodically in the case of class-B, or keeping the receiving window open all the time (unless the end-device is transmitting), in the case of class-C [92].

In the proposed architecture by LoRaWAN, the gateways forward the packets sent by sensor nodes to the network server, which does the filtering of the data and makes sure that only one copy of the packet is sent to our application server to avoid data redundancy. The maximum payload size ranges from 51 to 222 bytes, depending on the selected SF value (51 bytes for SF12) [103]. Regarding the duty cycle limitation in Europe, most of the 868 MHz sub-bands have a duty cycle of 1%, thus, each node must not exceed 1% of spectrum occupancy .

On top of that, LoRaWAN offers two security modes: 1) Over The Air Activation mode (OTTA) where a join procedure is performed at the beginning in order to generate dynamic addresses and security keys, and 2) Activation By Personalization mode (ABP) where addresses and security keys are preloaded prior to deployment. In both modes, a Message Integrity Code (MIC) is added to each message to ensure that data has not been changed and is used by the nodes and the network server to ensure data integrity. Thanks to the payload encryption, confidentiality is also ensured between the nodes and the application server in both modes. We choose to use OTTA as it is based on dynamic keys, and handles the initial nodes' authentication in addition to the confidentiality and data integrity.

LoRaWAN can be implemented through operated, collaborative, or private networks. In operated networks, the user only manages the end-devices while the gateways, network servers and application servers are operated by telecommunication companies (e.g., "Orange" and "Bouygues Telecom" in France). It is also possible to implement a private LoRaWAN network by deploying private gateways and implementing the network and application servers. Collaborative networks are more open to the public by building a community-based solution, to which anyone can contribute, by registering a new gateway to the network.

The most prevailing collaborative and open-source LoRaWAN implementation is "The Things Network" (TTN), which is a contributor member of the LoRa Alliance and offers a free-to-use LoRaWAN network, with over 20,000 deployed gateways [104]. TTN manages the cloud infrastructure, while the gateways are mainly deployed by volunteers (including our lab) [105]. We selected this solution for mainly three reasons: 1) the participatory aspect of our platform that meets the collaborative nature of TTN; 2) its robustness and security through a large number of gateways and an end-to-end encryption; 3) its growing adoption in many fields all over the world with an increasing number of deployed gateways [106, 107, 108].

We formatted our packets to contain 51 bytes (3×17 bytes) which satisfies the duty cycle constraint using SF7, SF8 and SF9 with a 20-second sensing period and one-minute transmission intervals. We note that the value (3) is the number of cycles

and is also configurable. We also note that higher values of SF can be used by increasing the transmissions period and/or reducing the amount of data to send (e.g., compressing data, reducing the measurement precision, or sending averaged data).

In all our tests, we send confirmed packets (i.e., the application server has to acknowledge the reception of the packet). However, we do not implement retransmissions since data are stored locally on the nodes and the loss rate was very low. Indeed, thanks to the duty cycle, the collision probability is low when the number of nodes in proximity is not large [109].

When packets are received by TTN, they are forwarded to a router/broker service in the TTN infrastructure that will decode the message. The latter is then published to the right application handler, which is our cloud server. For further details on how the transmission layer works, please refer to the TTN documentation [104].

3.2.3 Storage and processing

In addition to the local storage, the large data volume generated by the sensor nodes needs to be stored on a remote server to ensure the data availability and enable remote access to the data. In order to catch the data sent by the transmission layer, we developed a Node.js script based on the integration solution provided by TTN, which offers a set of open tools to facilitate the development of IoT applications. Whenever a new measurement is received from the gateway, it is parsed and then stored on a NoSQL database (MongoDB) using Mongoose, an object modeling tool that provides a schema-based solution to model application data.

The cloud server is composed of two main components; the database component and the end-user services component. The latter is responsible for answering user requests from the end-used layer, through a REST API. In addition to that, multiple data processing techniques can be added such as detection of outliers, measurements correction, concentrations prediction, missing data reconstruction, etc.

3.2.4 Visualization

In order to visualize the sensor nodes measurements, we developed a web application that offers a simple and intuitive interface for visualizing pollutant concentrations and weather parameters. The front-end of the web application is built using HTML, CSS and Angular, which is an open-source component-based front-end framework for building large-scale single-page applications. It is built on TypeScript which relies on JavaScript ECMAScript6 and offers a lot of features that simplify web development such as dependency injection and component independence. The interface is based on Bootstrap 4 and Angular Material 7 for high-quality UI components that work across the desktop and mobile.

For experimentation purposes, we designed two front-end components, namely "Dashboard" and "visualization map", as depicted in Figure 3.4 and Figure 3.5. Through the first component, it is possible to display useful information such as the list of nodes with their IDs, end-device identifier (DevEUI) and last seen online time, etc. The dashboard also presents statistics about the contribution of each node to the application, as well as the total number of measurements or the number of measurements performed per month. All the information presented in the dashboard is actively updated in real-time to keep the user informed about the last changes. The Maps component allows the user to visualize air quality and meteorological conditions within the city by choosing the wanted date and parameter to visualize. All requests

generated from the web application are sent to the web server asynchronously to improve the user experience with non-blocking data loading and waiting times.

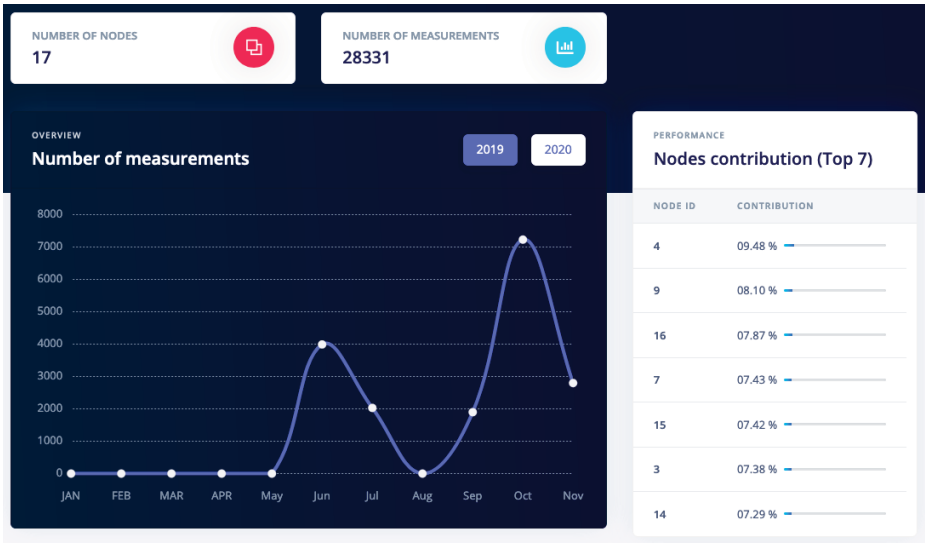


Figure 3.4: The web dashboard of the platform

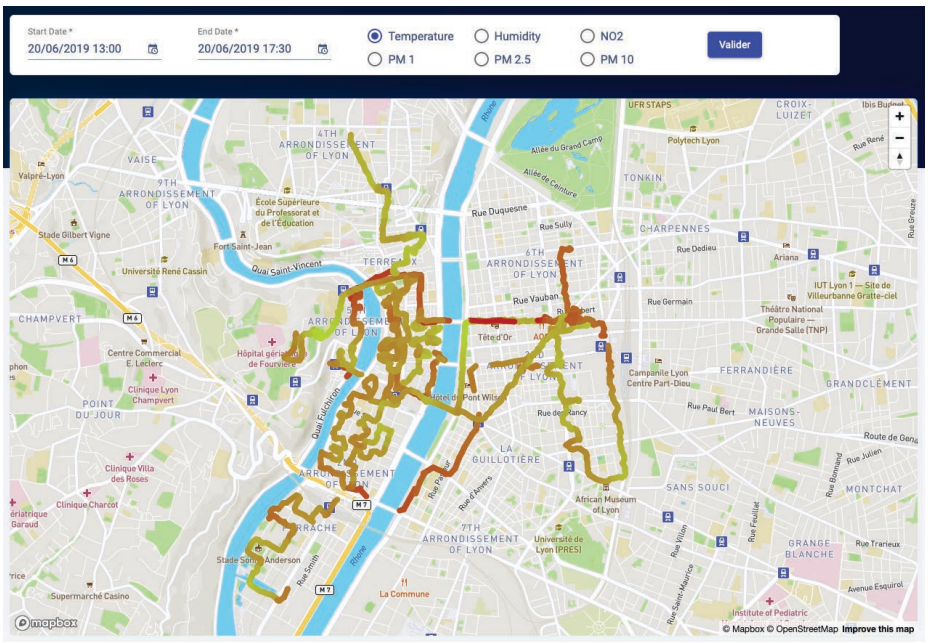


Figure 3.5: The web interface showing the measurements of the nodes on a map

3.3 Platform validation

Following the aforementioned guidelines and architecture, we built 16 mobile sensor nodes. The number of nodes was limited by our project budget. Prior to conducting measurement campaigns, we tested the proper functioning of the designed sensor nodes by: 1) comparing the sensor nodes to reference stations; 2) comparing the sensor nodes to each other; 3) evaluating their performance in terms of energy consumption. For this, we performed multiple tests, which we summarize in what follows.

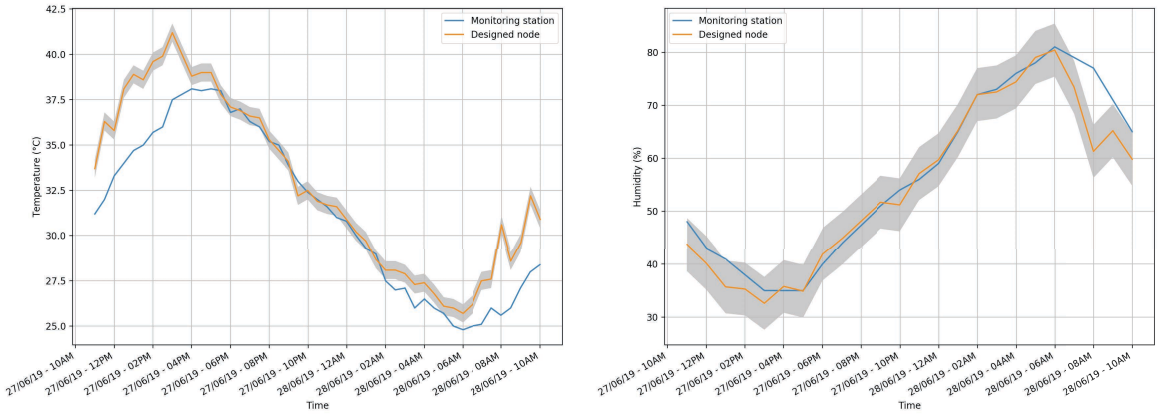
3.3.1 Comparison to reference sensors

To assess the accuracy of our nodes' measurements and the effect of environmental conditions, we tested them next to two reference devices: the first one was the "Météo France" monitoring station for temperature and relative humidity, and the second one was an approved fine dust measurement device (Fidas 200©) for particulate matters measurements.

Temperature and relative humidity

A metrological test was carried out with the help of our colleagues from the "Lyon 3" university in order to validate the measurements of the DHT22 sensor. Temperature and relative humidity were measured from June 27th, 2019 at 11am to June 28th, 2019 at 10am (local time), during the heat wave period of June, with a temperature that reached 38.10 °C at 5pm. The 3M'Air sensor node was suspended at a height of 1 meter in the instrumentation field of "Météo-France", Lyon (see on Figure 3.6), next to the monitoring station of "Météo-France". The reference sensors of the station were the PT100 for temperature and Vaisala HMP110 for relative humidity. The impact of direct solar radiation on the capabilities of the 3M'Air sensor was evaluated using two samples collected during two different periods. The first sample involved measurements taken from 11am to 9:30pm (sunset at 9:34pm) and from 6am to 10am (sunrise at 5:53am), with direct solar radiation. The second sample included measurements performed from 10pm to 5:30am without direct solar radiation [34]. Measurements from the 3M'Air node were then compared to those of the reference sensors. Since the data do not follow a normal distribution, the non-parametric Mann-Whitney test (also known as Wilcoxon-Mann-Whitney or Wilcoxon Rank-Sum test) was used [110, 111]. The coefficient of determination, the RMSE, and the bias were also calculated to compare the performance of the two sensors. Temperature and relative humidity measurements recorded during the whole period of sensing are plotted in Figure 3.7 for both the 3M'Air node and the monitoring station.

Figure 3.6: Our deployed node on the field of "Météo-France" in Lyon, France



(a) (b)

Figure 3.7: 3M'Air node vs reference station (a) temperature; (b) relative humidity

We can clearly distinguish two periods from Figure 3.7a, demonstrating the impact of solar radiations on the measurements. Indeed, the Wilcoxon-Mann-Whitney test indicated a significant difference (p-value = 0.021) in daytime temperatures, with 3M'Air recorded values (average temperature of $34.8°C$) higher than the reference sensors measurements (average of $32.7°C$). Significant differences were recorded punctually, with for example a difference of $4.30°C$ at 11am and $5°C$ at 8am. The correlation coefficient was 0.93, the $RMSE = 1.7°C$ and the bias was equal to $-1.9°C$. On the other hand, no difference was recorded for nocturnal temperatures ($p - value = 0.451, R = 0.99, RMSE = 0.4, bias = -0.5$). Regarding relative humidity, no differences were detected, neither for the daytime data ($p - value = 0.720, R = 0.97, RMSE = 4.3, bias = 1.4$), nor for the night data ($p - value = 1, R = 0.99, RMSE = 1.4, bias = 0.1$).

The differences obtained for the daytime air temperatures were expected because of the overheating of the shelter under direct exposure to solar radiation. The correlation coefficient for the temperature difference between the two sensors and the global radiation measured at the weather station was equal to 0.53. Indeed, although our node has an anti-radiation shelter, it is only open on 180° for design reasons (to keep the node small, the different chambers of the node were placed one behind the other as depicted in Figure 3.2). In addition, the small size of the sensor node may play a role in this observation. Nevertheless, the measurements of humidity and night temperature are, for their part, totally satisfactory. It is worth mentioning that the 3M'Air sensor nodes were not designed to perform static, but mobile measurements.

To evaluate the node's performance in mobility, six tests in four days during the period of July-September were performed with the 3M'Air sensor node next to two sensor nodes incorporating the Log32 sensor [112] inside two types of anti-solar radiation shelters (TFA [113] and DAVIS [114]). In two tests, all sensors were carried by participants walking on foot using straps, and in the last ones, the sensors were placed in a bicycle basket. All tests took place in the "Presqu'île" peninsula of Lyon city and measurements were taken every 20 seconds. Three non-parametric statistical tests were used on the data gathered by the sensors, namely Mann-Whitney, Komogorov-Smirnoff [115], Kruskal-Wallis [116].

The results showed divergence between the tests outputs. Indeed, this difference was due to the fact that unlike the 3M'Air anti-solar radiation shield, the TFA and DAVIS shelters are opened on 360 ° allowing air flow in all directions. In addition,

both sensor nodes with the TFA and DAVIS shelters are much bigger than our sensor node and do not include a circuit with a considerable number of components.

We also calculated the classical linear regression parameters (R, RMSE, and bias). For temperature measurement, the highest coefficient of determination we got was equal to 0.95 and the lowest was equal to 0.61 with an average of 0.82. The root mean squared error was between 0.4193 and 0.1631 °C and the measurement bias was equal to 0.37 °C. For relative humidity, The RMSE belonged to the interval [0.70 and 1.46]% while the coefficient of determination varied between 0.58 and 0.94. The measurement bias was around 0.93%. These results showed clearly that we were within the error ranges indicated for the DHT22 sensor [97], which made the use of 3M'Air nodes very satisfactory in mobile measurements of temperature and relative humidity.

Particulate matters

We evaluated the performance of the Grove HM3301 PM sensor embedded in our sensor node by testing it next to an approved fine dust measurement device called FIDAS 200© which is TUV Rheinland certified, and recognized by the LCSQA ("Laboratoire Central de Surveillance de la Qualité de l'Air") for monitoring particulate matter concentrations. The two sensor nodes were placed indoor, one next to the other, from October 4th, 2019 to October 9th, 2019. The series of measurements are highly correlated, with a low RMSE for PM_1 and $PM_{2.5}$, while performances degrades for PM10, as reported in Table 3.1. This may indicate that the composition of the largest PM in our region is different from that assumed when the sensor was calibrated. Another interesting observation is that the 3M'Air sensor slightly over-estimates the measurements in comparison to the reference device, as shown for $PM_{2.5}$ in Figure 3.8a. Considering the high correlation, the over-estimation can be compensated by an offline correction on the calibration function.

Metric	PM_1	$PM_{2.5}$	PM_{10}
RMSE	$3.86(\mu g/m^3)$	$5.784(\mu g/m^3)$	$7.92(\mu g/m^3)$
Pearson Correlation coefficient	0.98	0.92	0.63

Table 3.1: RMSE and correlation coefficient of PM_1, $PM_{2.5}$ and PM_{10} measurements taken by the designed node next to a reference device

Sensor calibration is needed to cope with the low accuracy and signal drifting of low-cost sensors. Indeed, calibration techniques for low-cost sensors have been extensively discussed in the literature. They are generally classified according to the availability of reference stations (reference-based, blind, or partially blind calibration), the mobility of the sensors (static, mobile, or hybrid), the calibration relationship (univariate or multivariate), and the used calibration model (e.g., ordinary Least Squares, multiple Least Squares etc.). [38, 37]

As a proof of concept, we implemented a univariate linear regression to calibrate our PM sensor's raw data based on the concentrations observed by the FIDAS sensor. In this process, we varied the learning duration and the temporal resolution to evaluate their impact on the calibration performance. Results show that with 6 hours of training on data with 1-minute temporal resolution, the RMSE significantly decreased (from 5.785 to 0.835 $\mu g/m^3$), which can be seen in Figure 3.8b. A training duration of 12 hours with the same temporal resolution allows achieving a even

lower error of 0.734 $\mu g/m^3$. Moreover, when decreasing the temporal resolution to 1-hour averages instead of 1 minute, the calibration achieves better performance with a RMSE decreasing from 5.735 to 0.558 $\mu g/m^3$ using 6 hours of data for training and 0.465 $\mu g/m^3$ when training on 12 hours data. Interested readers can refer to [38, 37] for more details on low-cost sensor calibration techniques.

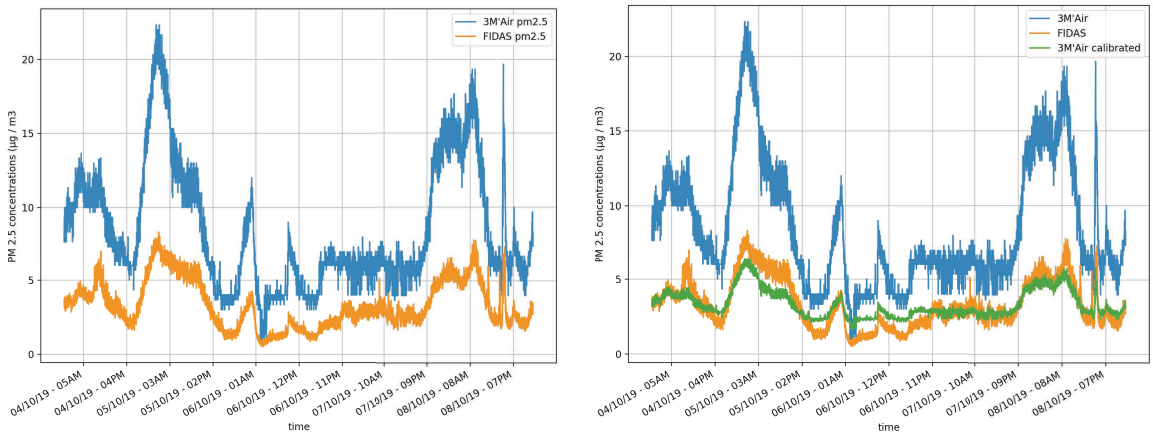


Figure 3.8: 3M'Air node vs reference station (a) raw measurements; (b) calibrated measurements

3.3.2 Cross comparison

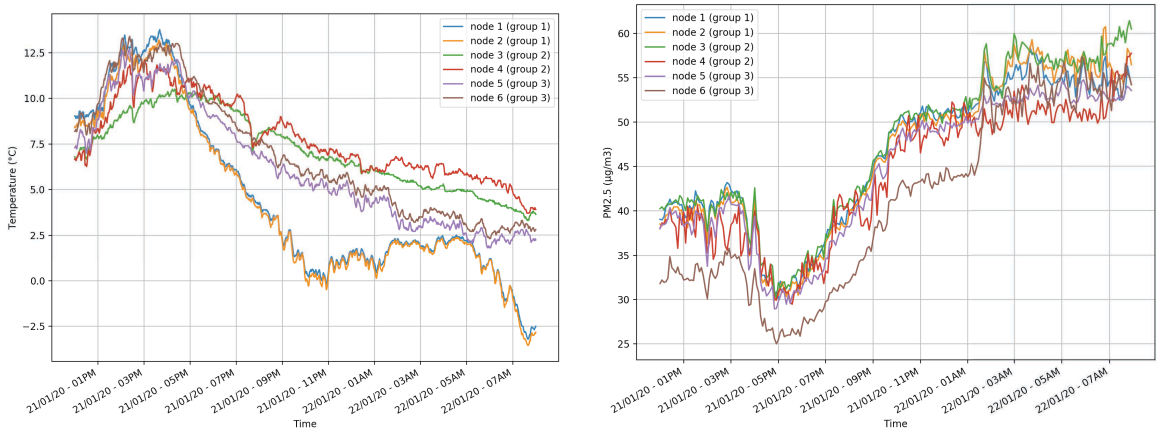
We also conducted other tests to compare the designed sensor nodes to each other. For this purpose, we deployed six nodes on the rooftop of a three-story building, forming three groups (two nodes per group). Each group of sensors was placed differently.

The reason behind this was to evaluate the impact of nodes' positions and orientations. The test was performed from January 21st, 2020 at 12pm to January 22nd, 2020 at 8am (local time). After aggregating data into one-minute averages, we calculated the RMSE and the correlation coefficient of Pearson within each group and also between in-averages of the three groups. Results are presented in Table 3.2

Temperature and relative humidity

We compared temperature and relative humidity measurements of every two sensors of each group. Figure 3.9a presents temperature measurements from all six sensors. The first observation that can be drawn from this figure is that each group recorded different temperature and relative humidity values. This was expected since the groups were placed in different positions and orientations, resulting in different sun exposure and wind direction. Nevertheless, there was a good correlation between the nodes of the same group, especially with temperature measurements. We also noticed that the difference in measurements between the groups still existed despite the absence of sun rays. This indicates that measurements can be impacted not only by the sun exposure, but also by the wind direction. Another observation that can be noted from Table 3.2 is that the RMSE of temperature and relative humidity values inside the same group sometimes exceeds the error margins claimed by the data sheets of the sensor. This is possible as these sensors are low-cost and may present in some cases larger errors. However, these values can be corrected, and the sensors can be calibrated regularly. On the other hand, the RMSE between groups stays high due to

the fact that each group had a different placement and orientation. All these results show that our nodes perform well and confirm again the impact of wind direction and angle of sun exposure on the sensors.



(a) (b)

Figure 3.9: Measurements from our designed nodes for (a) temperature (b) $PM_{2.5}$ concentrations

Particulate matter

In the case of particulate matter, Figure 3.9b presents $PM_{2.5}$ concentrations from the six nodes. For readability reasons, measurements have been aggregated into 30 min averages in this plot. However, all calculations were based on 1 min averages. We can observe that PM concentration values from all six nodes present the same trend and a good correlation during the whole period of the test, except for node 6 which recorded significantly lower concentrations than the others. Another interesting observation is that as the temperature decreases, the PM concentration tend to increase, which was also noticed in multiple studies [117, 118]

In addition, the results presented in Table 3.2 reveal that unlike temperature and relative humidity, the RMSE and the coefficient of correlation for $PM_{2.5}$ and PM_{10} concentrations have reasonable values regardless of the sun exposure of the nodes or their placement. Indeed, the correlation coefficient is identical between $PM_{2.5}$ and PM_{10} across all comparisons. It has to be noted that the distance separating the groups of sensor nodes was just few meters as they were deployed on the rooftop of the same building.

3.3.3 Energy consumption

Energy consumption is of great importance in low-cost WSNs as energy requirements differ from one application to another. For our case, we conducted multiple tests with different configurations to evaluate the power consumption and also to determine which configuration or sensor is power consuming. The first configuration was the default one, with all sensors performing sampling every 20 seconds and data transmission after 1 minute. In the second configuration, the sampling rate was divided by three, i.e., sampling and transmission every 1 and 3 minutes, respectively. The third configuration was the same as the default one, except that the PM sensor was unplugged. In the last configuration, the PM sensor was plugged back, but its fan was

	Group number	Temperature		Relative humidity		PM 2.5		PM 10	
		RMSE	Pearson's coefficient	RMSE	Pearson's coefficient	RMSE	Pearson's coefficient	RMSE	Pearson's coefficient
Intra-group	1	0.30 °C	0.99	10.17 %	0.99	1.28 $\mu g/m^3$	0.99	1.66 $\mu g/m^3$	0.99
	2	0.87 °C	0.96	5.11 %	0.95	4.11 $\mu g/m^3$	0.96	5.25 $\mu g/m^3$	0.96
	3	0.71 °C	0.99	13.65 %	0.99	4.73 $\mu g/m^3$	0.96	5.58 $\mu g/m^3$	0.96
Inter-group	1 – 2	4.21 °C	0.85	15.13 %	0.92	2.02 $\mu g/m^3$	0.97	2.53 $\mu g/m^3$	0.97
	1 – 3	2.84 °C	0.93	9.62 %	0.95	3.80 $\mu g/m^3$	0.98	4.77 $\mu g/m^3$	0.98
	2 – 3	1.75 °C	0.93	9.01 %	0.97	3.59 $\mu g/m^3$	0.97	4.49 $\mu g/m^3$	0.97

Table 3.2: RMSE and correlation coefficient for measurements of temperature, relative humidity and particulate matter concentrations

turned off. Table 3.3 reports the results of the test. As can be expected, the higher the sampling rate, the higher the energy consumption. The interesting observation is that dividing the sampling rate by 3, from one sample every 20 sec (config 1) to one every minute (config 2), hardly reduces the consumption by 8 mA, around 3.5%. It is worth mentioning that the sensors are not completely turned off between measuring cycles because the convergence time of some sensors is larger than the selected sampling rate. Indeed, the convergence time of the DHT22 is about 2 seconds [97], while the PM sensor needs at least 30 seconds after power-on to start giving reliable results [96]. On the other hand, the required convergence time is larger for the NO_2 sensor ($t_{90} < 80$ seconds, stated by the manufacturer [95]) and might reach around ten minutes, according to previous tests conducted by our team. One additional element is the energy consumption of the PM sensor. According to our results, turning it off (config 3) reduces the energy consumption by almost a half (from 231 mA to 115 mA). This is mainly due to its integrated fan used to aspire the airflow. This is confirmed by the last test (config 4) where only the fan of the PM sensor was kept off. The energy consumption in this configuration dropped to 154 mA compared to the default configuration, meaning that the fan consumes 77 mA while the electronics of the PM sensor uses 39 mA. Nevertheless, in order to obtain reliable measurement data, it is recommended to use the fan of the sensor.

Config	NO_2 sensor	Temp and RH sensor	PM sensor	Sampling rate	Transmission rate	Average Power Consumption
Config 1	X	X	X	20 sec	1 min	231 mA
Config 2	X	X	X	1 min	3 min	224 mA
Config 3	X	X	-	20 sec	1 min	115 mA
Config 4	X	X	no fan	20 sec	1 min	154 mA

Table 3.3: Comparison of different operating configurations

Based on these tests, we decided to keep the first configuration as the default configuration in our sensing campaigns, i.e., maintaining the sampling rate at 20 seconds and sending data every minute. This will guarantee a good temporal and spatial resolutions with an estimated lifetime of 22 hours using a 5100 mAh battery. We believe that this is an acceptable autonomy, since our participatory measurement campaigns lasted about two hours on average.

3.4 Conclusion

In this chapter, we highlighted the main motivation behind our air quality monitoring platform, based on the review of the related systems presented in Chapter 2. Next, we presented the four-layer architecture of our platform, while providing technical details for each layer. Although the primary use case is the use by pedestrians, our sensor nodes can be easily mounted on vehicles or bikes. In addition, the component-based logic we followed throughout this work allows the system to be extended to other applications. Furthermore, we performed various tests to validate the performance of our proposed solution prior to deployment. Our sensor readings were compared to each other and to other reference sensors. The results showed satisfactory performance in terms of correlation and measurement error. Finally, we evaluated the energy

consumption of our nodes using four different configurations, which revealed that the PM sensor is the most demanding because of its integrated fan.

The design of the 3M'Air monitoring system helped us understand the different aspects of air pollution sensors, such as the energy consumption and the impact of sensing rate and solar radiations on the sensing probes. In the following chapter, we first present a data analysis of the collected data during the sensing campaigns, and emphasize on the great potential of low-cost sensors in estimating air pollution and correcting numerical models. We also investigate the possibility of putting a sensing probe into sleep mode during time intervals and evaluate its impact on the sensing quality. Subsequently, we consider a large-scale deployment scenario and compare the performance of different spatial interpolation and assimilation techniques in estimating pollutant concentrations using low-cost sensors' data.

Chapter 4

Data analysis of WSN-based air quality monitoring systems

The design and validation of our air quality monitoring solution allowed the collection of extensive data through several mobile sensing campaigns that we co-organized with the help of the partners of the 3M'Air project. Thus, the goal of this chapter is to explore the benefit of using low-cost air quality sensors in air quality monitoring, by analyzing the collected data and drawing insights both on the mapping of the studied phenomenon and on the eventual optimization of the platform.

In addition to the collected data, air quality mapping often involves explanatory variables or physicochemical models as part of the data assimilation, as discussed in Chapter 2. In the first part of this chapter, we present the region of interest and the datasets that were considered during this thesis. These datasets include simulated data from numerical models, data collected using the 3M'Air platform presented in the previous chapter, and explanatory variables from various sources.

The second part of this chapter highlights the potential of low-cost sensors through a data analysis of pollutant concentrations collected during multiple sensing campaigns using the 3M'Air monitoring platform. We first compare the estimation quality of four statistical models and investigate the impact of sampling rate on the quality of estimation and energy consumption of the nodes using an energy model based on the sensing duty cycle. In addition, we evaluate the capacity of regression models to recover missing data of one sensor based on measurements from other sensors.

In the last part, we break the budget limit and consider a scenario with a larger number of sensors. In this vein, we reconsider and compare different regression approaches to data assimilation, while taking into account the intrinsic characteristics of dense deployment of low-cost WSN for air quality monitoring (high density, numerical model errors and sensing errors). We present a generic framework that allows the comparison of different strategies based on numerical simulations and an adequate estimation of the simulation error and the sensing errors.

4.1 Area of interest and datasets

4.1.1 Area of interest

During this thesis, we considered the agglomeration of Lyon, which is located in the region of "Auvergne-Rhône-Alpes" in the southeast of France. It comes third in the ranking of the largest metropolis in France, with over 1.4 million people over 533.6

km^2 [119]. The focus was particularly on a 5 $\times$5 km^2 area, mainly regrouping the first, second, fourth, sixth, districts. The area of interest also covered portions of the third and ninth districts of the city.

4.1.2 3M'Air platform data

The sensing campaigns' sites and participants recruitment were carried out in collaboration with the "La Métropole de Lyon" metropolis and the "Environnement Ville Société" (EVS) laboratory of Lyon city. The goal was to select an area at the heart of Lyon city and another one at its vicinity. With this respect, we chose the "Presqu'île" peninsula located in the heart of the city of Lyon and the region of Saint-Fons south of the city. For the participants' recruitment, we wanted to mobilize volunteers from different backgrounds with no particular technical knowledge in order to evaluate the ease of use of our nodes. In this vein, the "La Métropole de Lyon" broadcasted a call for participants through their means of communication for each measurement campaign.

Following that, we received a decent amount of applications for the measurement campaigns taking place on the "Presqu'île" peninsula of Lyon city, which allowed us to conduct 12 sensing campaigns between June 2019 and October 2019, with an average of 10 participants each, ranging from students from the "Lyon 3" university, members of the team, retired citizens, etc. Moreover, some sensing campaigns were conducted without the presence of vehicles, as part of the pedestrianization project of the peninsula. On the other hand, we received no applications for the region of Saint-Fons, which resulted in only one sensing campaign on July 2019. This lack of participation could be related to the fact that the issue of air pollution was not among the main preoccupations of the residents of this region. It is worth mentioning that due to the restrictions related to the COVID-19 pandemic, we were not able to organize more sensing campaigns.

Moreover, in order to keep a certain control over the sampling and maximize spatial coverage, the EVS laboratory provided predefined routes for participants (see Figure 4.1). In addition, some routes were affected to more than one node/participant for performance evaluation purposes.

The performed mobile sensing campaigns enabled measuring temperature, relative humidity, NO_2, and particulate matters concentrations across the streets of the "Presqu'île" peninsula of Lyon and its neighborhood during different periods of the year, resulting in over 11,000 data points.

4.1.3 Explanatory variables

In addition to the aforementioned data, more than 120 explanatory variables that characterize the city of Lyon were collected from multiple sources, as part of the Urpolsens project (2015-2018), funded by the Labex IMU. Meteorological conditions were provided by "Météo-France" [1]. Traffic and land-use information were obtained from Data Grand Lyon [2] and Open Street Map. For each point of the grid map representing the area of interest, these variables provided information about population density, land-use, meteorological conditions, road network, and traffic. Depending on its nature, each explanatory variable represented either:

[1] https://donneespubliques.meteofrance.fr/

[2] https://data.grandlyon.com/

Figure 4.1: An example of a pre-defined sensing path

- **the surface in a buffer:** considering a circular spatial buffer of a predefined radius, this type of variables gives the surface of a particular type of zone (e.g., residential zone) within the buffer;
- **the number in a buffer:** Similarly, these variables provide information about the count of buildings roads in a spatial buffer;
- **the distance:** gives the distance (in meters) between a point on the map and a certain infrastructure (e.g., distance to highways)

Following the preprocessing of the correlation between the variables, only 35% of the initial explanatory variables were retained. The variables are grouped in four classes, as can be seen in Table 4.1.

4.1.4 Numerical simulations

Given the need for numerical simulations to perform data assimilation, we obtained, for a proof of concept, annual averages of NO_2 concentrations in Lyon city in 2008, generated by the means of SIRANE, an atmospheric dispersion simulator widely used by the certified associations of air quality monitoring in France [10, 120, 121]. The dataset represented NO_2 concentrations in $\mu g/m^3$ on a 25 km^2 grid covering the city of Lyon and its immediate surroundings, as illustrated by Figure 4.2. The phenomenon

Categories	Example of Variables
Land-use	Number of buildings in a given buffer
Traffic and road networks	Number and length of roads in a given buffer
Population	Density of population in a given buffer
Meteorology	Temperature, humidity

Table 4.1: Main categories of dependent variables

was assumed to be stationary during the study period, hence the consideration of annual averages of NO_2. The grid was formed by 63,001 data points with a distance of 20 meters between each data point.

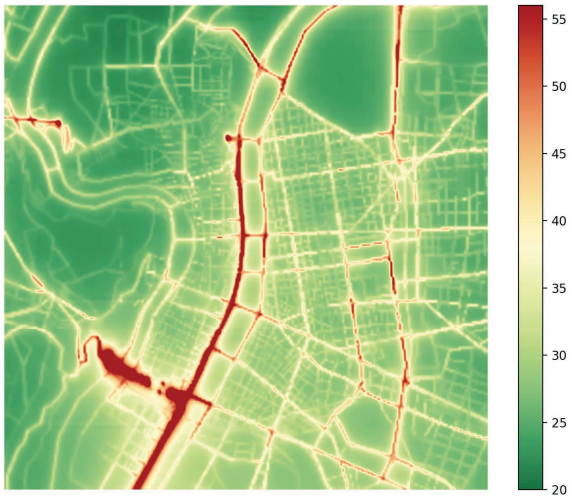


Figure 4.2: Annual averages of NO_2 concentrations for the city of Lyon in 2008 (numerical simulations)

4.2 3M'Air platform's data Analysis

The goal of this study is to explore the potential benefits of using low-cost sensors in estimating air pollution using regression models. We investigate the impact of the sampling rate of sensors on the performance of air pollution estimation and the possible impact of lowering the sampling rate on the autonomy of the node and the estimation error. In addition, we evaluate the capacity of regression models to recover/fill missing data of one sensor based on the other sensors of the network, which gives an insight on the overall performance of the system and the degree of correlation between sensors.

We consider in this part $PM_{2.5}$ concentrations collected during four sensing campaigns that were conducted on the "Presqu'île" peninsula of the city of Lyon. The sensing campaigns took place between the 20th of June 2019 and the 26th of October 2019 with a minimum of 8 sensors. nodes each

As explained in Chapter 2, estimation models often require extra features to explain the phenomenon and complement the measurements obtained by fixed or mobile sensors. With this respect, we used the dataset of explanatory variables introduced

in Section 4.1.3, in addition to the collected $PM_{2.5}$ concentrations. Since our nodes perform sampling at fixed time intervals (fixed temporal resolution) rather than predefined positions with fixed distance intervals (variable spatial resolution), the distance between two sampled points is variable. Therefore, we assigned to each measured point by our nodes the explanatory variables of the nearest point on the grid, as illustrated in Figure 4.3.

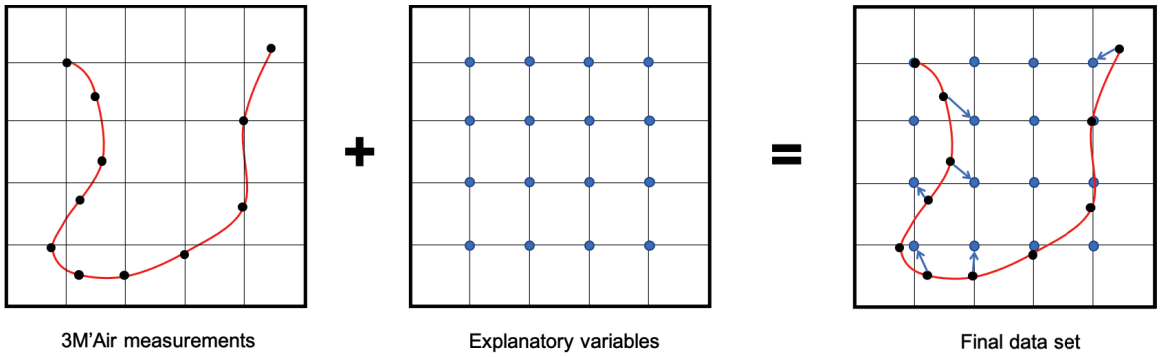


Figure 4.3: Assignment of explanatory variables to points measured by the sensors

4.2.1 Comparison of regression models

In order to get an insight on which model has the best performance with our data, we compared the performance of four models that we explained in Chapter 2, namely the multiple linear regression (MLR), k-nearest neighbors (KNN), extreme gradient boosting (XGBoost), and a multi-layer perceptron neural network (MLP). In this vein, we ran multiple iterations of estimations for each measurement campaign, using the train/test split method, with 80% of data randomly selected for training and the remaining 20% for testing. We also evaluated the impact of the parameter k on the MAE, by varying its value, which showed that the lowest Mean Absolute Error (MAE) was reached with $k = 4$.

Results in Figure 4.4 show the MAE for each model, applied to all sensing campaigns. We first observe that the estimation error is relatively lower for the campaign of June 6th, 2019, while the other three campaigns had similar behavior. Furthermore, we observe that MLR has the worst performance overall while KNN, MLP, and XGBoost give approximately the same estimation error with a slight advantage for KNN.

We also calculated the average execution time for one iteration of estimation for each model over 40 iterations, using 80% of the data for training and 20% for testing. Results in Table 4.2 show that the MLR model is the fastest with 0.035 seconds for one iteration, followed by KNN (60% slower), then XGBoost with 1.8 seconds, while the MLP is the slowest model among the four. We also observe that the overall MAE is relatively higher for the three October campaigns compared to the June campaign. This could be related to potentially smaller variations in $PM_{2.5}$ during the first campaign.

In addition, we evaluated the estimation error of the four models in function of the size of the training set. For that, we varied the fraction of the training set from 10% to 90%, and for each fraction, we randomly constructed the training set and evaluate the performance of the four models. This process was repeated 40 times for each fraction. In Figure 4.5, the MAE of each model in function of the size of the

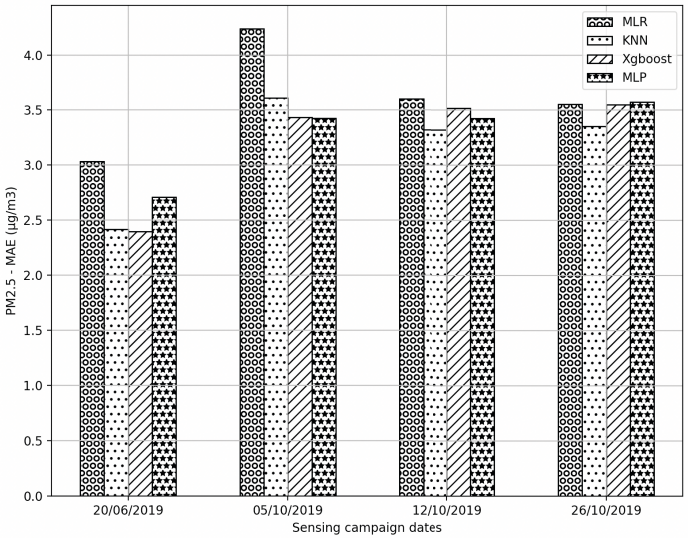


Figure 4.4: MAE of $PM_{2.5}$ concentrations estimation for each sensing campaign

Model	KNN	MLR	XGBoost	MLP
Average execution time (seconds)	0.056	0.035	1.86	7.6

Table 4.2: Average execution time of 1 iteration for KNN, MLR, XGBoost, and MLP

training set is plotted. Results indicate that the impact of the size of training set size could vary, depending on the model. For instance, MLR becomes less sensitive to the fraction of training set after 30%, while the MLP shows more variation. KNN and XGBoost have globally the same behavior with respect to the size of the training set.

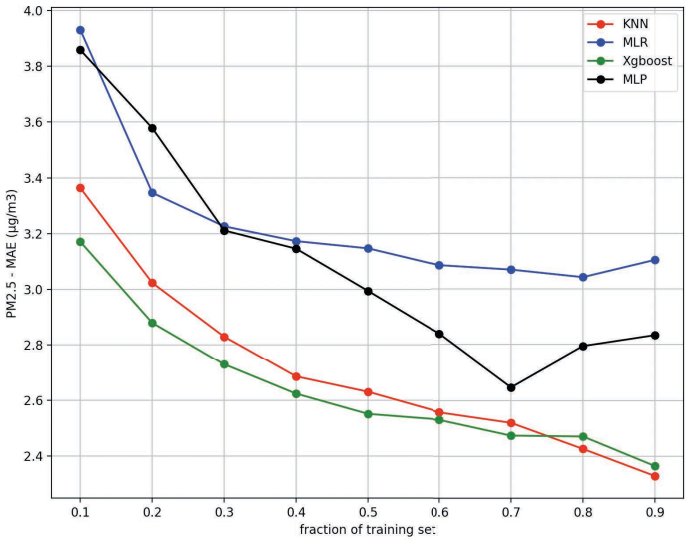
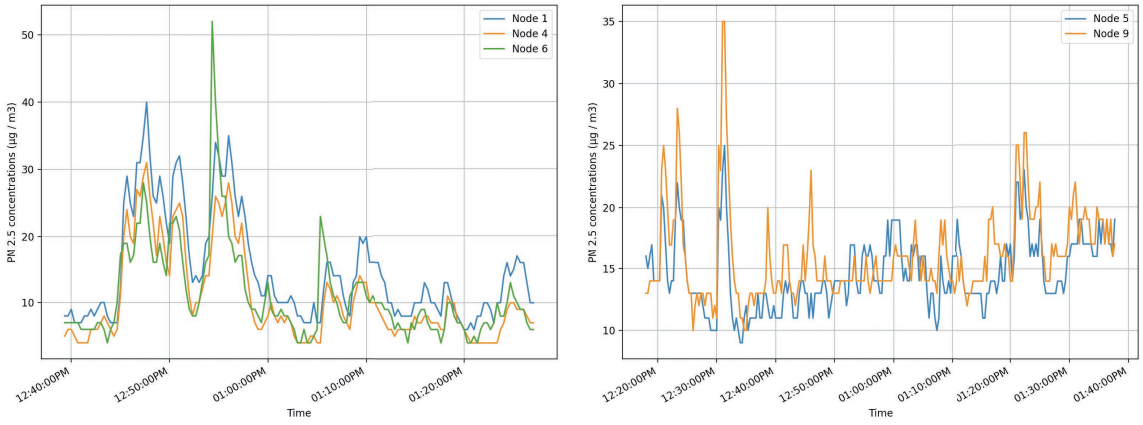


Figure 4.5: MAE of $PM_{2.5}$ concentrations estimation in function of the training set fraction

Based on these observations, we chose to use KNN for the coming tests with $k = 4$, because it performed better than the other algorithms in almost all campaigns, while having a significantly lower execution time, especially compared to XGBoost and MLP.

4.2.2 Sensor data reconstruction

Deploying low-cost WSNs often involves multiple challenges such as loss of connection, sensor failures, etc. In air quality monitoring applications, these events can have a huge impact on the performance of the application, whether it is a simple loss of communication or complete failure of the sensing probe. Thus, it is important to evaluate the capacity of reproducing a sensor's measurements based on the other available sensors of the network. We plotted in Figures 4.6 $PM_{2.5}$ concentrations for two different routes of two sensing campaigns. Figure 4.6 (a) shows measurements from sensors 1, 4, and 6 which were sampling the same route during the campaign of October 5th, 2019, while Figure 4.6 (b) depicts measurements from sensors 5 and 6 which were sensing the same route during the campaign of the 12th of October 2019. It can be seen from the plots that sensor readings follow globally the same trend, except for some differences that could be related to improper handling of the sensors or simply to the sensing probes' performance. The Pearson coefficient of correlation was calculated between these sensors, and it confirmed the visual observation as indicated in Table 4.3.



(a) (b)

Figure 4.6: Example of $PM_{2.5}$ concentrations measured by our sensing nodes during the campaign of (a) October, 5th 2019 (b) October, 12th 2019

Date	Group of nodes	Pearson's coefficient
October 5th, 2019	nodes 1 and 4	0.976
	nodes 4 and 6	0.813
	nodes 1 and 6	0.807
October 12th, 2019	nodes 5 and 9	0.783

Table 4.3: An example of Pearson's coefficient of correlation of $PM_{2.56}$ for sensors concentrations

In order to evaluate the possibility of predicting faulty sensor's data based on the remaining operational ones, we imagined a scenario in which the system receives no measurements from a node due to an operation problem. For this, performed a cross validation by taking one sensor's measurements for testing while using the other sensors to train the model. This process was repeated for each measurement campaign. The bar chart depicted by Figure 4.7 presents the MAE of predicting

$PM_{2.5}$ concentrations for each sensor during the four sensing campaigns. The error of prediction varies depending on the considered sensor and the sensing campaign, with the largest MAE (10.31 $\mu g/m^3$) reached during the campaign of October, 12th for sensor number 2. On the other hand, sensor number 10 gets the lowest error (2.18 $\mu g/m^3$) during the same campaign.

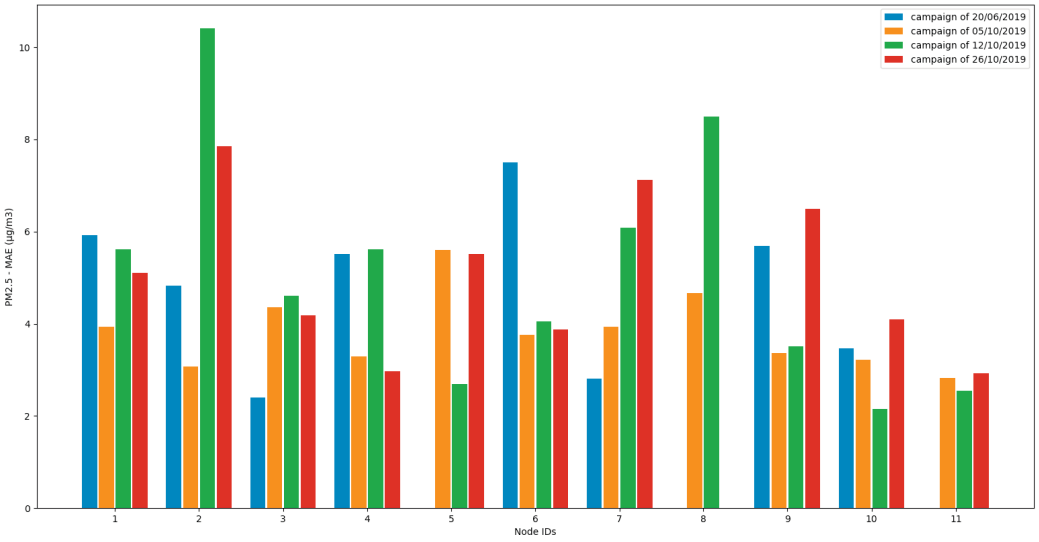


Figure 4.7: $PM_{2.5}$ concentration estimations MAE for cross-validation

These results help to evaluate the performance of the low-cost sensors as they are likely to present some divergence in sensing, despite being of the same type, hence the need for frequent calibration. Moreover, it can be noted from Figure 4.7 that sensors number 10 and 11 for example have in general the lowest MAE across the four campaigns, meaning that they are well represented by the other sensors. It should be noted, however, that these results were obtained using a simple spatial interpolation method (KNN) and explanatory variables that are not completely up-to-date. They must be compared with other methods and using more recent datasets. Nevertheless, these observations can provide some indications of the approach to adopt with a dense network of sensors in function of the predictability of each sensor, such as fixing different sampling rates for different sensors or choosing a scheduling approach in which one sensor stops measuring when it is located in the vicinity of another sensor.

4.2.3 Sensing rate VS energy consumption

As explained in the previous chapter, energy consumption is of utmost importance when dealing with low-cost WSNs. In fact, sensing nodes and especially portable ones are often equipped with small batteries to meet multiple requirements in terms of size and budget, which can result in a greatly limited operating time. To cope with that, one can maximize the idle state of the nodes in order to extend the lifetime of the sensor nodes. However, this is often achieved at the expense of the spatio-temporal resolution, which has a significant impact on the knowledge of the phenomenon.

In section 3.3.3 of the previous chapter, we have presented the evaluation of the power consumption for our sensing nodes with multiple configurations. Results revealed that with sampling and transmission rates fixed at 20 seconds and 1 minute, respectively, the average energy consumption of our sensors is 231 mA. However, by

removing the PM sensor, the power consumption drops remarkably to 115 mA because of its integrated fan. Therefore, by carefully deactivating the PM sensor, one could considerably increase the operating time of the sensor.

Based on these results, we estimated the energy consumption for different configurations of sensing rate using a simple energy model. We introduced a sensing duty cycle D that represents the fraction of time during which the PM sensing probe is active over the sensing window. For example, if the sensor is active for 30 seconds and off for 10 seconds, then $D = 3/4$. The sensing duty cycle has to take into account the convergence time of the sensor, which is the time needed for a sensor to provide reliable measurements. This value varies from a sensor to another [122]. The formula of the energy consumption is given by:

$$I_{average} = I_{PM_{ON}} * D + I_{PM_{OFF}} * (1 - D) \tag{4.1}$$

where $I_{average}$ is the average operating current of the 3M'air node, $I_{PM_{ON}}$ is the operating current of the node when the PM sensor is ON (231 mA in our case), and $I_{PM_{OFF}}$ is the operating current when the PM sensor is turned off (115 mA in our case). It is worth mentioning that we do not turn off just the fan of the PM sensor, but the whole sensor instead. In addition, we do not turn off the other sensing probes because their energy consumption is much lower compared to the PM sensor. The GPS receiver is not turned off either because it needs a longer time to get to acquire satellite signals

Furthermore, in addition to evaluating the energy consumption of our nodes with a sensing rate of one sample every 20 seconds, we estimated their energy consumption for a sampling rate of 40, and 60 seconds using (4.1) and a convergence time of 30 seconds for the PM sensor, as indicated in its data sheet [96]. For a rate of 20 seconds, the PM sensor cannot be turned off because it does not have enough time to reach a steady state ($D = 1$). In contrast, the PM sensor can be turned off for 10 seconds and 30 seconds when the sampling rate is set to 40 ($D = 3/4$) and 60 seconds ($D = 1/2$) respectively. Table 4.4 shows the estimated operating current and time of the nodes for the three configurations using a 5100 mAh battery. It can be observed that a small change in the sampling rate can have a significant impact on the node's autonomy.

Sampling rate (seconds)	20	40	60
Average operating current (mA)	231	202	173
Average operating time (hours)	22.07	25.24	29.48

Table 4.4: 3M'Air node's operating time in function of the sampling rate

The performance of estimating PM concentrations was evaluated when reducing the sampling rate. For the first test, we randomly picked 80% of our sensors' data for each sensing campaign to train the model, and three configurations were tested on the training set. In the first configuration, we kept the initial sampling rate (i.e., 20 seconds). The sampling rate was then reduced to 40 seconds (by considering one measurement every two measurements) for the second configuration. Similarly, we lowered the rate to 60 seconds for the third configuration (i.e., taking one measurement every 3 measurements). The temporal resolution for the testing set was kept at 20 seconds and the process was repeated 40 times.

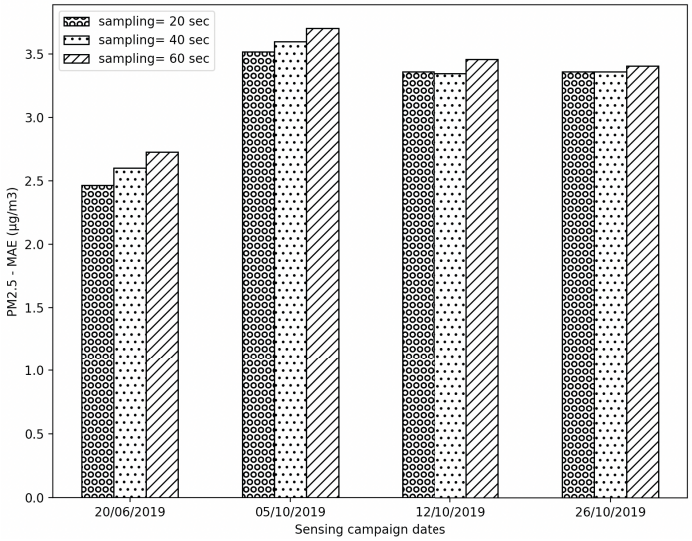


Figure 4.8: MAE of $PM_{2.5}$ concentrations estimation with three different sampling rates using 80% of data for training

Figure 4.8 shows the MAE of the estimation model for all sensing campaigns, in function of the sensing rate. We observe that even when reducing the sampling rate by a factor of two to three, the estimation model can still achieve acceptable results compared to using the initial rate. Indeed, reducing the sampling rate to 40 seconds resulted in an error 1.57% larger while achieving 14.36% longer operating time. Moreover, by lowering the rate to one sample every 60 seconds the performance of estimation decreased by around 4.64%, but the node managed to save 33.57% more energy. Therefore, the gain in power could outweigh the loss in estimation quality, depending on the addressed application. Depending on the application, this extra battery autonomy may allow sampling more locations and hence, reduce the estimation error even further.

4.3 Air quality mapping using dense sensor networks

After highlighting the potential of low-cost environmental sensors through a data analysis of the pollutant concentrations collected during our sensing campaigns. In this part, we break the budget limit and consider a dense deployment of fixed low-cost WSN for air quality monitoring, while comparing regression approaches to data assimilation and taking into consideration multiple aspects, such as high density of sensors, numerical model errors, and sensing errors. However, to achieve this scaling towards data analysis of such dense network, the use of real data is impossible. Thus, the need to develop a generic framework to generate synthetic data and enable the performance evaluation based on these synthetic data.

4.3.1 Framework overview

In this section, we present an overview of the methodology that we followed in order to compare different regression and assimilation methods (see Figure 4.9). We propose to first generate multiple ground truth datasets based on simulated concentrations from a physicochemical model and an adequate estimation of the covariance matrix of the simulation errors. We explain in the next section the approach that we use to

estimate the covariance matrix of simulation errors.

Once the ground truth datasets are generated, we generate for each one a large number of observations (pollutant concentrations) in positions where low-cost sensors are deployed. This generation takes into account the ground truth realizations as well as the measurements' errors variance-covariance matrix. Based on the measurement realizations, the simulation values, and the ground truth realizations, multiple regression and assimilation methods can be implemented and compared.

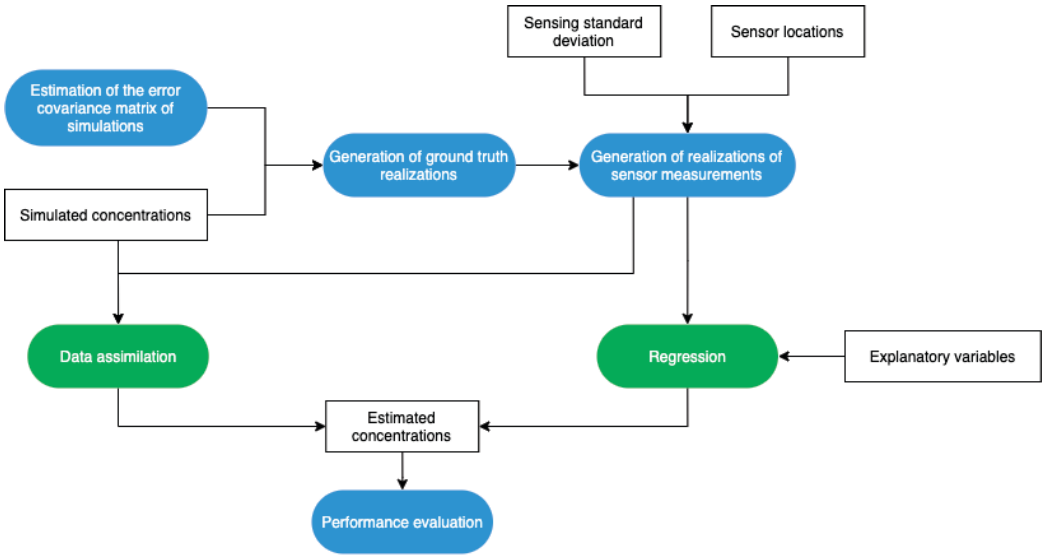
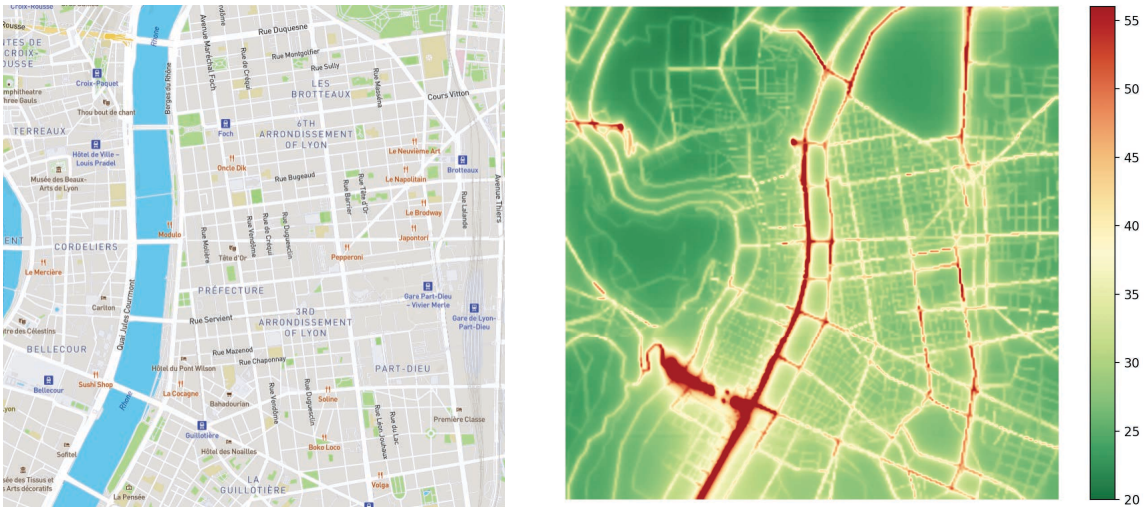


Figure 4.9: Overview of the methodological flowchart used for the framework

Regarding the datasets, we consider the simulated NO_2 concentrations in Lyon and the explanatory variables presented in Sections 4.1.4 and 4.1.3, respectively, while focusing on a $2.5 \times 2.5 km^2$ zone which corresponds to the center of Lyon and its immediate vicinity (see Figure 4.10).



(a) (b)

Figure 4.10: The zone of interest (a) map of the center of Lyon and its immediate vicinity (b) annual averages of NO_2 concentrations in 2008 (simulated data)

4.3.2 Characterization of the variance of simulation errors

In order to characterize the variance of the simulation errors, we used NO_2 concentration values provided by 16 reference stations in Lyon and compared them to the simulated concentrations of SIRANE. We considered monthly values in both cases. For each simulated value z, we computed the standard deviation of the errors associated to the simulated values in $z \pm 5\mu g/m^3$. Figure 4.11 shows that the model's error standard deviation σ_b, depends linearly on the pollutant concentration starting from a given threshold z_0:

$$\sigma_b(z) = \alpha(z - z_0) + \beta, z \in [z_0, +\infty[\tag{4.2}$$

Where z is the simulated value. As shown in Figure 4.11, a linear regression with $z_0 = 24\ \mu g/m^3$ results in $\alpha = 0.344$ and $\beta = 2.27\ \mu g/m^3$ with a high R^2 value ($R^2 = 0.94$).

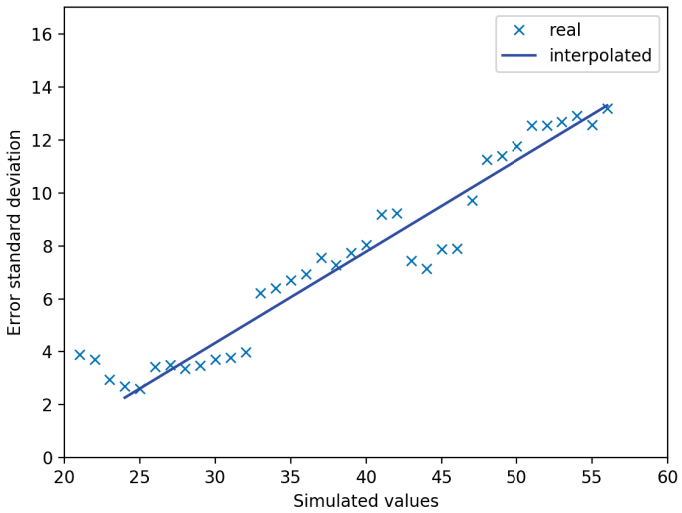


Figure 4.11: Error standard deviation vs Simulation Values

4.3.3 Ground truth and measurements generation

Based on the variances of simulation errors and correlation coefficients $\mathbf{w}_{ll'}$, we generate the variance-covariance matrix. We consider here the correlation coefficient as a function of the distance [123] given as: $\mathbf{w}_{ll'} = e^{-\delta \mathbf{d}_{ll'}}$, where δ is the attenuation coefficient of the correlation function, and $\mathbf{d}_{ll'}$ is the euclidean distance between locations l and l'. Assuming that the simulation error follows a multivariate normal distribution, we generate multiple ground truth datasets based on the simulated values and the computed variance-covariance matrix.

Once the ground truth datasets generated, we generate, for each ground truth set, a large number of observations (measurements) in positions where low-cost sensors are deployed. Sensor measurements are generated using a normal distribution, since the sensing errors are not correlated with each other. Hence, the measurement error variance-covariance matrix that we note $\mathbf{R}$ is given by: $\mathbf{R} = \mathbf{v}_0 I$, where $\mathbf{v}_0$ is the vector of measurements' error variance and I is the identity matrix. For each sampled location l, the sensor reading is generated using a normal distribution of mean y_l and variance σ_m^2, with y_l the ground truth value, and σ_m the standard deviation of the sensing error.

4.3.4 Evaluation of regression approaches

In this section, we compare different mapping strategies using four regression approaches, namely multiple linear regression, k-nearest neighbors, random forest and XGBoost, while taking into consideration the intrinsic characteristics of dense deployment of low-cost WSN for air quality monitoring. We compared these approaches on 30 realizations of ground truth obtained from a simulation with $\alpha = 0.05$.

Choice of the parameter k for the KNN algorithm

To choose a good value of k, we evaluated the MAE for different values of k. The results are presented in table 4.5. We noticed that the parameter k had no significant impact on the MAE for $k \in [3, 15]$. However, the minimum MAE was reached when $k = 5$. k was then set to 5 for the following comparisons.

k	**3**	**5**	**7**	**9**	**11**	**13**	**15**
MAE	3.246	3.230	3.234	3.240	3.245	3.251	3.256

Table 4.5: MAE vs k value of KNN regression ($\sigma_m = 2\ \mu g/m^3$, Fraction of deployed sensors = 0.3)

Impact of the number of deployed sensors

The aim of this study is to evaluate the performance of regression methods in function of the percentage of deployed sensors, i.e., the fraction of sampled points. For that, we evaluate the estimation MAE of these methods when increasing the fraction of deployed sensors. For each ground truth dataset, all approaches are executed ten times with different measurement realizations (with $\sigma_m = 1 \mu g/m^3$). The results, depicted in Figure 4.12, show that all methods present a better performance when the number of sensors increases. Indeed, the more deployed sensors, the lowest the error and the better the estimate. Random forest outperforms all the methods, reaching an MAE of $1.48 \mu g/m^3$ when 40% of the points are sampled. It is followed closely by XGBoost, then KNN, and finally MLR.

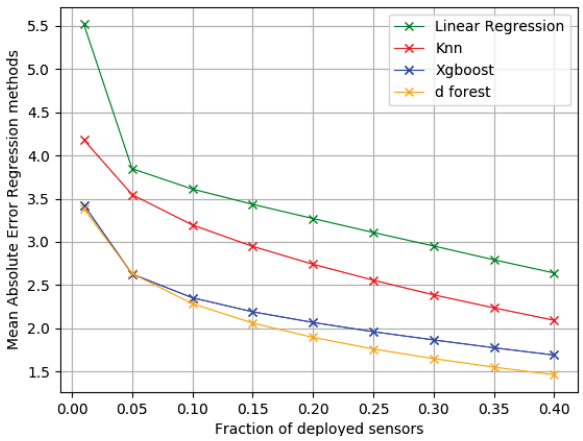
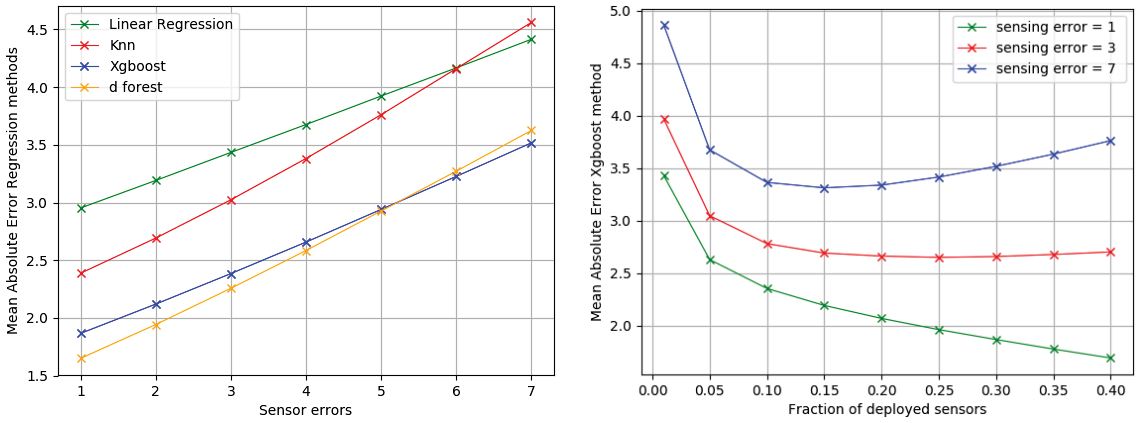


Figure 4.12: MAE vs percentage of sampled points, with σ_m= 1 $\mu g/m^3$

Impact of sensor errors on regression methods

The main characteristic of low-cost environmental sensors, besides their cost and size, is their low-accuracy. Therefore, we assessed the impact of the sensing error by varying σ_m from 1 to 7 $\mu g/m^3$, while fixing the number of deployed sensors to 30% of the total number of possible points. Results, plotted in Figure 4.13a, show that the MAE increases for all four methods as the sensing error increases, with XGBoost and random forest being close and outperforming KNN and MLR. However, we can observe different sensitivity to the σ_m for each method. Indeed, from the different curve plots, we notice that XGBoost and the multiple linear regression are less sensitive to variations in sensing errors than random forest and KNN. Interestingly, although KNN is relatively close to XGBoost and random forest when the sensing error is small, it quickly deviates from the two as σ_m starts increasing. Another interesting observation is that starting from a given error threshold, XGBoost and MLR can outperform random forest and KNN, respectively. These results give an idea regarding the choice of which estimation method to use in function of the sensing error.



(a) (b)

Figure 4.13: (a) MAE vs standard error deviation, Fraction of deployed sensors = 0.3; (b) MAE of XGBoost method vs percentage of deployed sensors, $\sigma_m \in [1, 3, 7]$

Moreover, we analyzed the behavior of regression approaches regarding the sensing error and the density of the network. In this vein, we computed the MAE of XGBoost while considering three different sensing standard deviation errors (σ_m = 1, 3 and 7 $\mu g/m^3$) and varying the percentage of covered points from 1% to 40%. Results reveal in Figure 4.13b that when the sensing error is relatively high ($\sigma_m \geq 3\mu g/m^3$), increasing the number of deployed sensors does not necessarily improve the results, but instead decreases the performance of the mapping when the sensor density exceeds a certain value. In this case, when the sensing error is large, it is not useful to deploy more sensors. The results of the four regression approaches were similar with different sensing errors and sensor density.

4.3.5 Evaluation of data assimilation

In this part, we evaluate the performance of the data assimilation approach, namely the best linear unbiased estimation (BLUE), presented in Section 2.7 of Chapter 2, in function of the size of the network (number of deployed sensors), the simulation

error, and the sensing error. Similar to regression methods, the results were obtained over 30 realizations of the ground truth.

Impact of simulation errors

As seen in Chapter 2, data assimilation uses the output of numerical models as a base, then considers measurements collected in the field in order to correct the model's estimation. Indeed, these models are not perfect and present some errors due to the complexity of the studied phenomenon and the inputs that are usually averaged in space and time. With this respect, we computed the MAE of data assimilation for two models with α equals to 0.05 and 0.4, a sensing error $\sigma_m = 3$, and a percentage of sampled points ranging from 1% to 40%. We recall that the higher α is, the higher the error of the simulation model. Results depicted in Figure 4.14 show, as expected, that an assimilation with an accurate simulation output (i.e., small α) performs better than an assimilation applied to a less accurate physical model. Nonetheless, the two systems benefit from the increasing number of deployed sensors to further correct the model. However, the decrease in MAE is more important for the network with $\alpha = 0.4$.

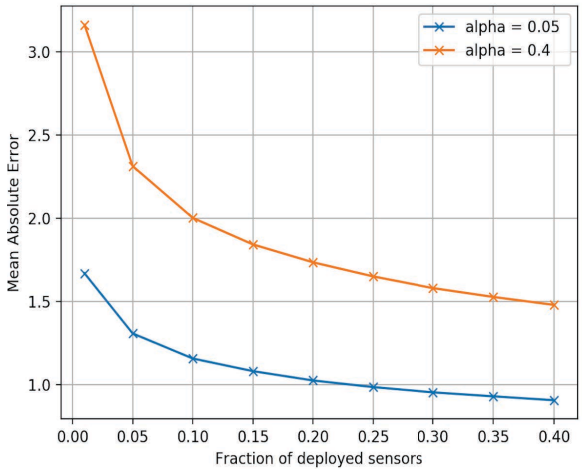


Figure 4.14: MAE of data assimilation in function of the sensing error and the percentage of covered points $\sigma_m = 3\ \mu g/m^3$

Impact of sensing errors

Since data assimilation relies on sensor readings to correct the numerical simulations, the sensing errors certainly affect the final output. In order to evaluate this effect, we performed the BLUE assimilation with a fixed α (0.05) and a variable number of sensors, on four networks with a sensing error equals to 1, 3, 5, and 7 $\mu g/m^3$, respectively. As illustrated in Figure 4.15, the MAE decreases as we increase the fraction of deployed sensors. Interestingly, in contrast to regression methods, increasing the number of sensors deployed always helps to correct the model. Indeed, we can observe from the figure that even with a σ_m=7, the data assimilation still manages to reduce the error of the model, thanks to the consideration of both the simulation errors and the sensing errors.

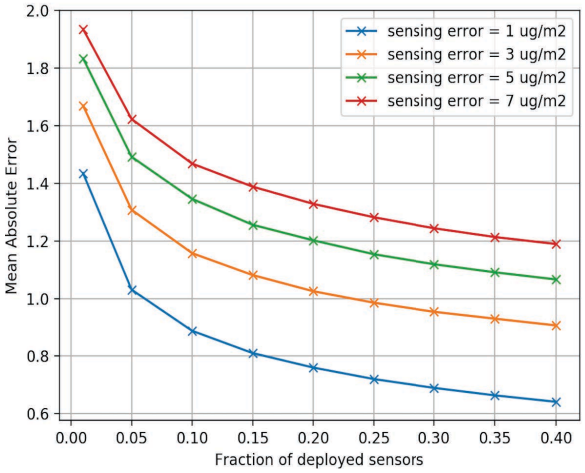


Figure 4.15: MAE of data assimilation in function of the sensing error and the percentage of deployed sensors, α=0.05

4.3.6 Regression vs data assimilation

In this last simulation, we compare the four regression methods to the BLUE assimilation with different simulation errors. For that, we plot in Figure 4.16 the MAE of all methods while varying the value of α from 0.05 to 0.45. The results show that BLUE provides 40% better air quality estimation compared to random forest, which already have the best performance among the regression methods. This is mainly due to the consideration of the model's error as well as the sensing errors in the assimilation process. It is worth mentioning that BLUE performs very well when the simulation error covariance matrix and sensing error covariance matrix are well estimated.

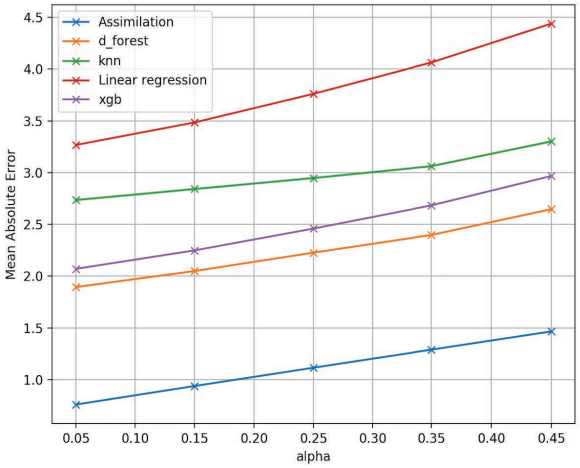
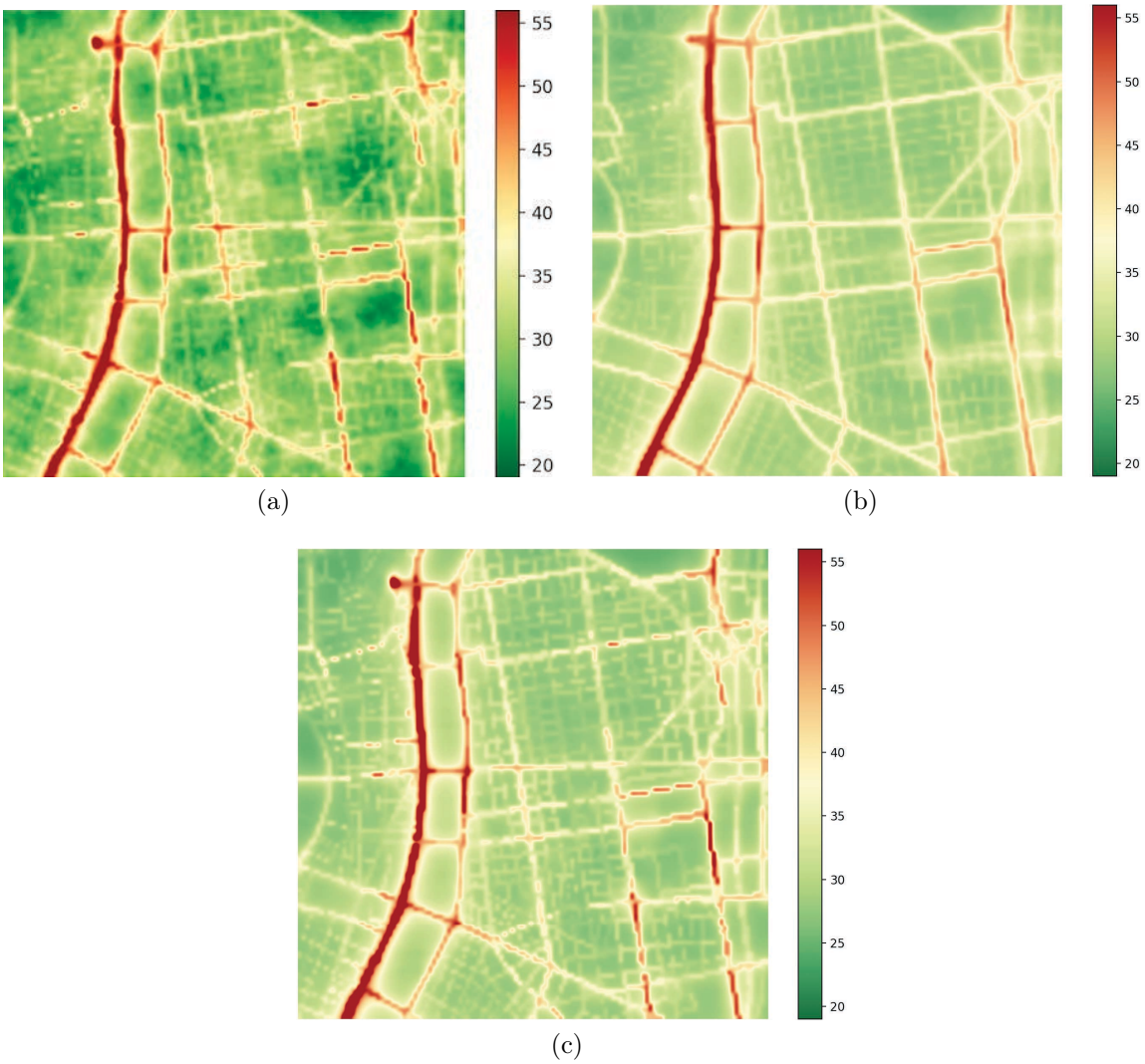


Figure 4.16: MAE of all regression and data assimilation methods in function of α, with $\sigma_m = 1\ \mu g/m^3$ and 30% of points sampled

The advantage of data assimilation is illustrated by Figure 4.17, which compares a reference heatmap of NO_2 concentrations (left) to two heatmaps estimated using random forests (right) and data assimilation (bottom). The estimated heatmaps were generated using a percentage of coverage of 20%, $\alpha = 0.05$, and $\sigma_m = 3\mu g/m^3$. From the figure, we can observe that data assimilation captures the variations of the pollutant concentrations with higher resolution than random forest, which tend to smooth the heatmap. Moreover, random forest visibly misses some hotspots compared

to BLUE.



(a) (b)

(c)

Figure 4.17: NO_2 concentrations heatmap for the area of interest, (a) a realization of the ground truth (b) random forest estimation (b) BLUE estimation

4.4 Conclusion

Through this chapter, we first provided spatial analysis of data collected during four sensing campaigns, using our lab-designed sensor nodes. This analysis allowed us to explore questions related to participatory sensing using low-cost WSN, such as the ability of the network to reconstruct missing data of a faulty sensor, or the gain/loss in energy and sensing quality in function of the sensing rate. We compared four statistical models and highlighted their ability to achieve acceptable performance despite lowering the sampling rate of the sensors. Moreover, we investigated in the last part, the use of dense low-cost WSN in air quality assessment. We proposed a general framework that allows the comparison of different regression and data assimilation approaches based on numerical simulations, while considering the size of the network, the simulation errors, and the sensing errors. We observed that data assimilation is less sensitive to the variations of measurement errors. In addition, we pointed out that

a large number of sensors with high errors is not always good for regression methods. However, this is not the case for data assimilation, especially when covariance matrices of the model's errors and the sensing errors are well estimated.

In the next chapter, we tackle another issue related to participatory sensing, which is the selection of participant routes. We propose two algorithms that take into consideration the accuracy of the sensors and the relationship between participants' routes, in addition to the distance constraint.

Chapter 5

Route selection in participatory mobile sensing

Given the mobile nature of the crowd, mobile crowdsensing platforms need to implement adequate route planning/selection solutions to better guide the crowd through the area of interest and maximize the quality of monitoring.

Route planning and route selection are of great importance in mobile crowdsensing applications. On one side, participants in route planning delegate the construction of their route to the monitoring platform, which will drive participant movements by leading them through specific points of interest. On the other side, in route selection, each participant can have multiple favorite routes (generated using route planner algorithms), and the role of the system is to select the most appropriate path regarding the needs of the task. It is clear that the second method considerably limits the degree of freedom of sensing platforms as they are restricted to choose from already built candidate routes that may inevitably overlap, which results in the same point/area being sampled several times, and thus in data redundancy. Data redundancy can sometimes be beneficial when it comes to low-cost sensors, which by definition present low-accuracy properties. Some monitoring applications can be more tolerant regarding redundant data, in order to mitigate the impact of wrong sampling caused by faulty sensors.

In this chapter, we address the route selection problem in a participatory sensing context, while focusing on the application of air quality monitoring to validate our proposal. We first provide an overview of some related research work discussing route selection in crowdsensing applications. We then introduce the scenario we consider in this study. Afterwards, we present two route selection algorithms that take into consideration the low accuracy of the sensors. The first algorithm computes the similarity between the different possible routes in order to find a combination of participant routes that maximizes spatial coverage. The second algorithm takes advantage of clustering to evaluate the resemblance between the points of the area of interest in order to cover as many distinct points as possible. Finally, we evaluate the performance of the proposed algorithms and compare them to other baseline route selection algorithms.

5.1 Route selection : a paradigm of great interest in mobile crowdsensing

Mobile crowdsensing have raised questions regarding the movement of the crowd and how to maximize coverage while taking into consideration the constraints of the participants and the addressed application. This has opened the door for new research studies that harness citizen's mobility to improve the mapping of environmental phenomena.

Synthetic measurements were used in [124] to predict a map of NO_2 concentrations between the 12th and 25th of February 2018 in the city of Marseille, France, while considering up to 4500 bike-tracks randomly generated. The simulated concentrations were generated using a numerical model with a spatial resolution of $25 \times 25 km^2$, while the explanatory variables (including number of trees, buildings, configuration of the road network, etc.) of the area were collected from "Open Street Map". Fictive bike tracks were constructed using a cyclist route planner API, with a distance following a normal distribution of mean $2.5km$ and standard deviation $4.5km$. The resulting dataset served to train and compare the performance of three estimation models, namely ordinary kriging, multilayer perceptron neural network, and a generalized additive model (GAM), which is the generalization of the linear regression to non-linear systems. The evaluation of the models used the simulated map generate by the model (without the measurements along the selected bike tracks) as a testing set. The performance assessment mainly evolved around the impact of the number of tracks and the spatial resolution of the measurements on the prediction error of the three models. The observed results showed that while the Mean Absolute Error (MAE) of kriging continued to decrease as the number of tracks increases, the RMSE for MLP and GAM showed almost no improvement with more than 100 tracks. The same observation was made for the coefficient of correlation.

In multiple crowdsensing applications, the system owner pays the participants for their collected data. In these use cases, the participants often perform sensing tasks at specific locations rather than continuous sampling. The payment is generally related to the number of accomplished tasks, their priority, the travel distance/duration, or the quality of the collected data. In this context, given a budget, the system owner optimizes the mobility of the crowd in order to maximize the number of completed tasks or the coverage of the study area, while satisfying the constraints of the participants.

A three-phase routing algorithm is proposed in [125] in a task assignment context. A task is characterized by its location and its added value to the system. In this work, the researchers do not rely on a third-party routing API, but instead, they construct the participant routes from scratch. The proposed algorithm is composed of three phases, where in the first phase the algorithm iteratively constructs the route from the origin to the destination, by considering at each step the task that has the largest added value. The task is assigned to the user if his device energy is enough to travel from the current location to the target task, then to the destination point. The second phase of the algorithm is similar to the first one, except that it starts the construction backwards (i.e., from the destination point to the starting point of the participant). The last part of the solution is the selection of the route with the highest added value. This solution is appropriate if we consider specific sensing locations and a limited budget constraint.

A crowd-based urban sensing framework is presented in [126] with the objective of maximizing area coverage for noise sensing. It uses a graph-based task assignment

algorithm and an objective function to maximize data coverage with a limited budget constraint. An urban sensing task is defined as a sequence of collecting locations with corresponding time intervals. The assignment of a task is done by estimating the travel time needed for the participant traveling from its origin to the task location and the travel time from this location to the destination of the participant. The authors also propose a recruitment mechanism that randomly selects participants from a candidate pool and then replaces them gradually by other candidates left in the pool, until no improvement of coverage is possible anymore.

Instead of suggesting a whole route to a participant, the work in [127] takes a different approach to maximize the coverage quality while avoiding redundant data. The study proposes a reverse greedy algorithm that selects only a subset of segments from the participant's route, based on the cost of the task and the available budget. The selected segments are those along which the participant performs the task. Thus, the participant will be rewarded according to the selected segments instead of his whole route. To achieve such result, the algorithm eliminates redundant segments. Two segments are considered redundant if the distance separating their respective endpoints is less than a predefined distance. After each round of trimming, if the total cost is greater than the available budget, a new round of trimming starts with a larger threshold distance to further reduce the number of selected segments. Two coverage metrics were used to evaluate the solution, namely the coverage percentage which indicates the ratio of covered to non-covered grids, and the uniform degree which describes the spatial distribution of selected segments.

TaskMe [128] identifies two bi-objective optimization problems. The first aims at maximizing the number of accomplished tasks and minimizing the total traveling distance in a situation where there are few participants and many tasks. To solve this problem, the authors present two algorithms based on Minimum Cost Maximum Flow models, where the first one selects all tasks as candidate whereas the other one takes into account only the nearest tasks to a given user. In contrast, the second optimization problem is designed for the case where there are more participants and only few tasks, and its objective is to find the set of participants to complete all tasks while minimizing the total incentive payments and the total traveled distance, knowing that the incentive is inversely proportional to the traveled distance. Since the two objectives are contradictory, the authors propose to convert the multi-objective problem to a single-objective one by either introducing weights to each objective or considering one of them as a constraint in the optimization model.

Gong et al. [129] addressed the path planning problem to maximize the total task quality in a scenario where users and tasks arrive dynamically. Each user has a limited distance budget and has to register his starting and destination points upon arrival. One of the proposed algorithms in this work selects the tasks that lead to the largest gain-cost ratio, one by one in a greedy manner, as long as they satisfy the travel distance budget of the user. An alternative solution tends to guide users to task-dense areas to maximize the cost-gain ratio. In addition to that, the authors designed an algorithm that takes into account the impact of a candidate task on the travel distance budget, in order to leverage tasks with low impact on the available budget. Given a candidate task, this algorithm evaluates the possibilities of the next step by computing the distance from this task to the others. The task that does not distract the participant from the rest of the tasks is selected. This algorithm yielded better performance compared to the other two solutions.

In [130], the goal is to maximize the sub-profit in each time slot to approximately approach the maximum profit of all slots for a given task. A task is represented as

a number of sensing locations during a certain period, the area of interest is divided into N cells, and the task duration into M time slots with the same length. Each participant in this scenario is guided to move along the shortest path. However, to avoid similar routes and maintain a stable dispersed distribution, the authors use 2D entropy (E) to guide participant distribution. The more decentralized the participants, the higher the entropy value. The solution first randomly selects a number of cells and compute E, then repeats the selection process for multiple. At last, the participant distribution with the largest E is selected as an approximately optimal solution.

Although the aforementioned studies explore mobile crowdsensing capabilities to accomplish sensing tasks, the majority of them focus more on specific sensing locations rather than sampling the entire zone. This prevents performing continuous sampling, which considerably limits the potential of the crowdsensing and significantly reduces the spatial resolution, especially in air quality monitoring applications where there is no sensing range. Moreover, the majority of these work do not take into consideration the inaccuracy of the sensors, which is one of the main challenges when it comes to monitoring air pollution using low-cost environmental sensors. Furthermore, in a participatory sensing context, participants may use their own sensors, which adds a new constraint that is sensor heterogeneity. Therefore, taking into account sensing errors during the participant recruitment phase and the route selection process should be highly considered.

5.2 Problem statement

In this section, we describe the scenario we focus on as well as the global objective we aim at. Afterwards, we give a mathematical formulation of the problem to solve it.

5.2.1 Scenario Description

In this part, we focus on a scenario in which multiple participants are equipped with heterogeneous low-cost environmental sensors to measure a specific phenomenon in a delimited area. Each participant has a starting point and wants to get a path to reach his destination using a routing service. The participant is also willing to take a path that contributes to the knowledge of the studied phenomenon, without deviating too much from the optimal path.

5.2.2 Objective

Our global mission, in this context, is to suggest to each participant a route that might not necessarily be the optimal/shortest path, but does improve the estimation while being acceptable in terms of journey distance/duration. This implies taking into consideration not only the length of the participant's routes and their relationships with other participant routes, but also the accuracy level of his sensor. In other words, our goal is to find a best combination of routes that allows the system to reduce the estimation error while still satisfying participants' constraints in terms of trip distance/duration.

5.2.3 Mathematical notation

Let $U = [u_1, u_2, ..., u_n]$ be the set of participants, $S = [s_1, s_2, ..., s_n]$ the set of their respective sensors, σ_k the standard error of the *k-th* sensor, and $P_k = [p_{k1}, p_{k2}, ..., p_{kq}]$ the set of possible paths for the *k-th* participant. The goal is to select for each participant u_k a path p_{kj} from his possible q routes, knowing that he is equipped with a sensor s_k which has a standard error σ_k.

A brute force solution would be to test all the possible combinations (i.e., q^n combinations). However, as the space of solutions grows exponentially with the number of participants, implementation of such solution in a real life scenario is impractical. To cope with that, we rely on heuristics that are not optimal, but have smaller solution spaces, hence running much faster than the exhaustive search.

5.3 Traditional route selection algorithms

Through this contribution, we address route selection algorithms in the context of participatory air quality sensing. The objective is to maximize the quality of the prediction where there are no measurements, using a spatial interpolation method and an efficient route selection algorithm.

Generally, route planning APIs always offer either the shortest path or q possible paths, while taking into consideration multiple parameters (e.g., traffic condition, waypoints, type of path, etc.). Route selection consists of choosing for each participant a path among the q proposed ones. This is a key element as it determines the geographical zones that will be sampled, which highly impacts the estimation quality. We consider in this part three traditional routing approaches as baseline:

- ***Shortest-path-based routing (SPR):*** This algorithm prioritizes participant comfort by reducing the travel distance for all participants, through suggesting the shortest route among the q paths. In SPR, all route selections are performed independently for each participant.

- ***Longest-path-based routing (LPR) :*** In contrast to SPR, this approach aims at maximizing the set of points to visit. It suggests to all participants the longest route possible among the q suggested routes. It is to be noted that in this contribution, we consider the longest path within a predefined stretch factor with respect to the shortest path. Suggesting the longest-path for all participants will intuitively expand the set of collected measures, and hence, improve the estimation quality. However, the correlation between paths and especially the information provided by different points, which is very complex to evaluate, means that this method does not always maximize the quality of the spatial prediction.

- ***Random routing (RR):*** For each participant, randomly select a route among the q possible ones without taking into consideration its length, the already selected routes for previous participants, or the accuracy level of the sensors.

All three algorithms cited above are easy to implement, but their downside is that they do not take into consideration the correlations between participant routes or the accuracy levels of the sensors. In fact, the routes can pass through similar points or even different points that provide redundant information (points with similar characteristics). These observations feed the need for more sophisticated techniques,

that will efficiently attribute routes to the participants that are not necessarily the longest or the shortest ones, but which bring as much diversity in the dataset as possible. This can be achieved by introducing metrics and techniques that help to choose the most distinct routes possible and thus bring more information.

5.4 Similarity-based route selection

The level of accuracy differs from a sensor to another, even between sensors of the same type. Therefore, to offer a good estimation, platforms should have an efficient routing approach by also taking into consideration the heterogeneity of sensors quality. In this regard, our algorithm processes first the participants with the most accurate sensors. Hence, instead of independently suggesting routes for the participants, the path provided to those with lower quality sensors are adapted according to the choices made for participants with more accurate sensors. In the example depicted in Figure 5.1, participant A has a better sensor than participant B. As a result, the path of A is chosen first before the path of B. In addition, the route of participant A overlaps with one of the two possible paths for B (along the bold black segment). Our similarity-based routing therefore selects the non-overlapping route $B2$ in order to maximize the coverage of the map.

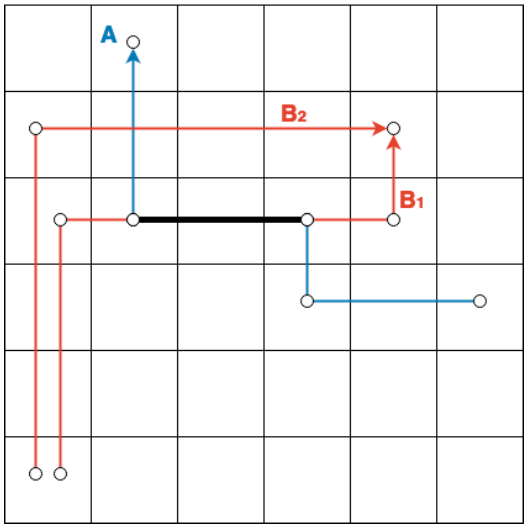


Figure 5.1: An example of overlapping segments

Given a pool of participants sorted according to their sensors' accuracy. The similarity-based algorithm considers the q possible routes of the first participant (q routes satisfying a given distance threshold between the shortest and the longest path) and selects the longest one. After that, the algorithm iterates over the next participants in the pool and chooses for each one the route that has the lowest similarity with the already selected routes from the previous participants. Every time a route is selected for a participant, the latter is removed from the pool and the algorithm moves to the next participant in a greedy manner, until the pool is empty. In Figure 5.1, the route B_1 would be privileged over B_2 for user B as it has a lower degree of similarity. This process is illustrated by Algorithm 1.

The calculation of the similarity percentage between routes has a major role in finding the most distinct routes possible, and can be computed in multiple ways. We have mainly explored two commonly used metrics in image segmentation, namely the

Jaccard index (also known as Intersection over Union or IoU) [131] and Sørensen–Dice coefficient (also known as Dice Similarity Coefficient or DSC) [132]. Considering two participant routes A and B, the formulas for these two coefficients are given as follows:

$$IoU_{A,B} = \frac{|A \cap B|}{|A \cup B|} \tag{5.1}$$

$$DSC_{A,B} = \frac{2 \times |A \cap B|}{|A| + |B|} \tag{5.2}$$

Where $|A|$ and $|B|$ are the areas of route A and route B, respectively, $|A \cap B|$ the area of the intersection between the two routes (i.e., area of overlapping), and $|A \cup B|$ the area of the union between route A and route B (with no duplicates). It is obvious that these two scores are positively correlated. Thus, we only consider the DSC in this study.

Algorithm 1: Similarity-based route selection

Input: U: The set of users

Output: P: the set of selected paths

Initialization: $P \leftarrow \emptyset$ `// initialize the set of slected paths`

`// Order the set of users based on the accuracy of their sensors`
$U \leftarrow order(U, descending\_accuracy)$

`// take the longest path for the first user in U`
$P \leftarrow longest\_path(u_0)$

for $u \in U$ *with* $u \neq u_0$ **do**

`// select the path with the least similarity`

$$s\_path \leftarrow \underset{p \in Paths(u)}{\arg\min}\ Similarity(p, P)$$

$$P \leftarrow P \cup s\_path$$

return P

5.5 Cluster-based route selection

Instead of computing the similarity between participant routes, this approach focuses on the similarity between the points of the map. The main idea is to regroup the points of the map not based on the spatial distance separating them but on explanatory variables related to surrounding conditions (such as distance to routes, meteorology, etc.). To achieve such a goal, we have opted for the agglomerative hierarchical clustering, which is a widely used technique of hierarchical clustering [133], in order to create groups of similar points that might be far from each other, but present similar properties. First, each point of the map is assigned to an individual cluster, and we calculate the distance between the clusters based on the independent variables. Then, clusters are merged successively while minimizing the sum of squared differences between the clusters being merged. As a result, all points of the map are classified into c clusters. Then, for each route, we calculate the number of

clusters it traverses. The main idea of this route selection approach is to choose for each participant the route that passes through the largest number of clusters (see Algorithm 2).

In the example depicted in Figure 5.2, a participant has two possible routes to reach his destination. Each route covers a different area of the map and passes through different clusters. In this case, the algorithm will choose route B as it goes through more clusters, despite being a bit longer than route A. In addition, the selection of a participant's route should take into account the clusters that have been already visited by other participants. This will help the algorithm to have a larger area coverage and therefore more data to train on for the estimation models.

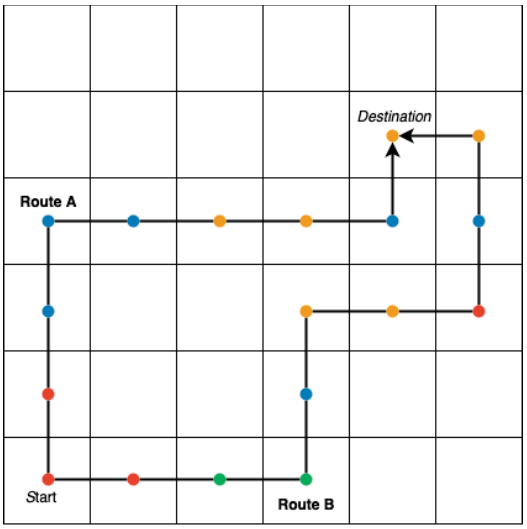


Figure 5.2: An example of two possible paths activating different clusters

Algorithm 2: Cluster-based route selection

Input: U: The set of users

Output: P: the set of selected paths

Initialization: $P \leftarrow \emptyset$; `// initialize the set of selected paths`

$C \leftarrow \emptyset$ `// the set of visited clusters`

`// Order the set of users based on the accuracy of their sensors`
$U \leftarrow order(U, descending\_accuracy)$

for $u \in U$ **do**

`// select the path visiting more new clusters`

$$s\_path \leftarrow \underset{p \in Paths(u)}{\operatorname{argmax}} NbClusters(p, C)$$

$$C \leftarrow C \cup Clusters(s\_path)$$

$$P \leftarrow P \cup s\_path$$

return P

5.6 Validation

5.6.1 Methodology

In order to validate our proposal, we followed the methodology presented in Figure 5.3, by considering simulated NO_2 concentrations as a reference map. We first start with a pool of participants, each with a starting and destination points. Each participant has a NO_2 sensor with its own standard error σ_k. Instead of constructing the different routes for each user, we rely on a routing service that provides us with several alternative paths whose length is within a predefined stretch factor of the shortest path. After that comes the route selection phase, during which the algorithm suggests a path for each participant based on the reliability of the sensors and the relationship between the routes. Following that, synthetic measurements are generated following the same approach explained in Chapter 4). For each location l visited by a participant's route, a synthetic measurement is generated using a normal distribution of mean y_l and variance σ_k^2, with y_l being the reference concentration at the location l. Finally, these synthetic observations are passed to three spatial interpolation models presented in Chapter 2, namely multiple linear regression (MLR), k-nearest neighbors (KNN), and extreme gradient boosting (XGBoost), in order to produce a predicted map. The latter is then compared against the reference map to evaluate the impact and performance of the route selection approaches.

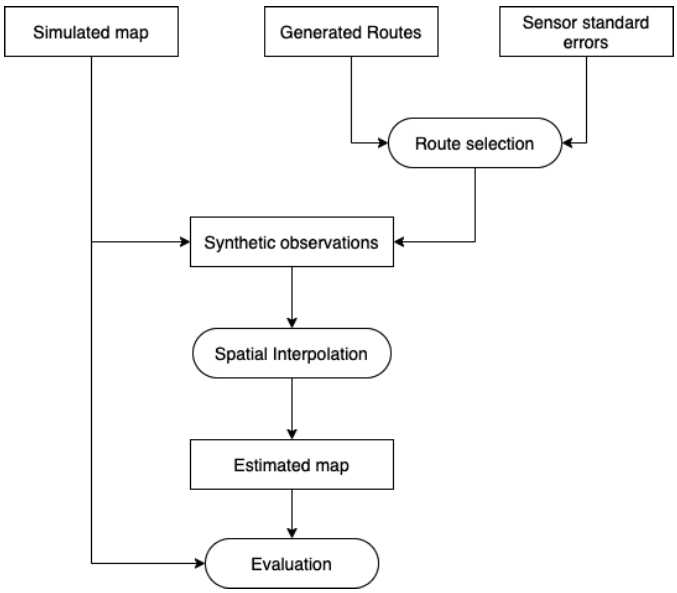
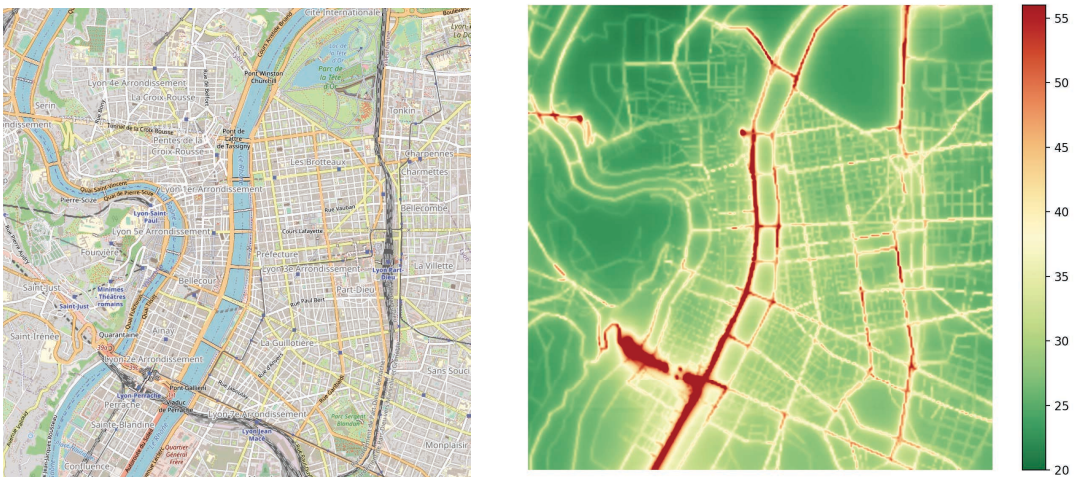


Figure 5.3: The general methodology used in validation

5.6.2 Study area and reference map

We consider in this study an area of interest of $25km^2$, which corresponds to the center of the city of Lyon and its immediate vicinity (see Figure 5.4a).

In order to build a reference map of NO_2 concentrations, we consider the two datasets presented in the previous chapter, i.e., the simulated NO_2 concentrations map of Lyon city (see Figure 5.4b) with a spatial resolution of $20m \times 20m$, and the set of 40 explanatory variables for each point on the map.



(a) (b)

Figure 5.4: (a) Area of interest (b) reference heatmap of NO_2 concentrations (simulated data)

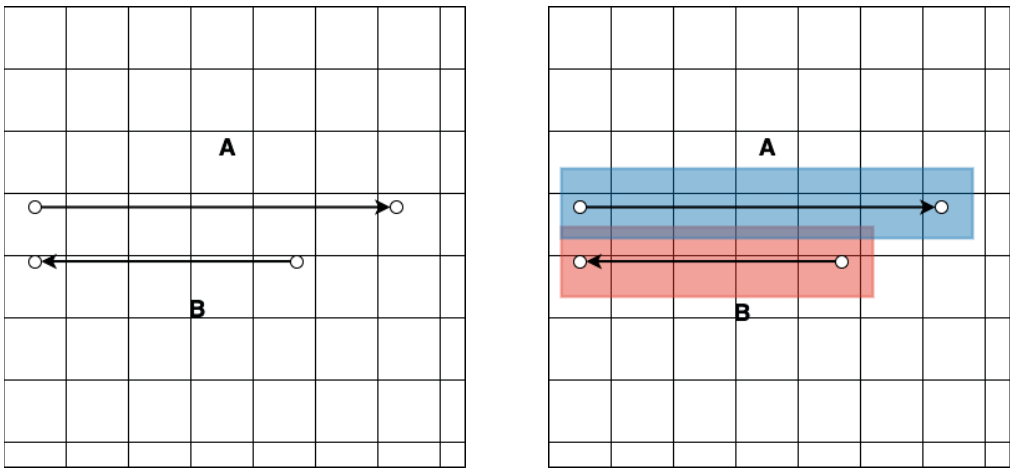
5.6.3 Participant routes and sensor measurements generation

In our use case, a participant is represented as a triple consisting of his starting point, his destination, and the accuracy level of the sensor he carries. First, we generate 200 random participants with a distance between the starting and destination ranging from $1km$ to $5km$. To match a realistic scenario in which participants have heterogeneous sensors, the accuracy of participant sensors varied between $1\mu g/m^3$ and $20\mu g/m^3$. Then, we use an existing routing API to generate a minimum of 5 alternative routes for each participant to meet a realistic scenario and to have a substantial search space. Furthermore, to avoid ending up with very long alternative routes, we only consider routes that are at most 30% longer than the shortest path.

In the second step, synthetic sensor measurements are generated using the standard errors of the sensors and a normal distribution at each point of the map that a participant's route passes through.

5.6.4 Computing the similarity

In order to implement the similarity-based algorithm, we need to compute the similarity metric between the different routes. This metric should consider segments that do not really overlap but are very close to each other. This may happen when two participants pass by the same road but in opposite directions or parallel sidewalks. For this reason, we build a buffer around each segment of a route, and then compute the similarity between the buffers. The size of the buffer highly influences the similarity metric. Indeed, the larger the buffer is, the higher the similarity value. This size should also be adapted to the spatial resolution of the available data. Figure 5.5 shows an example of two routes that do not exactly overlap but are close enough to be considered when computing the similarity metric. For the following validation tests, we choose a buffer size of 60 meters around the segment (i.e., 30 meters from each side of the segment).



(a) (b)

Figure 5.5: An example of two routes not overlapping but close enough (a) routes not considered overlapping without a buffer (b) routes considered overlapping using a buffer

5.6.5 Clusters construction

The construction of clusters is a crucial phase in the cluster-based route selection approach. It represents the foundation upon which the whole algorithm is built. Therefore, the choice of the optimal number of clusters to construct has a great importance. On one hand, choosing few clusters lowers the similarity threshold. As a result, more points are clustered together without showing much resemblance. On the other hand, a large number of clusters seeks very similar points. Hence, the clusters may gather very few points.

In order to choose an adequate number of clusters in our context, we have evaluated the MAE of the estimation while varying the number of clusters from 250 to 1500 with a step of 250 clusters and 20 iterations for each step. Results depicted in Table 5.1 show that the lowest MAE value is reached with 1250 clusters.

Number of clusters	250	500	750	1000	1250	1500
MAE ($\mu g/m^3$)	5.19	5.08	5.13	5.11	4.97	5.16

Table 5.1: MAE vs the number of clusters with 40 participants

5.7 Performance evaluation and discussion

To evaluate the performance of our proposal, we first compared multiple spatial interpolation models using the same route selection strategy in order to get insights on which model performs better in estimating NO_2 concentrations. The best model was then used in the second part, which compares our proposed route selection strategies with the previously pressed baseline approaches, in terms of estimation error and traveled distance, using a variable number of participants increasing from 15 to 40 participants.

5.7.1 Comparison of the performance of MLR, KNN, and XGBoost

We conducted multiple simulations with the MLR, KNN, and XGBoost, using the similarity-based route selection approach, while increasing the number of participants from 15 to 40 and randomly generating sensor errors between $1\mu g/m^3$ to $10\mu g/m^3$. The number of neighbors for KNN was fixed to 5, based on the previous results from Section 4.3.4. Figure 5.6 depicts the results of the average MAE in function of the number of participants, obtained after 20 iterations. We observe that as the number of participants increases, the prediction error decreases for all three models. XGBoost outperforms KNN by around 30% and MLR by nearly 42% in terms of MAE. Moreover, we present in Table 5.2 the average execution time of the three models for one iteration. The table shows that MLR is the fastest model, while KNN is the slowest among the three. The reason that KNN is much slower compared to the tests in Section 4.2.1 is that the dataset used to perform the prediction is much larger in this study, which makes the prediction step computationally intensive for KNN, because it needs to loop over the entire dataset to find the closest points to each unsampled one. The observations from Figure 5.6 along with those of Table 5.2 give hints about which model is more suitable for a real life scenario. In our case, we chose XGBoost for the upcoming evaluations.

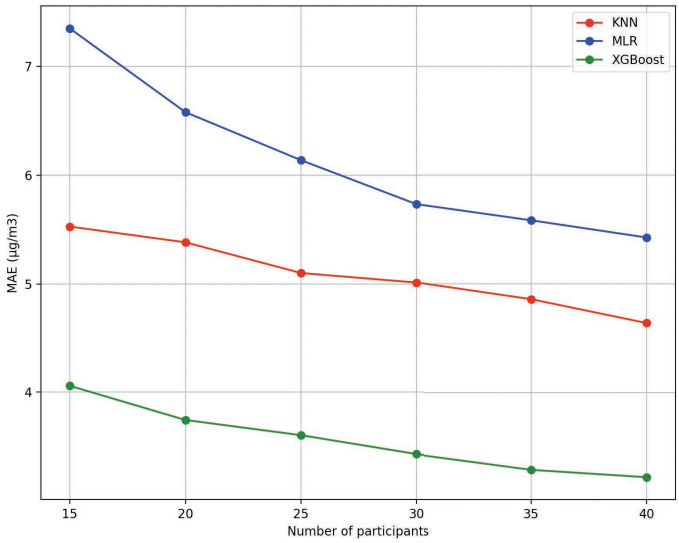


Figure 5.6: MAE vs number of participants

Estimation method	KNN	MLR	XGBoost
Average Execution time (seconds)	12.95	0.31	0.73

Table 5.2: Average execution time of 1 iteration For KNN, multiple linear regression, and XGBoost

5.7.2 Comparison of route selection algorithms

With the aim of evaluating the impact of route selection strategies on the estimation of NO_2 concentrations, we carried out multiple tests considering the shortest-path-based routing, the longest-path-based routing, and the two proposed route selection

algorithms, i.e., the similarity-based routing, and the cluster-based routing. The two proposed solutions have the same goal, that is, offering the most informative routes possible while taking into account the correlations between them. However, the difference lies in the evaluation of the relationship between the routes. The similarity-based routing computes the similarity between participant routes and tries to reduce the overlapping between the selected routes, while the cluster-based routing takes advantage of hierarchical clustering to form groups of similar points and then tries to maximize the set of visited groups.

The performance of the four route selection algorithms are evaluated using XGBoost, based on the observations from the previous test. Figure 5.7 depicts the MAE of estimation in function of the number of participants. The experiments are performed with a number of participants increasing from 15 to 40 and repeated 20 times. At each iteration, we randomly select a new set of participants with sensor errors between $1\mu g/m^3$ to $20\mu g/m^3$. Results clearly show that increasing the number of users helps decreasing the estimation error due to the presence of more routes and hence a larger area being covered. An interesting observation is that while the shortest-path-based routing has the worst performance in terms of MAE of the estimation, the similarity-based and the cluster-based solutions outperforms the longest-path-based routing. This means that choosing the longest path is not always a good decision to improve the estimation. Indeed, the similarity-based route selection performs up to 15.42% better than the shortest-path-based algorithm and 3.49% better than the longest-path-based solution, while the cluster-based solution surpasses the shortest-path-based and the longest-path-based by 16.84% and 5.22%, respectively.

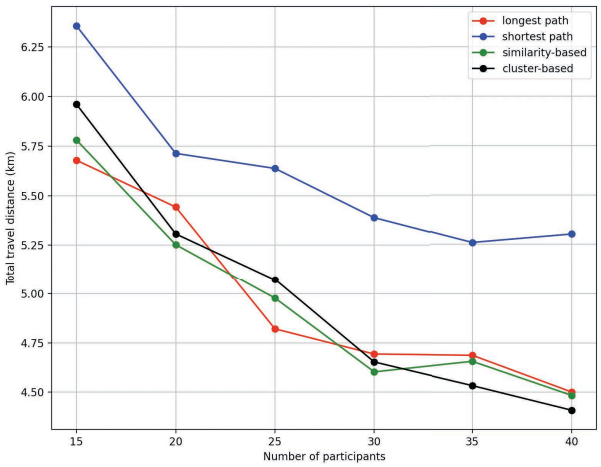


Figure 5.7: MAE vs number of participants

In addition, we investigate the impact of the different route selection approaches on the travel distance of participants. For this purpose, we calculate the total traveling distance for each route selection approach. Figure 5.8 shows the total travel distance accumulated for all participants. As expected, the shortest-path-based algorithm offers the shortest distance by definition and hence outperforms all the other solutions. Nonetheless, results reveal that our proposed solutions are also more efficient than the longest-path selection in terms of traveled distance. Indeed, the similarity-based algorithm improves the travel distance by up to 32.22% compared to the longest-path-based algorithm and 15.50% compared to the cluster-based solution, while being 13.68 % longer than the shortest-path. These promising results state that our proposed solutions give the lowest error while offering shorter routes compared to the best of baseline algorithms.

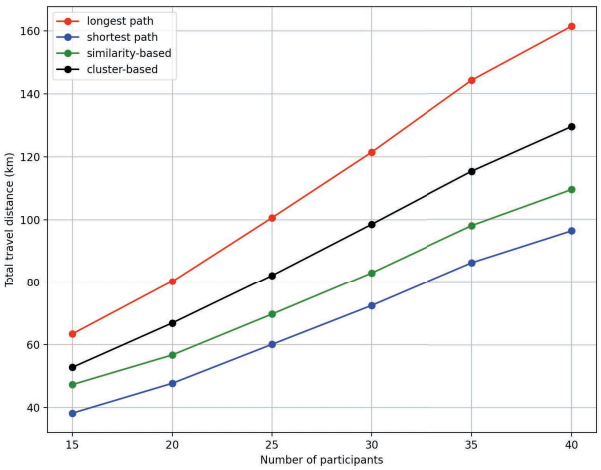
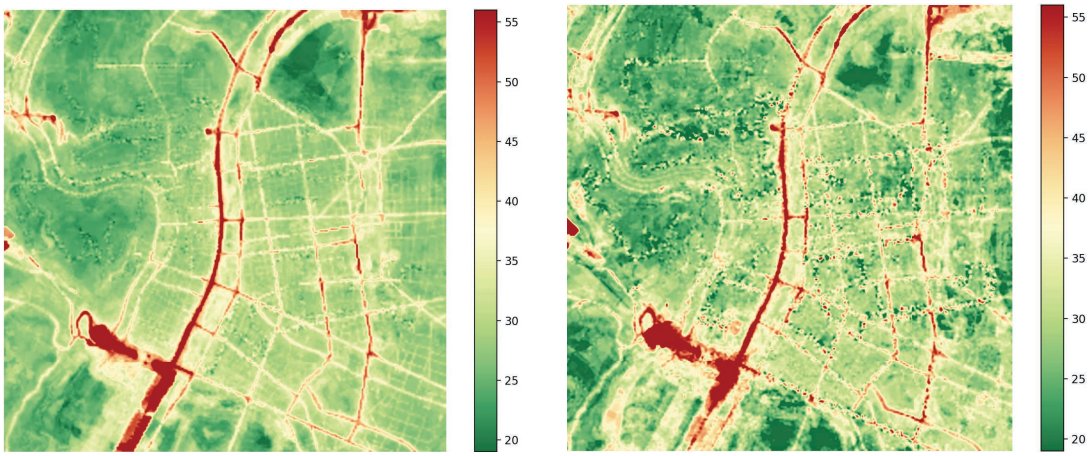


Figure 5.8: Total travel distance for the different route selection algorithms

In addition, to get an idea on the impact of the sensing error over the estimation quality, we show in Figure 5.9 two heatmaps of the estimated NO_2 concentrations using the similarity-based algorithm, achieved with 40 users and sensor standard errors between $1\mu g/m^3$ and $5\mu g/m^3$ (left) and sensor standard errors ranging from $1\mu g/m^3$ and $20\mu g/m^3$ (right). We can visually observe that the estimation with lower sensing errors is more smooth and less noisy than the one with higher sensing errors.



(a) (b)

Figure 5.9: Heatmap of NO_2 concentrations, estimated with 40 users and the similarity-based algorithm with (a) sensor standard errors between $1\mu g/m^3$ and $5\mu g/m^3$ (b) sensor standard errors between $1\mu g/m^3$ and $20\mu g/m^3$

5.8 Conclusion

In this chapter, we addressed the problem of route selection in participatory sensing with low-cost sensors. We proposed two route selection algorithms that take into consideration the characteristics of low-cost sensors in the decision process. The similarity-based routing algorithm aims to give spatially dispersed routes to maximize coverage using the Sørensen-Dice coefficient, a commonly used metric in image segmentation. The cluster-based routing algorithm takes advantage of hierarchical

clustering to build groups of similar points, then tries to visit as many groups as possible to increase the diversity of the collected information. Both algorithms perform in a greedy manner by suggesting at each iteration a route for the user with the most accurate sensor. We compared our solutions to two baseline route selection algorithms, namely the longest-path-based and the shortest-path-based algorithms.

We showed through the results that our route selection approaches can obtain close results to the longest-path-based approach or even outperform it, while being efficient regarding the travel distance. Our solution is adapted to a scenario in which participants do not necessarily arrive at the same time. Moreover, the idea behind the clustering-based approach can be used to find the safest route possible in terms of exposure to air pollution, by reducing the number of high polluted locations to be visited.

In the next chapter, we provide a general conclusion to summarize all the work that has been achieved during this thesis, as well as some future research directions that can be explored.

Chapter 6

Conclusion and perspectives

Today, mobile crowdsensing and low-cost wireless sensor networks form together an attractive and promising approach in various fields. Low-budget applications can make use of this solution and benefit from the cost-efficiency of low-cost WSNs and the crowd density in order to reach large-scale deployments and collect extensive data.

In this chapter, we summarize the contributions of this thesis and subsequently point out some perspectives on possible future research directions.

6.1 Main conclusion

Throughout this Ph.D. thesis, we addressed the use of low-cost WSN in mobile crowdsensing for air quality monitoring, while focusing on three key elements: 1) the design of low-cost participatory air quality monitoring systems, 2) the analysis of dense low-cost WSN data and their benefit in generating high-resolution air pollution maps, and 3) the selection of the participant routes in order to improve the knowledge of the phenomenon. We provided details on the design of the 3M'Air participatory air quality monitoring platform, along with some engineering guidelines that take into consideration the characteristics of the addressed application. We leveraged small, battery-powered, and low-cost sensor nodes equipped with three environmental sensing probes and long-range wireless communication. The proposed solution was compared to reference sensors, and the first results showed that our sensor readings are satisfactory. The 3M'Air platform operated during multiple sensing campaigns that were organized in the city of Lyon with the help of the 3M'Air project partners.

The collected measurements from the sensing campaign helped us to evaluate the performance of our nodes on the field and the consistency of their data. Moreover, this dataset allowed us to study the relationship between the energy consumption, the sensing quality, and the sensing rate. Furthermore, we considered a large-scale low-cost WSN deployment with the help of numerical simulations, and designed a general framework that enables the comparison of data assimilation and regression approaches, while taking into account the low accuracy of low-cost sensors. Through the obtained results, we observed that data assimilation outperformed all the considered regression methods, thanks to its capacity to consider both simulation and measurement errors. In addition, the results pointed out that increasing the number of deployed sensors is not always suitable for regression methods, especially when the sensing error is significantly large.

Our last contribution during this thesis focused on the problem of route selection in low-cost participatory sensing. We introduced two new route selection algorithms

that consider the low accuracy of low-cost sensors in their decision process. The first algorithm aims at reducing overlapping between participant routes using a metric inspired by the imagery field. The second algorithm leverages agglomerative clustering in order to maximize coverage in terms of distinct locations. Both algorithms were compared to baseline route selection approaches, and the impact of their strategy on the assessment of air quality was evaluated. The results indicated that our solution can help assessment models to achieve good performance, while suggesting shorter routes compared to the best baseline algorithm.

6.2 Extensions and future work

The contributions of this thesis form a basis on which we could build new solutions and address other challenges in order to improve our knowledge of air quality.

6.2.1 Future extensions of our participatory sensing platform

Our platform is operational and has been used in several measurement campaigns in Lyon. However, we believe that there is still room for improvement. To this aim, we are planning to improve our platform by adding the possibility to remotely manage the nodes. This will give us the ability to change the sampling/transmission parameters of the nodes and to put certain sensors/receivers in sleep mode. Therefore, the web application will not be used for data visualization only, but also for nodes administration. Regarding the node, we could add BLE support, which will give us the ability to use, when possible, the GPS of smartphones instead of the integrated GPS module, which could save us some energy. Another possible improvement is the support of downloading sensor data by just connecting the node to a computer or a smartphone via BLE.

Moreover, our sensing nodes are not limited to be used on foot, but can also be mounted on other mobile platforms, such as buses or bicycles [31]. However, some challenges need to be addressed in order for our node to operate properly. First, we need to evaluate the impact of the air flow on the reading of the sensing probes. In addition, we might need to adapt the design of the nodes to provide natural ventilation on 360° instead of the current 180°. Another parameter to be adapted is the sensing rate. In fact, buses and bicycles move faster than pedestrians, which means that with the current sensing rate of 20 seconds, the node will cover fewer locations as the traveled distance at each cycle is longer compared to the current scenario. Therefore, nodes should perform measurements at a higher rate to maintain a high spatial resolution. This could imply adopting another communication technology that is more adapted for this use case.

6.2.2 Expanding the network's lifetime

The sensing duty cycling consists of putting some sensors/receivers in sleep mode unless they are performing measurements. Such duty cycle behavior is limited by the convergence time of the sensors embedded in the node. The convergence time is indeed the duration that a sensor needs in order to reach a steady state in stationary conditions and then to output valid readings. Every sensor has its own stabilization time that depends not only on the type of the probe but may also depend on the duration of the sleep phase. Based on the data sheets, the convergence time of the

DHT22 is about 2 seconds [97], while the PM sensor's convergence time is about 30 seconds [96]. Based on the first experiments conducted in [42], the convergence time of the Alphasense NO_2 sensor in stationary conditions varies from around 50 seconds to ten minutes when the previous sleep phase goes from one minute to one hour, respectively.

Based on these observations and those on the impact of the sensing rate on the energy consumption and the sensing quality, we are planning to conduct more analysis in order to propose adaptive sampling [134] in function of the meteorological conditions, the node's trajectory, speed, and the convergence time of the sensors. This may lead to irregular sampling intervals in order to expand the node's lifetime, while enhancing the overall air quality estimation. We also believe that it would be interesting to investigate whether the GPS receiver can be deactivated at some sensing cycles and the location of a measurement predicted, based on the time of the measure and its value. This would make it possible to extend the battery life of the node even further. Another solution that can be leveraged to expand the operating time of the network is the notion of "rendez-vous" presented in [135]. Besides the problem of sensor calibration, this solution could be used to put on sleep mode some sensors when there are more than one in a given spatial buffer.

6.2.3 Extensions of our route selection solutions

The first simulations that we performed using our proposed route selection algorithms showed promising results. Yet, we still need to validate their performance in a real life scenario. Currently, the clustering-based algorithm takes a significant amount of time to find the routes, which is not suitable, especially in a scenario where routes are not known *a priori*. Therefore, we need to optimize the algorithm in order to reduce its computational cost. In addition, the similarity-based algorithm highly depends on the size of the buffer surrounding the routes. In turn, the size of the buffer depends on the sensing rate and the spatial resolution of the measurements. We believe that the impact of the buffer's size on the selection strategy should be investigated in order to achieve better results.

Furthermore, the temporal aspect is worth taking into consideration in the decision process. Indeed, in a near real-time monitoring scenario, where participants do not start the sensing task at the same time, only recent segments should be kept in memory for the route selection algorithm, i.e., the algorithm should take into consideration the freshness of the measurement. Additionally, route planning could be used in order to better lead the participants through locations where the estimation error is large.

Bibliography

[1] World Health Organization, "Burden of disease from the joint effects of household and ambient air pollution for 2016," 2018. [Online]. Available: https://www.who.int/airpollution/data/AP_joint_effect_BoD_results_May2018.pdf

[2] I. Manisalidis, E. Stavropoulou, A. Stavropoulos, and E. Bezirtzoglou, "Environmental and health impacts of air pollution: a review," *Frontiers in public health*, vol. 8, 2020.

[3] G. Cannistraro, M. Cannistraro, A. Cannistraro, A. Galvagno, and F. Engineer, "Analysis of air pollution in the urban center of four cities sicilian," *Int. J. Heat Technol*, vol. 34, no. 2, pp. S219–S225, 2016.

[4] European Environment Agency,, "Air quality in europe — 2020 report," 2020. [Online]. Available: https://www.eea.europa.eu/publications/air-quality-in-europe-2020-report

[5] A. Dechezleprêtre, N. Rivers, and B. Stadler, "The economic cost of air pollution: Evidence from europe," 2019.

[6] F. Chen and Z. Chen, "Cost of economic growth: Air pollution and health expenditure," *Science of The Total Environment*, vol. 755, p. 142543, 2021.

[7] D. Li, W. Liao, A. J. Rigden, X. Liu, D. Wang, S. Malyshev, and E. Shevliakova, "Urban heat island: Aerodynamics or imperviousness?" *Science Advances*, vol. 5, no. 4, p. eaau4299, 2019.

[8] A. Mohajerani, J. Bakaric, and T. Jeffrey-Bailey, "The urban heat island effect, its causes, and mitigation, with reference to the thermal properties of asphalt concrete," *Journal of environmental management*, vol. 197, pp. 522–538, 2017.

[9] Y.-H. Ryu, J.-J. Baik, and S.-H. Lee, "Effects of anthropogenic heat on ozone air quality in a megacity," *Atmospheric environment*, vol. 80, pp. 20–30, 2013.

[10] L. Soulhac, P. Salizzoni, F.-X. Cierco, and R. Perkins, "The model sirane for atmospheric urban pollutant dispersion; part i, presentation of the model," *Atmospheric environment*, vol. 45, no. 39, pp. 7379–7395, 2011.

[11] L. Menut, B. Bessagnet, D. Khvorostyanov, M. Beekmann, N. Blond, A. Colette, I. Coll, G. Curci, G. Foret, A. Hodzic *et al.*, "Chimere 2013: a model for regional atmospheric composition modelling," *Geoscientific model development*, vol. 6, no. 4, pp. 981–1028, 2013.

[12] M. Sorek-Hamer, A. C. Just, and I. Kloog, "The use of satellite remote sensing in epidemiological studies," *Current opinion in pediatrics*, vol. 28, no. 2, p. 228, 2016.

[13] F. Karagulian, M. Barbiere, A. Kotsev, L. Spinelle, M. Gerboles, F. Lagler, N. Redon, S. Crunaire, and A. Borowiak, "Review of the performance of low-cost sensors for air quality monitoring," *Atmosphere*, vol. 10, no. 9, p. 506, 2019.

[14] M. Gerboles, L. Spinelle, and A. Borowiak, "Measuring air pollution with low-cost sensors," *European Commission*, 2017, jRC107461.

[15] R. E. Mohamed, A. I. Saleh, M. Abdelrazzak, and A. S. Samra, "Survey on wireless sensor network applications and energy efficient routing protocols," *Wireless Personal Communications*, vol. 101, no. 2, pp. 1019–1055, 2018.

[16] A. Mondal, I. S. Misra, and S. Bose, "Building a low cost solution using wireless sensor network for agriculture application," in *2017 International Conference on Innovations in Electronics, Signal Processing and Communication (IESC)*. IEEE, 2017, pp. 61–65.

[17] R. Hawi, G. Okeyo, and M. Kimwele, "Smart traffic light control using fuzzy logic and wireless sensor network," in *2017 Computing Conference*. IEEE, 2017, pp. 450–460.

[18] A. Boubrima, W. Bechkit, and H. Rivano, "A new wsn deployment approach for air pollution monitoring," in *2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 2017, pp. 455–460.

[19] T. Lin, H. Rivano, and F. Le Mouël, "A survey of smart parking solutions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 12, pp. 3229–3253, 2017.

[20] U. Gogate and J. Bakal, "Healthcare monitoring system based on wireless sensor network for cardiac patients," *Biomedical & Pharmacology Journal*, vol. 11, no. 3, p. 1681, 2018.

[21] A. B. Noel, A. Abdaoui, T. Elfouly, M. H. Ahmed, A. Badawy, and M. S. Shehata, "Structural health monitoring using wireless sensor networks: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1403–1423, 2017.

[22] J. Ding, M. Nemati, C. Ranaweera, and J. Choi, "Iot connectivity technologies and applications: A survey," *arXiv preprint arXiv:2002.12646*, 2020.

[23] H. Alobaidy, J. Mandeep, R. Nordin, and N. F. Abdullah, "A review on zigbee based wsns: concepts, infrastructure, applications, and challenges," *Int. J. Electr. Electron. Eng. Telecommun*, vol. 9, no. 3, pp. 189–198, 2020.

[24] B. Foubert and N. Mitton, "Long-range wireless radio technologies: A survey," *Future internet*, vol. 12, no. 1, p. 13, 2020.

[25] J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen, and M. Pettissalo, "On the coverage of lpwans: range evaluation and channel attenuation model for lora technology," in *2015 14th international conference on its telecommunications (itst)*. IEEE, 2015, pp. 55–59.

[26] T. Bala, V. Bhatia, S. Kumawat, and V. Jaglan, "A survey: issues and challenges in wireless sensor network," *Int. J. Eng. Technol*, vol. 7, no. 2, pp. 53–55, 2018.

[27] A. Capponi, C. Fiandrino, B. Kantarci, L. Foschini, D. Kliazovich, and P. Bouvry, "A survey on mobile crowdsensing systems: Challenges, solutions, and opportunities," *IEEE communications surveys & tutorials*, vol. 21, no. 3, pp. 2419–2465, 2019.

[28] R. K. Ganti, F. Ye, and H. Lei, "Mobile crowdsensing: current state and future challenges," *IEEE communications Magazine*, vol. 49, no. 11, pp. 32–39, 2011.

[29] A. Karim, A. Siddiqa, Z. Safdar, M. Razzaq, S. A. Gillani, H. Tahir, S. Kiran, E. Ahmed, and M. Imran, "Big data management in participatory sensing: Issues, trends and future directions," *Future Generation Computer Systems*, vol. 107, pp. 942–955, 2020.

[30] Y. Gao, W. Dong, K. Guo, X. Liu, Y. Chen, X. Liu, J. Bu, and C. Chen, "Mosaic: A low-cost mobile sensing system for urban air quality monitoring," in *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*. IEEE, 2016, pp. 1–9.

[31] D. Hasenfratz, O. Saukh, C. Walser, C. Hueglin, M. Fierz, and L. Thiele, "Pushing the spatio-temporal resolution limit of urban air pollution maps," in *2014 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 2014, pp. 69–77.

[32] A. Anjomshoaa, F. Duarte, D. Rennings, T. J. Matarazzo, P. deSouza, and C. Ratti, "City scanner: Building and scheduling a mobile sensing platform for smart city services," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4567–4579, 2018.

[33] X. Xie, I. Semanjski, S. Gautama, E. Tsiligianni, N. Deligiannis, R. T. Rajan, F. Pasveer, and W. Philips, "A review of urban air pollution monitoring and exposure assessment methods," *ISPRS International Journal of Geo-Information*, vol. 6, no. 12, p. 389, 2017.

[34] M. A. Fekih, W. Bechkit, H. Rivano, M. Dahan, F. Renard, L. Alonso, and F. Pineau, "Participatory air quality and urban heat islands monitoring system," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–14, 2020.

[35] M. A. Fekih, I. Mokhtari, W. Bechkit, Y. Belbaki, and H. Rivano, "On the regression and assimilation for air quality mapping using dense low-cost wsn," in *International Conference on Advanced Information Networking and Applications*, vol. 1151, 2020, pp. 566–578.

[36] M. Fekih, W. Bechkit, and H. Rivano, "On the data analysis of participatory air pollution monitoring using low-cost sensors," in *ISCC 2021-26th IEEE Symposium on Computers and Communications*, 2021.

[37] B. Maag, Z. Zhou, and L. Thiele, "A survey on sensor calibration in air pollution monitoring deployments," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4857–4870, 2018.

[38] F. Delaine, B. Lebental, and H. Rivano, "In situ calibration algorithms for environmental sensor networks: A review," *IEEE Sensors Journal*, vol. 19, no. 15, pp. 5968–5978, 2019.

[39] D. Charpin, "Microcapteurs mobiles connectés de polluants atmosphériques: évolution ou révolution?" *Bulletin de l'Académie Nationale de Médecine*, vol. 203, no. 7, pp. 613–617, 2019.

[40] G. Dardier and F. Jabot, "Ambassad'air: A french example of how citizen sensing can fuel the smart city," *European Journal of Public Health*, vol. 30, no. Supplement_5, pp. ckaa166–135, 2020.

[41] D. Hasenfratz, O. Saukh, C. Walser, C. Hueglin, M. Fierz, T. Arn, J. Beutel, and L. Thiele, "Deriving high-resolution urban air pollution maps using mobile sensor nodes," *Pervasive and Mobile Computing*, vol. 16, pp. 268–285, 2015.

[42] A. Boubrima, "Deployment and scheduling of wireless sensor networks for air pollution monitoring," Ph.D. dissertation, University of Lyon, 2019.

[43] S. Ferdoush and X. Li, "Wireless sensor network system design using raspberry pi and arduino for environmental monitoring applications," *Procedia Computer Science*, vol. 34, pp. 103–110, 2014.

[44] L. Lombardo, S. Corbellini, M. Parvis, A. Elsayed, E. Angelini, and S. Grassini, "Wireless sensor network for distributed environmental monitoring," *IEEE Transactions on Instrumentation and Measurement*, vol. 67, no. 5, pp. 1214–1222, 2017.

[45] P. Arroyo, J. L. Herrero, J. I. Suárez, and J. Lozano, "Wireless sensor network combined with cloud computing for air quality monitoring," *Sensors*, vol. 19, no. 3, p. 691, 2019.

[46] S. Metia, H. A. Nguyen, and Q. P. Ha, "Iot-enabled wireless sensor networks for air pollution monitoring with extended fractional-order kalman filtering," *Sensors*, vol. 21, no. 16, p. 5313, 2021.

[47] N. Castell, M. Kobernus, H.-Y. Liu, P. Schneider, W. Lahoz, A. J. Berre, and J. Noll, "Mobile technologies and services for environmental monitoring: The citi-sense-mob approach," *Urban climate*, vol. 14, pp. 370–382, 2015.

[48] M. Artzrouni, "Mathematical demography," 2005.

[49] Y. Wu and M.-C. Hung, "Comparison of spatial interpolation techniques using visualization and quantitative assessment," *Applications of Spatial Statistics*, pp. 17–34, 2016.

[50] J. Li and A. D. Heap, "Spatial interpolation methods applied in the environmental sciences: A review," *Environmental Modelling & Software*, vol. 53, pp. 173–189, 2014.

[51] ——, "A review of spatial interpolation methods for environmental scientists," *Geoscience Australia*, p. 137, 2008.

[52] G. Matheron, "Principles of geostatistics," *Economic geology*, vol. 58, no. 8, pp. 1246–1266, 1963.

[53] N. Cressie, "The origins of kriging," *Mathematical geology*, vol. 22, no. 3, pp. 239–252, 1990.

[54] M. Bocquet, H. Elbern, H. Eskes, M. Hirtl, R. Žabkar, G. Carmichael, J. Flemming, A. Inness, M. Pagowski, J. Pérez Camaño *et al.*, "Data assimilation in atmospheric chemistry models: current status and future prospects for coupled chemistry meteorology models," *Atmospheric chemistry and physics*, vol. 15, no. 10, pp. 5325–5358, 2015.

[55] S. Baillargeon, "Le krigeage: revue de la théorie et application à l'interpolation spatiale de données de précipitations," 2005.

[56] N. Cressie, "Spatial prediction and ordinary kriging," *Mathematical geology*, vol. 20, no. 4, pp. 405–421, 1988.

[57] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015.

[58] M. Alloghani, D. Al-Jumeily, J. Mustafina, A. Hussain, and A. J. Aljaaf, "A systematic review on supervised and unsupervised machine learning algorithms for data science," *Supervised and unsupervised learning for data science*, pp. 3–21, 2020.

[59] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[60] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[61] J. Kerckhoffs, M. Wang, K. Meliefste, E. Malmqvist, P. Fischer, N. A. Janssen, R. Beelen, and G. Hoek, "A national fine spatial scale land-use regression model for ozone," *Environmental research*, vol. 140, pp. 440–448, 2015.

[62] H. I. Rhys, *Machine Learning with R, the tidyverse, and mlr*. Manning Publications, 2020.

[63] X. Ren, Z. Mi, and P. G. Georgopoulos, "Comparison of machine learning and land use regression for fine scale spatiotemporal estimation of ambient air pollution: Modeling ozone concentrations across the contiguous united states," *Environment International*, vol. 142, p. 105827, 2020.

[64] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to linear regression analysis*. John Wiley & Sons, 2021.

[65] M. Tranmer and M. Elliot, "Multiple linear regression," *The Cathie Marsh Centre for Census and Survey Research (CCSR)*, vol. 5, no. 5, pp. 1–5, 2008.

[66] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[67] G. Biau and E. Scornet, "A random forest guided tour," *Test*, vol. 25, no. 2, pp. 197–227, 2016.

[68] J. Friedman, T. Hastie, R. Tibshirani *et al.*, *The elements of statistical learning*. Springer series in statistics New York, 2001, vol. 1, no. 10.

[69] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[70] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho *et al.*, "Xgboost: extreme gradient boosting," *R package version 0.4-2*, vol. 1, no. 4, pp. 1–4, 2015.

[71] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[72] H. Taud and J. Mas, "Multilayer perceptron (mlp)," in *Geomatic Approaches for Modeling Land Change Scenarios*. Springer, 2018, pp. 451–455.

[73] K. Gurney, *An introduction to neural networks*. CRC press, 1997.

[74] M. W. Gardner and S. Dorling, "Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences," *Atmospheric environment*, vol. 32, no. 14-15, pp. 2627–2636, 1998.

[75] R. Navares and J. L. Aznarte, "Predicting air quality with deep learning lstm: Towards comprehensive models," *Ecological Informatics*, vol. 55, p. 101019, 2020.

[76] R. Yan, J. Liao, J. Yang, W. Sun, M. Nong, and F. Li, "Multi-hour and multi-site air quality index forecasting in beijing using cnn, lstm, cnn-lstm, and spatiotemporal clustering," *Expert Systems with Applications*, vol. 169, p. 114513, 2021.

[77] J. Li, H. Zhang, C.-Y. Chao, C.-H. Chien, C.-Y. Wu, C. H. Luo, L.-J. Chen, and P. Biswas, "Integrating low-cost air quality sensor networks with fixed and satellite monitoring systems to study ground-level pm2. 5," *Atmospheric Environment*, vol. 223, p. 117293, 2020.

[78] A. Marjovi, A. Arfire, and A. Martinoli, "Extending urban air quality maps beyond the coverage of a mobile sensor network: data sources, methods, and performance evaluation," in *Proceedings of the International Conference on Embedded Wireless Systems and Networks*, no. CONF, 2017.

[79] P.-C. Chen and Y.-T. Lin, "Exposure assessment of pm2. 5 using smart spatial interpolation on regulatory air quality stations with clustering of densely-deployed microsensors," *Environmental Pollution*, p. 118401, 2021.

[80] C. C. Lim, H. Kim, M. R. Vilcassim, G. D. Thurston, T. Gordon, L.-C. Chen, K. Lee, M. Heimbinder, and S.-Y. Kim, "Mapping urban air quality using mobile sampling with low-cost sensors and machine learning in seoul, south korea," *Environment international*, vol. 131, p. 105022, 2019.

[81] L. Alonso and F. Renard, "Compréhension du microclimat urbain lyonnais par l'intégration de prédicteurs complémentaires à différentes échelles dans des modèles de régression," *Climatologie*, vol. 17, p. 2, 2020.

[82] O. Pannekoucke, "Modélisation des structures locales de covariance des erreurs de prévision à l'aide des ondelettes," Ph.D. dissertation, Université Paul Sabatier-Toulouse III, 2008.

[83] N. Daget, "Revue des méthodes d'assimilation," 2007.

[84] F. Bouttier and P. Courtier, "Data assimilation concepts and methods march 1999," *Meteorological training course lecture series. ECMWF*, vol. 718, p. 59, 2002.

[85] A. Tilloy, V. Mallet, D. Poulet, C. Pesin, and F. Brocheton, "Blue-based no 2 data assimilation at urban scale," *Journal of Geophysical Research: Atmospheres*, vol. 118, no. 4, pp. 2031–2040, 2013.

[86] C. V. Nguyen and L. Soulhac, "Data assimilation methods for urban air quality at the local scale," *Atmospheric Environment*, vol. 253, p. 118366, 2021.

[87] U. Kumar, K. De Ridder, W. Lefebvre, and S. Janssen, "Data assimilation of surface air pollutants (o3 and no2) in the regional-scale air quality model aurora," *Atmospheric environment*, vol. 60, pp. 99–108, 2012.

[88] S. Lopez-Restrepo, A. Yarce, N. Pinel, O. L. Quintero, A. Segers, and A. W. Heemink, "Urban air quality modeling using low-cost sensor network and data assimilation in the aburrá valley, colombia," *Atmosphere*, vol. 12, no. 1, p. 91, 2021.

[89] M. Al-Azzawi and R. Raeside, "Modeling pedestrian walking speeds on sidewalks," *Journal of Urban Planning and Development*, vol. 133, no. 3, pp. 211–219, 2007.

[90] R. Rastogi, S. Chandra *et al.*, "Pedestrian flow characteristics for different pedestrian facilities and situations," 2013.

[91] K. Kotkar, R. Rastogi, and S. Chandra, "Pedestrian flow characteristics in mixed flow conditions," *Journal of Urban Planning and Development, ASCE*, vol. 136, no. 3, pp. 23–33, 2010.

[92] "LoRaWAN^{TM} 1.0.3 Specification," LoRa Alliance Technical Committee, 2018. [Online]. Available: https://lora-alliance.org/wp-content/uploads/2020/11/lorawan1.0.3.pdf

[93] M. C. Bor, J. Vidler, and U. Roedig, "Lora for the internet of things." in *EWSN*, vol. 16, 2016, pp. 361–366.

[94] H. Price, "Air analysis | field portable instruments for the measurement of airborne hazards," in *Encyclopedia of Analytical Science (Third Edition)*, P. Worsfold, C. Poole, A. Townshend, and M. Miró, Eds. Oxford: Academic Press, 2019, pp. 40 – 43. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780124095472126800

[95] Alphasense Ltd, "No2-b43f nitrogen dioxide sensor." [Online]. Available: http://www.alphasense.com/WEB1213/wp-content/uploads/2019/09/NO2-B43F.pdf

[96] Shenzhen Co., Ltd, "Hm-3300/3600," 2018. [Online]. Available: https://github.com/SeeedDocument/Grove-Laser_PM2.5_Sensor-HM3301/raw/master/res/HM-3300%263600_V2.1.pdf

[97] Aosong Electronics O., Ltd, "Digital-output relative humidity & temperature sensor/module dht22," 2019. [Online]. Available: https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf

[98] Microchip Technology Inc, "Sam d21 family," 2018. [Online]. Available: https://www.mouser.fr/pdfdocs/SAM-D21-Family-Datasheet-DS40001882C.pdf

[99] Murata Investment Co., Ltd, "Sub-g module data sheet," 2018. [Online]. Available: https://wireless.murata.com/datasheet?/RFM/data/type_abz.pdf

[100] Adafruit learning system, "Adafruit 4-channel adc breakouts," 2019. [Online]. Available: https://cdn-learn.adafruit.com/downloads/pdf/adafruit-4-channel-adc-breakouts.pdf?timestamp=1575440557

[101] GlobalTop Technology Inc., "Fgpmmopa6h," 2011. [Online]. Available: https://cdn-shop.adafruit.com/datasheets/GlobalTop-FGPMMOPA6H-Datasheet-V0A.pdf

[102] Adafruit learning system, "Adafruit powerboost 1000c," 2019. [Online]. Available: https://cdn-learn.adafruit.com/downloads/pdf/adafruit-powerboost-1000c-load-share-usb-charge-boost.pdf?timestamp=1575440467

[103] "RP002-1.0.2 LoRaWAN Regional Parameters," LoRa Alliance Technical Committee, 2020. [Online]. Available: https://lora-alliance.org/wp-content/uploads/2020/11/RP_2-1.0.2.pdf

[104] "The things network." [Online]. Available: https://www.thethingsnetwork.org/

[105] N. Blenn and F. Kuipers, "Lorawan in the wild: Measurements from the things network," *arXiv preprint arXiv:1706.03086*, 2017.

[106] N. Zinas, S. Kontogiannis, G. Kokkonis, S. Valsamidis, and I. Kazanidis, "Proposed open source architecture for long range monitoring. the case study of cattle tracking at pogoniani," in *Proceedings of the 21st Pan-Hellenic Conference on Informatics*. ACM, 2017, p. 57.

[107] "Placepod." [Online]. Available: https://www.pnicorp.com/placepod/

[108] "Vinduino r3 sensor station." [Online]. Available: https://www.thethingsnetwork.org/marketplace/product/vinduino-r3-sensor-station

[109] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, "A study of lora: Long range & low power networks for the internet of things," *Sensors*, vol. 16, no. 9, p. 1466, 2016.

[110] E. L. Lehmann and H. J. D'Abrera, *Nonparametrics: statistical methods based on ranks*. Holden-Day, 1975.

[111] Y. K. Cheung and J. H. Klotz, "The mann whitney wilcoxon distribution using linked lists," *Statistica Sinica*, pp. 805–813, 1997.

[112] Dostmann electronic GmbH, "Log 32 th pdf- data logger for temperature and humidity." [Online]. Available: https://dostmann-electronic.de/product/log-32-th-pdf-data-logger-for-temperature-and-humidity.html

[113] TFA Dostmann, "Protective cover for outdoor transmitter." [Online]. Available: https://www.tfa-dostmann.de/en/produkt/protective-cover-for-outdoor-transmitter/

[114] Davis Instruments, "Radiation shield(7714) specification sheets." [Online]. Available: https://www.davisinstruments.com/product_documents/weather/spec_sheets/DS7714_6838_Rad%20Shields_Spec_Sheet.pdf

[115] M. Schumacher, "Two-sample tests of cramér–von mises-and kolmogorov–smirnov-type for randomly censored data," *International Statistical Review/Revue Internationale de Statistique*, pp. 263–281, 1984.

[116] W. H. Kruskal and W. A. Wallis, "Use of ranks in one-criterion variance analysis," *Journal of the American statistical Association*, vol. 47, no. 260, pp. 583–621, 1952.

[117] E. Nam, S. Kishan, R. W. Baldauf, C. R. Fulper, M. Sabisch, and J. Warila, "Temperature effects on particulate matter emissions from light-duty, gasoline-powered motor vehicles," *Environmental science & technology*, vol. 44, no. 12, pp. 4672–4677, 2010.

[118] M. J. Kim, "Changes in the relationship between particulate matter and surface temperature in seoul from 2002–2017," *Atmosphere*, vol. 10, no. 5, p. 238, 2019.

[119] Data Grand Lyon, "Métropole de lyon," 2020. [Online]. Available: https://www.grandlyon.com/metropole/59-communes.html

[120] L. Soulhac, P. Salizzoni, P. Mejean, D. Didier, and I. Rios, "The model sirane for atmospheric urban pollutant dispersion; part ii, validation of the model on a real case study," *Atmospheric environment*, vol. 49, pp. 320–337, 2012.

[121] L. Soulhac, C. V. Nguyen, P. Volta, and P. Salizzoni, "The model sirane for atmospheric urban pollutant dispersion. part iii: Validation against no2 yearly concentration measurements in a large urban agglomeration," *Atmospheric environment*, vol. 167, pp. 377–388, 2017.

[122] M. Mead, O. Popoola, G. Stewart, P. Landshoff, M. Calleja, M. Hayes, J. Baldovi, M. McLeod, T. Hodgson, J. Dicks *et al.*, "The use of electrochemical sensors for monitoring urban air quality in low-cost, high-density networks," *Atmospheric Environment*, vol. 70, pp. 186–203, 2013.

[123] A. Boubrima, W. Bechkit, and H. Rivano, "On the deployment of wireless sensor networks for air quality mapping: Optimization models and algorithms," *IEEE/ACM Transactions on Networking*, vol. 27, no. 4, pp. 1629–1642, 2019.

[124] C. Bertero, J.-F. Léon, G. Trédan, M. Roy, and A. Armengaud, "Urban-scale no2 prediction with sensors aboard bicycles: A comparison of statistical methods using synthetic observations," *Atmosphere*, vol. 11, no. 9, p. 1014, 2020.

[125] X. Tao and W. Song, "Efficient path planning and truthful incentive mechanism design for mobile crowdsensing," *Sensors*, vol. 18, no. 12, p. 4408, 2018.

[126] S. Ji, Y. Zheng, and T. Li, "Urban sensing based on human mobility," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2016, pp. 1040–1051.

[127] Y. Chen, P. Lv, D. Guo, T. Zhou, and M. Xu, "Trajectory segment selection with limited budget in mobile crowd sensing," *Pervasive and Mobile Computing*, vol. 40, pp. 123–138, 2017.

[128] Y. Liu, B. Guo, Y. Wang, W. Wu, Z. Yu, and D. Zhang, "Taskme: Multi-task allocation in mobile crowd sensing," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2016, pp. 403–414.

[129] W. Gong, B. Zhang, and C. Li, "Location-based online task assignment and path planning for mobile crowdsensing," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1772–1783, 2018.

[130] Y. Chen, D. Guo, and M. Xu, "Prosc+: Profit-driven online participant selection in compressive mobile crowdsensing," in *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*. IEEE, 2018, pp. 1–6.

[131] S. Kosub, "A note on the triangle inequality for the jaccard distance," *Pattern Recognition Letters*, vol. 120, pp. 36–38, 2019.

[132] T. S. Mathai, L. Jin, V. Gorantla, and J. Galeotti, "Fast vessel segmentation and tracking in ultra high-frequency ultrasound images," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2018, pp. 746–754.

[133] B. S. Everitt, S. Landau, M. Leese, and D. Stahl, "Hierarchical clustering," *Cluster analysis*, vol. 5, pp. 71–110, 2011.

[134] Y. Zeng and K. Xiang, "Adaptive sampling for urban air quality through participatory sensing," *Sensors*, vol. 17, no. 11, p. 2531, 2017.

[135] A. Arfire, A. Marjovi, and A. Martinoli, "Model-based rendezvous calibration of mobile sensor networks for monitoring air quality," in *2015 IEEE SENSORS*. IEEE, 2015, pp. 1–4.

FOLIO ADMINISTRATIF

THESE DE L'UNIVERSITE DE LYON OPEREE AU SEIN DE L'INSA LYON

NOM : FEKIH
(avec précision du nom de jeune fille, le cas échéant)

DATE de SOUTENANCE : 04/02/2022

Prénoms : Mohamed Anis

TITRE : Low-cost Wireless Sensor Networks in Participatory Air Quality Monitoring

NATURE : Doctorat

Numéro d'ordre : 2022LYSEI007

Ecole doctorale : InfoMaths

Spécialité : Informatique

RESUME :

Mobile crowdsensing sensing is an emerging and promising paradigm that has attracted much attention in recent years, especially for environmental monitoring. Coupled with the power of low-cost wireless sensor networks (WSN), it leverages population density to collect extensive data in many applications, such as air pollution monitoring. In fact, air pollution is one of the main problems that still suffers from a lack of characterization due to the limitations of traditional assessment methods employed in terms of cost, network size, and flexibility. Mobile crowdsensing and WSN aim at filling this gap by enabling large-scale deployments in order to improve the local knowledge on the phenomenon.

In this thesis, we consider the air quality monitoring application using mobile crowdsensing, while focusing on three parts: 1) the design of low-cost participatory air quality monitoring systems; 2) the analysis of dense low-cost WSN data and their benefit in generating high-resolution air pollution maps; 3) the selection of the participants' paths in order to improve the characterization, while considering the constraints of travel distance and sensor errors. Through this work, we aim to show the potential of using low-cost WSN and participatory sensing in air quality monitoring. In this vein, we conduct substantial experimental work on the design of a participatory air quality monitoring system from scratch. We provide engineering guidelines regarding the design of low-cost participatory environmental monitoring platforms. Moreover, we propose a general framework that allows the comparison of different strategies based on numerical simulations and an adequate estimation of the simulation error as well as the sensing error. We also explore the impact of the sensing rate on the energy consumption and the characterization error. Furthermore, we propose two new algorithms for route selection in participatory sensing. Our approaches take into account the participants' constraints and the characteristics of low-cost WSN.

MOTS-CLÉS : Mobile crowdsensing, low-cost WSN, air quality, participatory sensing

Laboratoire (s) de recherche : CITI

Directeur de thèse :
- Hervé RIVANO, professeur des universités à INSA-Lyon, Directeur de thèse.
- Walid BECHKIT, maître de conférences à INSA-Lyon, co-directeur de thèse.

Président de jury :

Composition du jury :
- Nathalie MITTON, directrice de recherche INRIA (rapportrice)
- Nicolas MONTAVONT, professeur des universités à l'IMT Atlantique (rapporteur)
- Alexandre GUITTON, professeur des universités à l'Université Clermont Auvergne (examinateur)
- Nathalie REDON, maître de conférences à l'IMT Nord Europe (examinatrice)