Perception de l'environnement urbain à l'aide d'une flotte de capteurs sur des vélos. Application à la pollution de l'air

Christophe Bertero

Université Fédérale
Toulouse Midi-Pyrénées

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ FÉDÉRALE TOULOUSE MIDI-PYRÉNÉES

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

Présentée et soutenue le *12/06/2020* **par :**

Christophe BERTERO

PERCEPTION DE L'ENVIRONNEMENT URBAIN À L'AIDE D'UNE FLOTTE DE CAPTEURS SUR DES VÉLOS. APPLICATION À LA POLLUTION DE L'AIR.

JURY

Marie-Pierre GLEIZES	Professeur d'Université	Présidente du Jury
Hervé RIVANO	Professeur	Rapporteur
Sébastien PAYAN	Professeur d'Université	Rapporteur
Sonia BEN MOKHTAR	Directrice de Recherche	Examinatrice
Isabelle CHIAPELLO	Chargée de Recherche	Examinatrice
Jean-François LEON	Chargé de Recherche	Examinateur
Matthieu ROY	Chargé de Recherche	Examinateur
Gilles TREDAN	Chargé de Recherche	Invité

École doctorale et spécialité :
EDSYS : Informatique 4200018
Unité de Recherche :
LAAS - CNRS (UPR 8001), LA (UMR 5560)
Directeurs de Thèse :
Jean-François LEON et *Matthieu ROY*

Remerciements

Mes premiers remerciements vont à toute personne qui se vouera à la lecture ou même au survol de ce document, destination de son écriture. Ensuite, je remercie évidemment mes encadrants, Jean-François Léon, Matthieu Roy et Gilles Tredan. Je remercie particulièrement Matthieu et Gilles pour m'avoir donné la chance de faire cette thèse, Jean-François pour le temps pris pour échanger, me guider et la relecture du manuscrit et Gilles pour m'avoir fait découvrir de nouveaux concepts et de belles publications. Je tiens également à remercier Christophe Zanon avec qui j'ai passé de nombreuses heures à déboguer et Daniel Loche pour les retouches d'image, pour avoir géré la visio-conférence et pour l'aide régulière apportée à l'équipe.

Je remercie mon financeur, NéOCampus, mon université, l'Université Paul Sabatier, et mes laboratoires, le LAAS-CNRS et le Laboratoire d'Aérologie, mon équipe TSF et le CNRS pour avoir mis à ma disposition un cadre adapté pour mener à bien mes recherches ; ainsi qu'EDSYS pour son soutien dans les démarches administratives et sa compréhension. Je remercie ensuite mon jury de thèse pour leurs commentaires et particulièrement Hervé Rivano et Sébastien Payan pour leurs rapports et Sonia Ben Mokhtar pour son intérêt pour le jeu de données collecté lors de cette thèse ; et les relecteurs pour les corrections orthographiques (principalement mes parents).

Je remercie aussi Alice et la Maison du Vélo, l'association de location de vélos pour notre première tentative de déploiement, les expérimentateurs du LAAS-CNRS qui ont permis d'acquérir les données – à vélo en hiver, AtmoSud et Alexandre Armengaud qui ont rapidement partagé leurs données de simulations numériques et Epurtek pour leurs réponses à nos questions sur le système embarqué. Je remercie les enseignants de l'Université Paul Sabatier pour m'avoir donné l'opportunité d'animer tant des cours que des TD et des TP, à des niveaux variés, ainsi que René Sultra et Maria Barthélémy pour l'ouverture d'esprit qu'ils m'auront fait développer.

Enfin, merci aux doctorants de l'équipe pour leur bonne humeur au cours de toutes ces parties de tarot ainsi qu'au cours des différents débats sur la recherche ; merci à Matthieu et Romain pour ces pauses – discussions ; merci aux permanents pour leur convivialité et leurs échanges ; merci à mes co-bureau Éric et Jean-Charles pour nos conversations ; merci à toutes les personnes du LAAS-CNRS qui m'ont aidé à utiliser divers instruments (Xavier Dollat, le labo micro-électronique et Jérôme Manhes, Phillipe Menini et Aymen Sendi, Nicolas Mauran et Alexandre Rumeau) ; merci à Benoît Morgan, Nicolas Rivière, Anaïs Marshall et Yves Auda pour leurs explications ; merci à tous les acolytes des soirées de la « fête de la science » ; merci à Denis Arzelier pour ses mails sur annonce ; merci à ma famille et mes amis pour m'avoir soutenu ; merci à Nicolas pour m'avoir incité à faire cette simulation de vélos et Thomas pour les nombreuses soirées à discuter ; merci à tous ceux que j'oublie au moment d'écrire ces lignes.

Table des matières

Préambule

> *The only generally agreed upon definition of mathematics is "Mathematics is what mathematician's do." which is followed by "Mathematicians are people who do mathematics." What is true about defining mathematics is also true about many other fields: there is often no clear, sharp definition of the field. In the face of this difficulty many people, including myself at times, feel that we should ignore the discussion and get on with doing it. But as George Forsythe points out so well in a recent article, it does matter what people in Washington D.C. think computer science is. According to him, they tend to feel that it is a part of applied mathematics and therefore turn to the mathematicians for advice in the granting of funds. And it is not greatly different elsewhere; in both industry and the universities you can often still see traces of where computing first started, whether in electrical engineering, physics, mathematics, or even business. Evidently the picture which people have of a subject can significantly affect its subsequent development. Therefore, although we cannot hope to settle the question definitively, we need frequently to examine and to air our views on what our subject is and should become.*
>
> – Richard Hamming, *One Man's View of Computer Science.*

En guise de préambule, nous présentons nos réflexions **subjectives** sur la perception en général. Pour cela, nous allons d'abord tenter de répondre à la question posée par Hamming, qui est, définir le sujet dans lequel nous nous inscrivons – l'informatique et les mathématiques. Puis, nous questionnerons le lien entre représentations théoriques et perception au travers de visions plus consensuelles.

L'informatique est au « comment ? » ce que les mathématiques sont au « quoi ? ». Une tentative de réponse.

Un point commun entre informatique et mathématiques, sans doute le plus évident, est la volonté de répondre à la question susvisée de façon argumentée, à l'aide d'un raisonnement cohérent.

Lorsque la réponse est mathématique, elle l'est sous forme de démonstration. Elle vise à montrer un théorème, autrement dit une proposition de réponse, dans un cadre particulier défini par un système formel. Un système formel est l'outil du raisonnement : un langage (un alphabet et une méthode de construction de mots), un dictionnaire (des mots particuliers, les axiomes), et un ensemble de règles de déduction. Ces règles de déduction permettent de former les mots valides et l'ensemble de ces mots, exhibés ou non, forme la théorie engendrée. Avec ce vocabulaire, un théorème n'est qu'un mot particulier dont il faut montrer la validité à l'aide d'une succession de mots valides, la démonstration. Dès lors, pour mieux répondre il faut utiliser l'outil de raisonnement correctement, mais aussi en imaginer de nouveaux [1] et les comparer. Voilà l'ambition que nous attribuons aux

1. Pour davantage de détails, Jérôme Cottanceau (El Jj) vulgarise ce concept en vidéo : `https://www.youtube.com/watch?v=0_ZzZzvxnP0`.

mathématiques !

Une question fréquente lorsqu'on s'interroge sur les mathématiques est « sont-elles une découverte ou une invention humaine ? », et une suivante est « ont-elles vocation à être appliquées ? ». La question sous-jacente est en fait « s'agit-il d'une science ou d'un art ? ». En effet, ce qui différencie fondamentalement la science de l'art est la vocation des œuvres produites : en art, la production est autonome et s'adresse aux sens et aux émotions ; en science, la production repose intrinsèquement sur les autres productions et cherche à établir des connaissances utiles. La réponse est conciliante : ce sont les deux à la fois ; cela dépend du regard du mathématicien et de ce qu'il recherche. D'une part, la motivation de la recherche peut être guidée par une finalité, une instance de question à laquelle il faut répondre ; et dans ce cas, la réponse est souvent formulée à l'aide d'une approche scientifique par l'exploration de nouveaux mots valides dans une théorie connue. D'autre part, la motivation peut être davantage contemplative, par l'étude des formes de réponses possibles, en imaginant de nouvelles théories et en unifiant celles connues. Cette démarche artistique est conduite par une esthétique indéniable mais non définie de façon consensuelle. Elle pourrait être définie par la citation bien connue de Boileau-Despréaux : « Ce qui se conçoit bien s'énonce clairement – Et les mots pour le dire arrivent aisément ». En d'autre termes, les belles théories émerveillent de par l'émergence intuitive d'une profusion de théorèmes utiles, au regard de la simplicité des axiomes. Sans rentrer dans le détail, les seuls cinq axiomes de Peano permettent d'engendrer l'arithmétique ; leur expressivité est saisissante.

Dans les deux cas, la relation au réel des mathématiques n'est pas une nécessité, mais une source d'inspiration. Plus précisément, c'est par l'abstraction que le lien entre réel et mathématiques s'établit ; mais les mathématiques restent des concepts sans matérialité physique jusqu'à leur mise en application.

Tout ce qui a été dit pour les mathématiques peut s'étendre à l'informatique. Mais tout comme l'observation précède l'action, les mathématiques sous-tendent l'informatique [2]. En effet, en mathématiques, l'accent est porté sur la description et la représentation en vue de la caractérisation d'objets – c'est-à-dire les distinguer à l'aide de mots particuliers i.e. d'équations – et des théories ; alors qu'en informatique, l'enjeu est de fournir des méthodes pour résoudre une tâche – à l'aide d'algorithmes. Pour accomplir cette tâche de façon optimale, l'informatique exploite ces théories et les théorèmes connus. En ce sens, une proposition de définition du concept d'information est l'interface entre ces théories, i.e. le moyen de passer de l'une à l'autre, de poser comme nouveaux axiomes les conclusions déduites de la théorie en amont.

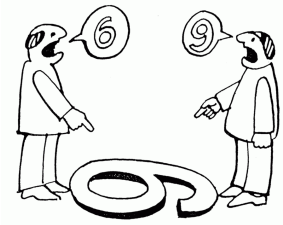
Ce qui distingue les mathématiques de l'informatique, c'est donc la manière de répondre i.e. de démontrer, par la description ou par l'action : l'informatique étudie les interactions entre les objets caractérisés mathématiquement. Plus précisément, nous pensons que toute réponse informatique peut se résumer au cycle look–compute–move (Suzuki et Yamashita, 1999; Flocchini et *al.*, 2000) : observation de la situation, calcul de possibilités d'actions, choix de l'une des possibilités et mise en œuvre, observation de la situation engendrée, etc. Métaphoriquement, les mathématiques sont comme « créer des objets et

2. Néanmoins l'informatique ne date pas du siècle dernier et se développe conjointement avec les mathématiques ; l'algorithme d'Euclide, transcrit vers 300 avant Jésus-Christ, en est une preuve.

les regarder sous tous les angles », et l'informatique « disposer ces objets entre eux ».

Espérons que, à l'avenir, l'informatique se focaliser sur l'autorégulation systémique et adaptative, notamment inspirées de systèmes biologiques, et non par régulations (indicateurs, empilement de contraintes). En ce sens, la cybernétique, étude des mécanismes d'information des systèmes complexes – voire cognition collective –, n'est qu'une redéfinition plus ambitieuse de l'informatique ; et l'ingénierie est la mise en pratique de ces mécanismes de pensée. Et "Computer Science is no more about computers than astronomy is about telescopes" [3] ou pour être davantage explicite, l'ordinateur n'est qu'un outil d'ingénierie permettant de vérifier dans le monde réel les concepts d'interactions formulés par l'informatique.

Perception et informatique.



Just because you are right doesn't mean I'm wrong.

La notion de perception est encore aujourd'hui très discutée en philosophie, psychologie et neurologie. Nous retiendrons la définition suivante : la perception est le processus qui permet à une entité de prendre conscience de son environnement, à l'aide de ses expériences. Cette définition renvoie à la notion de conscience, « l'un des mots les plus difficiles à définir » selon André Comte-Sponville. Néanmoins, nous la définissons du point de vue phénoménologique : la conscience est ce qui structure et organise les concepts utiles à nos actions dans le monde. Autrement dit, la perception est le processus qui permet à une entité de se créer une théorie des lois qui régissent le monde. Plus précisément, une théorie se distingue communément d'un modèle par le fait que le modèle est une interprétation d'une théorie : chaque axiome de la théorie est supposé être un énoncé vrai dans le modèle ; la notion de vérité n'existe pas dans une théorie.

Dretske distingue deux niveaux de perception : la perception des « choses » et la perception des « faits » (Dretske, 2000; Dokic, 2000). Le premier, naturel et élémentaire, est l'assimilation d'un évènement à l'aide des sens. Les sens traduisent un ensemble de stimuli, un percept, en une description destinée au cerveau ou un autre système d'information, via impulsions électriques ou n'importe quel autre canal. Par exemple, la vision humaine utilise la lumière pour la transformer en impulsions électriques ; la paire de cryptes vibrissales sur le museau des dauphins sert à détecter les variations du champ électrique. Le second est l'assimilation d'un évènement à l'aide de croyances, concepts subjectifs de l'individu, par l'abstraction.

3. Cette citation, d'origine discutée et tantôt attribuée à Dijkstra tantôt attribuée à Fellows, s'est répandue dans les années 1990.

La Figure 1 est un ensemble de taches noires réparties de façon particulière (« choses ») qui peuvent être perçu comme un dalmatien reniflant près d'un arbre (« faits »).

FIGURE 1 – Taches noires et dalmatien.

Sans la connaissance a priori du concept de dalmatien et d'arbre, cette vision est impossible. A l'extrême, cette perception des « faits » peut être une illusion, une surinterprétation en informatique ou une paréidolie en art, comme le fait de voir des visages dans les tableaux de Shupliak, illustré Figure 2.

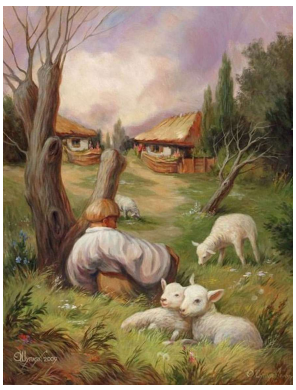


FIGURE 2 – Paysage de campagne qui évoque un visage, peint par Shupliak.

Plus fortement, Bateson soutient que les différences entre les objets ne sont perceptibles qu'au travers des concepts qui les décrivent. Rosen exprime la même idée et la schématise sous forme de « relations de modélisation » – c'est-à-dire de relations de création de modèle : le monde des représentations, propre à chaque individu, et le monde naturel sont liés par deux relations qui permettent de passer de l'un à l'autre (cf. Figure 3) ; et ainsi tout système naturel se modélise lui-même et peut être appréhendé comme une carte de relations entre modèles. Bateson définit le concept d'information comme la différence d'encodage d'attributs présents dans deux modèles. Cette définition souligne la pluralité et donc la partialité de la sélection d'attributs pour décrire un évènement. Selon ce point de vue, toute connaissance n'est que croyance (i.e. concept) subjective, communément admise par recherche d'objectivité, relative aux connaissances présupposées. Feynman instancie cette vision à l'aide du concept d'énergie :

There is a fact, or if you wish, a law, governing all natural phenomena that are known to date. There is no known exception to this law—it is exact so far as we know. The law is called the conservation of energy. [...] It is important to realize that in physics today, we have no knowledge of what energy is. However, there are formulas for calculating some numerical quantity, [...] This approach is called the principle of virtual work, because in order to apply this argument we had to imagine that the structure moves a little—even though it is not really moving or even movable. We use the very small imagined motion to apply the principle of conservation of energy.

– The Feynman Lectures on Physics

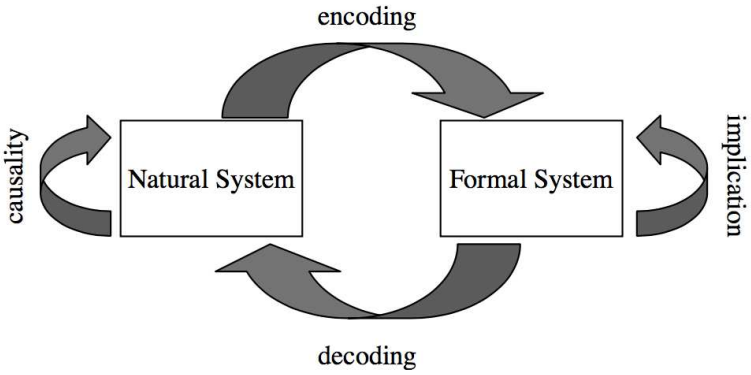


FIGURE 3 – Relations de modélisation de Rosen (Kineman et Anil Kumar, 2007).

L'actualisation des concepts de « causalité » et « implication » sont le « déterminisme » et la « cohérence ». Pour reprendre une idée déjà exprimée par Maxwell, la distinction entre déterminisme et causalité réside dans notre perception en temps fini du système :

Il y a une maxime selon laquelle les mêmes causes produisent les mêmes effets [...]. Mais il y a une autre maxime qu'il ne faut pas confondre : que des causes semblables produisent des effets semblables. Cela est vrai uniquement si de petites variations sur les circonstances initiales produisent de petites variations sur l'état final du système. Cela est vrai dans beaucoup de cas, mais il y en a d'autres pour lesquels une petite variation initiale peut produire de grands changements dans l'état final.

– Maxwell

Autrement dit, la sensibilité des conditions initiales dans les phénomènes physiques remet en cause la capacité humaine en la perception du déterminisme réel des choses, en supposant qu'un tel déterminisme existe et que l'on puisse caractériser de façon absolue les conditions initiales (sans perception relative – aux connaissances et donc au temps – des choses).

Ceci nous conduit à rejeter l'axiome du tiers exclu (et donc ZFC, au profit des mathématiques constructivistes) pour décrire le monde réel en général. En effet, comment s'assurer que la perception de la négation d'une proposition n'est pas en fait induite par des hypothèses absurdes sensiblement identiques aux hypothèses formulées ? Par exemple,

imaginer une dimension dans un autre ensemble que celui des naturels conduit à la formalisation des fractales. Une contraposée ne peut prétendre nier ce qui n'est encore imaginé.

Toutefois, cet axiome est utile et suffisant pour prédire d'autres phénomènes, tels que la probabilité de faire un pile ou une face en lançant une pièce. En outre, d'autres axiomes sont plus facilement acceptables sous hypothèse de cohérence de l'univers engendré – sinon, à quoi bon la science ? – lorsqu'un paradoxe est soulevé. Ainsi, certains axiomes peuvent être refusés par recherche de cohérence, mais aucun ensemble d'axiomes cohérent ne semble plus légitime qu'un autre pour décrire le monde ; et le potentiel de perception est induit par la capacité expressive des axiomes.

Une vision complémentaire, où la modélisation est guidée par l'action, est décrite par le cycle perception-action de Neisser (cf. Figure 4). Il résume le processus perpétuel d'interactions avec le monde qui permet de construire des concepts : l'objet d'étude modifie l'état des connaissances, l'état des connaissances dirige la recherche de nouvelles, la recherche de nouvelles connaissances induit une sélection des caractéristiques de l'objet d'étude (Neisser, 1976).

Enfin, la perception artificielle est l'ensemble des méthodes, développées en science des données, que peuvent adopter les machines pour appréhender leur environnement à la manière de l'humain. D'un point de vue informatique, le triptyque look–compute–move se retrouve dans la vision de Neisser.

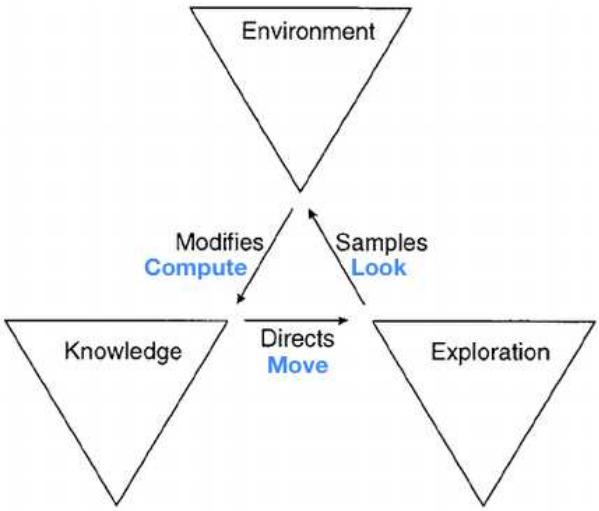


FIGURE 4 – Cycle perception-action et look–compute–move — adapté d'après Neisser (1976).

Bibliographie

DOKIC, J. . Le cercle bipolaire. Intentionnalité et contenu perceptif. De la perception à l'action. Contenus perceptifs et perception de l'action. PUF, Paris, 2000. https://hal.archives-ouvertes.fr/ijn_00000279.

DRETSKE, F. . Conscious Experience. Perception, Knowledge and Belief : Selected Essays. Cambridge University Press, 2000. DOI : 10.1017/CBO9780511625312.008.

Flocchini, P. , Prencipe, G. , Santoro, N. et Widmayer, P. . Distributed coordination of a set of autonomous mobile robots. Proceedings of the IEEE Intelligent Vehicles Symposium, 2000. DOI : 10.1109/IVS.2000.898389.

Kineman, J. J. et Anil Kumar, K. . Primary natural relationship : Bateson, Rosen, and the Vedas. Kybernetes, 2007. DOI : 10.1108/03684920710777838.

Neisser, U. . Cognition and reality : principles and implications of cognitive psychology. Freeman, 1976. ISBN 978-0-7167-0477-5.

Suzuki, I. et Yamashita, M. . Distributed Anonymous Mobile Robots : Formation of Geometric Patterns. SIAM Journal on Computing, 1999. DOI : 10.1137/S009753979628292X.

Introduction

La ville peut être vue comme le résultat de la concentration dans l'espace et dans le temps des hommes et de leurs activités. Elle est empreinte de leurs interactions quotidiennes et évolue pour répondre aux besoins de la communauté. Une ville intelligente (ou Smart City en anglais) est une zone urbaine qui utilise les technologies de l'information et de la communication pour gérer des services et établir des actions à mener.

L'Internet des Objets élargit ce concept à une infrastructure mondiale pour mettre en réseau tout objet pouvant être source d'informations. Il s'agit de tous les objets personnels de la vie courante (téléphone, réfrigérateur, brosse à dents...), mais aussi des objets communs (rues, transports, musées...) et de nouveaux à imaginer, particulièrement dans le domaine de la santé.

Concrètement, une ville « intelligente » possède deux caractéristiques. La première est de surveiller en continu l'état de l'infrastructure de la ville et des interactions grâce à des capteurs répartis géographiquement afin de permettre aux systèmes urbains de s'autoréguler. La seconde est de mettre en place des processus cognitifs pour traiter les données collectées – notamment à l'aide de l'apprentissage automatique et des algorithmes de communications décentralisées dits « répartis » – afin de répondre à une question.

Dans cette thèse, nous nous sommes intéressés au vélo et à ses utilisateurs. Ce moyen de transport a été inventé au XIX^e^ siècle à l'aube de la seconde révolution industrielle, qui verra s'imposer le pétrole comme la source d'énergie incontournable. Or les conséquences néfastes (pollution de l'air et réchauffement climatique) de l'utilisation de ce même pétrole pousse aujourd'hui les citoyens à se tourner davantage vers ce mode de transport. Vieux moyen de transport mais nouvel objet de mobilité, le vélo s'impose dans les villes qui se veulent durables.

Ainsi, peut-on transformer notre vieux « biclou » en un « smart bike » ? C'est la question à laquelle nous nous sommes intéressés dans le cadre du projet BICLUE (BICycle-based Laboratory of Urban Evolutions) initié par le LAAS-CNRS en collaboration avec le Laboratoire d'Aérologie et soutenu par l'initiative NeoCampus[4], une simulation de ville intelligente grandeur nature au sein du campus de l'Université Paul-Sabatier (Gleizes et *al.*, 2017).

Les destinations du smart bike peuvent être nombreuses. Il peut s'agir de renseigner sur l'utilisation du vélo en lui-même (trajectoire, pression des pneus, structure) ou sur l'utilisateur (vitesse, équilibre, habitudes), ou encore sur l'environnement (état de la chaussée, zone de freinage, bruit ambiant, conditions météorologiques et qualité de l'air). C'est dans ce dernier cadre que le travail de cette thèse s'inscrit, c'est-à-dire utiliser le vélo comme un vecteur permettant d'échantillonner l'espace urbain.

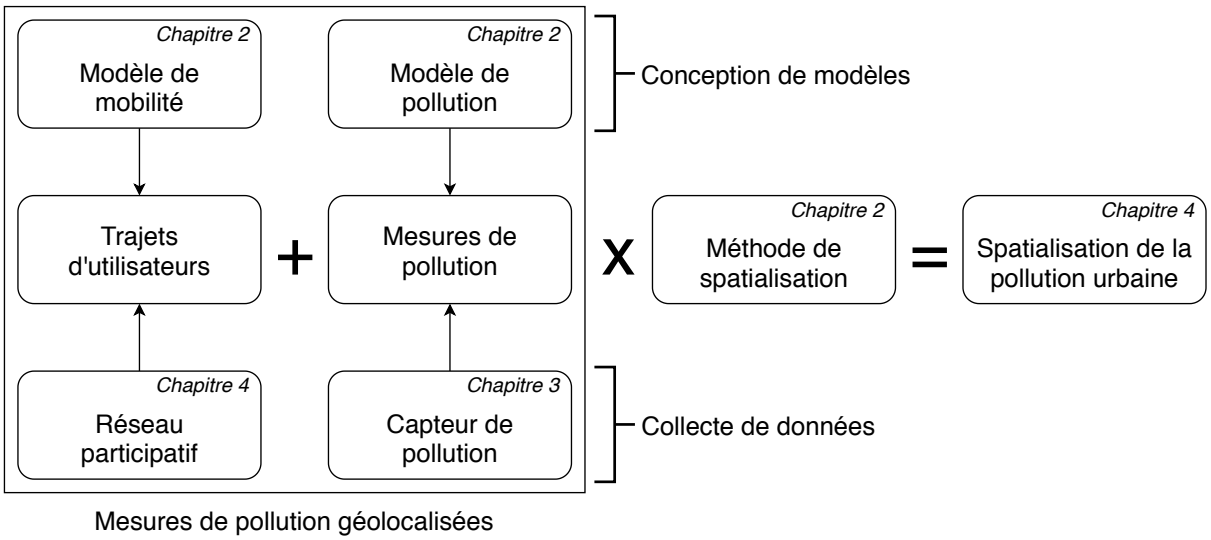
Le domaine d'application retenu concerne directement les utilisateurs : la qualité de l'air. Outre le fait que de plus en plus de citoyens choisissent ce mode de déplacement dit « propre » par souci de l'environnement, la fréquence respiratoire lors du déplacement à vélo augmente significativement et donc la qualité de l'air respiré est d'autant plus importante pour le cycliste. Il ne s'agit pas ici de s'intéresser à l'exposition des cyclistes lors

4. `https://www.irit.fr/neocampus/fr/`

de leur activité, mais d'utiliser le vélo pour étudier la répartition des polluants atmosphériques aux échelles spatiales et temporelles propres à ce mode de déplacement. La mobilité de la collecte de données permet de complémenter l'approche classique de surveillance de la pollution de l'air à l'aide de stations fixes déployées sur le territoire.

Le premier objectif est donc de concevoir un système embarqué capable de suivre les déplacements des cyclistes et de détecter certains polluants atmosphériques afin de collecter des informations sur les variations spatio-temporelles de la pollution de l'air. Le second objectif est de développer un cadre analytique adapté à l'exploitation de ces nouvelles données.

Le schéma suivant synthétise les travaux qui seront présentés dans la suite de ce document. Afin d'obtenir une estimation de la pollution, nous décomposons le problème en trois : l'acquisition de trajets d'utilisateurs, la mesure de la pollution le long de ces trajets et la spatialisation de la pollution dans l'espace. Chaque problème peut être étudier soit à l'aide de modèles théoriques soit à l'aide de données réelles. Nous choisissons de présenter d'abord la conception de modèles et leurs relations, puis la collecte des données et enfin la confrontation de ces deux moyens de résolution du problème.



Présentation synthétique de l'organisation des différents chapitres de cette thèse.

Dans le premier chapitre, nous présentons les enjeux de la modélisation de la pollution de l'air en ville et les moyens de collecter l'information à l'aide d'un réseau de capteurs mobiles. Ce chapitre introduit les concepts fondamentaux de la modélisation selon trois paradigmes : l'apprentissage automatique, les systèmes répartis et la création de connaissances. Il introduit également le domaine de l'aérologie au travers de la complexité du phénomène de pollution de l'air en ville, et expose les défis techniques et scientifiques de la modélisation de la pollution de l'air.

Le deuxième chapitre s'articule autour de la conception d'un instrument de mesure de la pollution de l'air dédié au vélo et qui peut être produit en grand nombre. Pour cela, nous caractérisons la forme du réseau de capteurs nécessaire à la modélisation, d'une part à l'aide de la littérature et d'autre part à l'aide d'une simulation de la collecte et de l'exploitation des données par une flotte de capteurs de taille et de période d'échantillonnage

variables. Cette simulation repose sur un modèle de mobilité décrivant le comportement des cyclistes, un modèle de qualité de l'air fournissant des mesures de la pollution (en NO_2 et en PM_{10}) et trois méthodes statistiques de spatialisation.

Le troisième chapitre traite de notre réalisation d'un tel instrument, en s'intéressant au matériel puis au logiciel. Avant tout, nous présentons le choix du micro-capteur MiCS-4514. Il s'agit d'un capteur à métal-oxyde semi-conducteur (MOx) qui cible les gaz CO et NO_2. Puis, nous exposons la conception de notre instrument de mesure et nous évaluons la qualité de nos capteurs en laboratoire et *in situ*. Nous en déduisons une méthode pour traiter le signal de nos capteurs.

Le quatrième chapitre présente les deux déploiements de cet instrument dans la ville de Toulouse auprès de deux réseaux de participants ; d'abord d'une association de location de vélos, puis de « vélo-taffeurs » avertis du fonctionnement de l'instrument de mesure et les données qui en découlent. Un vélo-taffeur est une personne qui se rend quotidiennement au travail à vélo. Ce profil est particulièrement intéressant pour décrire les mutations en cours concernant ce mode de transport, mais aussi parce que les heures d'utilisation du vélo correspondent aux pics de pollution dus aux automobiles. Enfin, nous présentons une estimation des niveaux de pollution de l'air dans Toulouse en exploitant l'analyse du deuxième chapitre.

Bibliographie

Gleizes, M.-P. , Boes, J. , Lartigue, B. et Thiébolt, F. . neOCampus : A Demonstrator of Connected, Innovative, Intelligent and Sustainable Campus. International Conference on Intelligent Interactive Multimedia Systems and Services (KES-IIMSS 2017), Vilamoura, Portugal, 21/06/17-23/06/17. Springer, 2017. DOI : 10.1007/978-3-319-59480-4_48.

Chapitre 1

De la collecte de l'information au modèle

Le dessin n'est pas la forme, il est la manière de voir la forme.

– Edgar Degas

Sommaire

1.1 Introduction

Ce chapitre propose une vision pluridisciplinaire de la modélisation de la pollution de l'air en milieu urbain. Le problème de sa perception artificielle est présenté au travers de la science des données. Nous commençons par expliquer ce qu'est la science des données, puis trois paradigmes jugés intéressants pour la modélisation de la pollution en ville : l'apprentissage automatique, les réseaux répartis de mesure et le processus de création de connaissance.

Ensuite, nous soulignons la complexité du phénomène de la pollution en ville au travers de la législation relative à sa surveillance, la diversité de ses sources d'émission et les différentes représentations des concentrations à l'échelle de la ville.

Enfin, nous exposons les enjeux relatifs à l'acquisition des observations à l'aide d'un réseau réparti. Le premier est la correspondance entre la mesure du capteur et la concentration réelle en gaz par étalonnage. Le second et le troisième s'articulent autour du substrat de ce réseau réparti, dans notre cas les participants et la plateforme de transport ; l'objectif étant de maximiser leur couverture spatiale. Le quatrième est la vie privée des utilisateurs au regard des données de mobilités collectées par le réseau.

1.2 Perception artificielle et science des données

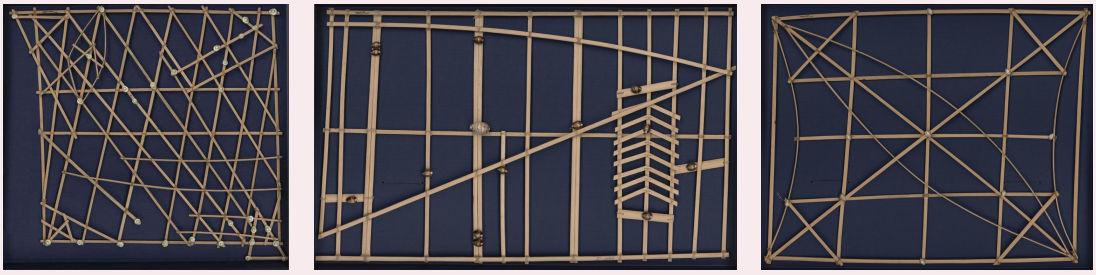
1.2.1 Exemple historique et trilogie de Jeff Wu

L'exemple historique suivant présente les enjeux de la perception au travers du cas concret de l'estimation de la houle dans les îles Marshall par la population locale, et leurs relations avec la science des données. En effet, le processus de création de connaissances naturellement mis en place par les Marshallais esquisse les principes du formalisme actuellement admis dans le monde de la recherche et de l'ingénierie.

Exemple historique : cartes à bâtonnets des îles Marshall

La navigation a toujours été un enjeu crucial dans les îles du Pacifique. Afin de faciliter leur déplacement d'îles en îles, les Marshallais confectionnaient des « cartes à bâtonnets » de la houle – conséquence de l'interaction entre terre, mer et vent –, transmises de père en fils. La culture du secret qui entoure ces cartes a conduit à de nombreuses variantes et leur découverte par l'Occident ne date que de 1862. Elles furent utilisées jusqu'après la Seconde Guerre Mondiale car de qualité comparable aux cartes produites par les technologies contemporaines.
Ces cartes étaient fabriquées avec des matériaux locaux rudimentaires, généralement à partir de tiges de cocotiers pour représenter l'énergie des vagues (front d'onde et rayon d'onde après réfraction, mais aussi réflexion et diffraction) et de coquillages ou intersections de bâtonnets pour représenter les sources de perturbations (atolls, récifs...).
Pour les élaborer, les Marshallais devaient se géolocaliser et estimer l'intensité de la houle. Pour cela, ils se servaient d'étoiles fixes et de chansons rythmées pour mesurer le temps et l'espace ; et stimulaient leur sensation de l'énergie des vagues en

s'allongeant dans leur pirogue (Davenport, 1960). Les informations ainsi collectées étaient mémorisées directement sous forme de cartes schématiques (Feinberg et *al.*, 2003).
Cette représentation abstraite, entre les cartes et les graphes, se focalise sur ce qui est jugé comme essentiel : la géométrie de la dynamique des vagues induite par la forme des îles et non les distances absolues. Elle est le fruit de la compréhension et de la modélisation du phénomène par les navigateurs (Ascher, 1995).
Enfin, trois types de cartes existent : le *rebbelith*, qui décrit un ou plusieurs archipels ; le *meddo*, qui se concentre sur une zone plus réduite ; le *mattang*, qui a pour vocation l'enseignement des concepts fondamentaux. Ainsi, les *mattangs* permettent de transmettre les connaissances acquises sur la houle et les dynamiques couramment rencontrées aux abords des îles, mais aussi sur le système de représentation. Les *rebbeliths* et *meddos*, eux, sont plus épurés et indiquent les spécificités de la région.



(a) Rebbelith (b) Meddo (c) Mattang

FIGURE 1.1 – Cartes à bâtonnets des îles Marshall.

Premièrement, les trois étapes fondamentales de la science des données, trilogie proposée par Jeff Wu (Donoho, 2017), se distinguent nettement :

— la collecte de données, capture d'une représentation de l'information, via les voyages en pirogue,
— l'analyse des données et la modélisation du phénomène, via les cartes à bâtonnets,
— la résolution du problème et la prise de décision, via l'utilisation des cartes pour minimiser l'effort fourni pour se déplacer.

Deuxièmement, l'importance de la collecte de données est soulignée par les méthodes sophistiquées élaborées par les Marshallais. Ils assurent des mesures précises par la définition d'un référentiel à l'aide de chants et d'étoiles et l'utilisation de leurs sens, seuls « capteurs » de la réalité dont ils disposent. Issu d'expéditions en pirogue, le coût de l'instrumentation – mis en exergue par la culture du secret – conduit à la coopération.

Troisièmement, l'abstraction Marshallaise concilie la complexité nécessaire à une représentation utile et manipulable de l'environnement, la simplicité nécessaire à sa mémorisation et sa transmission et la flexibilité nécessaire à la naissance d'autres concepts.

Quatrièmement, le perfectionnement de la modélisation – de générations en générations – est essentiel à la conception d'un modèle fiable, et donc de connaissances.

Ainsi, cet exemple souligne l'approche empirique et itérative de la perception vue par la science des données.

1.2.2 Développement de la science des données

Les statistiques peuvent être considérées comme les prémisses de la sciences des données. Étymologiquement, *statistica* provient de l'italien *stato* (« état ») et *statista* (« homme d'état »), et eux même du latin *status* au sens équivoque « situation, état » et « état social, état politique » (Oriol, 2010; Heuschling, 1847) ; et réfère à la nécessité intrinsèque de l'État de caractériser la société pour agir, quelque soit sa vocation affichée (État-providence, État-planificateur, État-stratège...). Par exemple au point que, selon Hérodote, le pharaon Amasis (570–526 avant Jésus-Christ) édicte une loi pour contraindre les égyptiens, sous peine de mort, à déclarer annuellement l'origine de leurs ressources (Hombert et Préaux, 1952).

Initialement, les techniques consistaient essentiellement à recenser des données, notamment cadastrales et démographiques, à vocation descriptive pour les États. Puis, inspirées du concept de science de l'État (*staatskunde*, composé de « staats », l'État, et « kunde », la connaissance), elles sont formalisées au cours du XVIII^{e} siècle. Achenwall est le premier à utiliser le terme *statistik* en allemand, *statistica* en latin moderne, (« relatif à l'État ») pour nommer ce domaine des sciences politiques : l'ensemble des connaissances de l'État Heuschling (1847).

Ensuite, les techniques se sont diversifiées pour mieux servir les besoins de prévisions de l'État. Par exemple, en Angleterre au début du XX^{e} siècle, pour assurer une pendaison mortelle mais non sanguinolente, un tableau de correspondances entre le poids du condamné et la longueur de la corde a été établi au fil des essais [1].

L'évolution des analyses statistiques, sous différentes appellations, conserve cette vocation perceptive mais se généralise à tout l'environnement dans lequel l'humain évolue. Le but de cette science est d'une part, de caractériser un environnement de façon efficace, i.e. assurer un réalisme utile à l'action tout en s'épargnant le plus que possible la mesure, et d'autre part, d'estimer la confiance à accorder à cette caractérisation.

Dans la deuxième moitié du XX^{e} siècle, propulsée par les premiers ordinateurs, la digitalisation de pans entiers de la société ouvre une ère d'abondance pour les données. Cette abondance fait évoluer les pratiques au sein de la statistique.

Dans les années 1970, le concept d'analyse ou science des données émerge en se différenciant des statistiques par la place centrale qu'elle accorde au traitement de ces données. La science des données s'est davantage focalisée sur le développement de techniques d'analyse. Figure centrale de ces développements, Tukey distingue deux types d'analyses complémentaires en fonction de l'approche : déductive, sous le nom « analyse exploratoire de données » et inductive, sous le nom « analyse confirmatoire de données ». L'analyse exploratoire de données inspecte le jeu de données, décèle les valeurs aberrantes, envisage plusieurs relations concernant les causes du phénomène observé, identifie des tendances ou des motifs et enfin conjecture des propriétés sur les données par le biais d'hypothèses statistiques. L'analyse confirmatoire de données détermine la plausibilité d'une hypothèse statistique au regard d'un jeu de données inexploré et confirme ou infirme la valeur de postulat de l'hypothèse. Inspirés de la perception humaine, ces deux modes de raisonnement sont complémentaires dans le processus de création de connaissances.

Fin des années 1980, Gregory Piatetsky-Shapiro propose en parallèle un processus

1. `https://www.youtube.com/watch?v=lgk7hIOVMcI&t=97s`

agnostique de création de connaissances en science des données, sous la terminologie *Knowledge discovery in data bases* (KDD), plus général, incluant l'exploration de données. Il est présenté section 1.3.3.

Rapidement, toute information numérisée devient donnée candidate à l'exploitation. Les entreprises commencent à traiter massivement les données internes à l'entreprise pour mieux cibler les clients et optimiser les décisions. A l'heure actuelle, le nouvel enjeu est de traiter la masse de données collectées continuellement par l'Internet des Objets.

1.3 Paradigmes retenus

La doctrine non-absolutiste du jaïnisme, l'Anekãntavãda présentée en annexe 1, met en scène six aveugles rencontrant pour la première fois un éléphant et proposant des modèles de représentation de ce nouvel objet par comparaison avec des objets déjà connus : l'un fait référence à un mur lorsqu'il heurte le corps de la bête, un autre évoque une lance lorsqu'il touche une défense, etc. Localement, leur description est correcte ; mais elle ne suffit pas à d'écrire l'objet dans sa globalité.

La description du global à partir du local est une problématique fondamentale de la science des données et commune aux paradigmes retenus.

1.3.1 Apprentissage automatique

L'apprentissage automatique est un type d'intelligence artificielle qui épargne l'intervention d'un expert en généralisant des observations.

A partir d'un ensemble $X = \{x_i\}, x_i \in D$ de données associées chacune à un avis de l'expert à émuler $Y = \{y_i\}, y_i \in E$, nous « paramètrons » un sous-ensemble de l'espace des fonctions de D dans E. A chaque jeu de paramètres w correspond une fonction $f_w : D \to E$; nous cherchons alors w tel que $|f_w(X) - Y|$ soit faible.

Domingos (2012) propose une vision simple de l'apprentissage automatique au travers de la formulation suivante :

$$apprentissage = représentation + évaluation + optimisation.$$

Le système de représentation permet de définir l'ensemble des modèles envisageables. La fonction d'évaluation permet de mesurer la vraisemblance d'un modèle au regard des observations. La fonction d'optimisation permet de naviguer dans l'ensemble des modèles envisageables pour trouver celui qui simule au mieux les observations.

La complexité du système de représentation est à adapter au problème. Par exemple, les réseaux de neurones utilisent une structure particulière de graphes pondérés comme système de représentation ; un jeu de données de test pour évaluer la proximité entre la prédiction et la valeur cible ; et un algorithme de descente de gradient, pour pondérer la structure du graphe. Dans le cas où la structure du graphe n'est composée que d'un neurone, seuls les séparateurs linéaires seront « apprenables ». De plus, si la complexité du système de représentation est plus élevée que celle du problème i.e. $|w| >> |X|^{|D|}$, le modèle choisi par l'algorithme d'apprentissage aura un comportement très fortement lié aux observations et une faible capacité de généralisation. Cela s'appelle le surapprentissage.

En outre, plus les observations sont décrites dans un espace à grande dimension, plus elles paraissent éloignées dans l'espace car ont moins de chance d'être similaires. Ce questionnement autour de la représentation des observations en grande dimension est connu sous le nom de *malédiction de la dimensionnalité*.

De surcroît, la nature du système de représentation $f_.$ impacte directement la fonction apprise et l'optimisation de la recherche de celle-ci (vanishing gradient problem).

Encore aujourd'hui, ces questions demeurent essentielles en apprentissage automatique. La force des algorithmes d'apprentissage profond est de considérer de très nombreux systèmes de représentation, de façon générique, à l'aide de neurones (Bengio et *al.*, 2012).

Cette approche souligne le fait que la modélisation reste une approximation et permet de guider l'évaluation des modélisations.

Exploration de fichiers de journalisation et détection d'anomalies

Cette section présente brièvement une publication publiée à ISSRE2017 [a] (cf. annexe 2) qui n'est pas directement reliée à la perception de l'environnement urbain.
Il s'agit d'une méthode de détection automatique d'anomalies d'un système informatique par le traitement en langage naturel des fichiers de journalisation. Les fichiers de journalisation décrivent les évènements qui surviennent dans un système. Ils sont difficilement lisibles par l'Homme car générés par une machine et difficilement interprétables par une machine car à destination de l'Homme. L'idée originale de cette publication est de considérer que l'ensemble des fichiers de journalisation définit une nouvelle langue, comparable au français ou à l'anglais. Dès lors, nous utilisons une méthode du domaine du traitement du langage naturel (Natural Language Processing), word2vec développée par Google, pour transformer un fichier de journalisation en un point d'un espace vectoriel. Ensuite, nous comparons plusieurs algorithmes d'apprentissage supervisé pour la détection d'anomalies à partir de ces points de l'espace vectoriel. Les données d'entraînement et de validation sont générées par des systèmes virtuels sur lesquels nous pouvons simuler des anomalies (problème de mémoire, de temps de communication, etc). La classification des anomalies donne de bons résultats et cette technique semble prometteuse.
Bien que le lien ne soit pas direct avec la perception de l'environnement urbain, la notion de détection en apprentissage automatique est une notion élémentaire qui permet de bien appréhender les tenants et aboutissants de la modélisation.

a. The 28th IEEE International Symposium on Software Reliability Engineering.

1.3.2 Réseaux de capteurs et systèmes répartis

Un réseau de capteurs est un ensemble de capteurs qui mettent en commun leurs données collectées, soit en alimentant une base de données partagée, soit en communicant entre eux. Leur répartition échantillonne une zone de l'espace et influence le traitement ultérieur.

Il peut alors être intéressant de représenter le réseau de capteurs comme un système réparti. En effet, du point de vue de la mesure, l'avantage est de considérer chaque capteur comme un système autonome qui communique son point de vue aux autres. Cela peut

par exemple permettre de détecter des capteurs défectueux (Saukh et *al.*, 2014) ou de considérer des réseaux de capteurs de types différents (réseaux hétérogènes). D'un point de vue statistique, cela suggère de ne pas traiter tous les capteurs comme des variables identiquement distribuées (Predd et *al.*, 2006; Radosavljevic et *al.*, 2010).

Les méthodes d'ensembles reposent sur la même idée. Elles permettent de construire un nouveau modèle à partir d'un ensemble de modèles et tirer partie au maximum d'un modèle là où il est le plus certain (Dietterich, 2000). Une nouvelle technique d'apprentissage automatique qui vise à intégrer davantage cette notion de subjectivité de la perception, l'apprentissage fédéré, est actuellement développée par Google (McMahan et *al.*, 2016). L'idée est que le modèle de chaque utilisateur est composé d'un modèle générique et de paramètres confidentiels, et seulement les modèles génériques sont partagés pour en construire un meilleur. Il en résulte un nombre de communications plus faible pour un modèle de qualité équivalente.

De plus, la mobilité d'un système de mesure peut être intéressante pour couvrir une zone de l'espace plus grande qu'avec un réseau de capteurs statiques et éventuellement échantillonner plus finement le phénomène spatial. Les systèmes répartis étudient la mobilité notamment au travers du déplacement d'une flotte de robots dans l'espace.

Par exemple, un objectif récurrent est leur rencontre au bout d'un certain temps (Kranakis et *al.*, 2006). Cela s'appelle un *Rendez-Vous*. Cette notion permet d'ordonner partiellement les évènements rencontrés par les robots (« happened-before »).

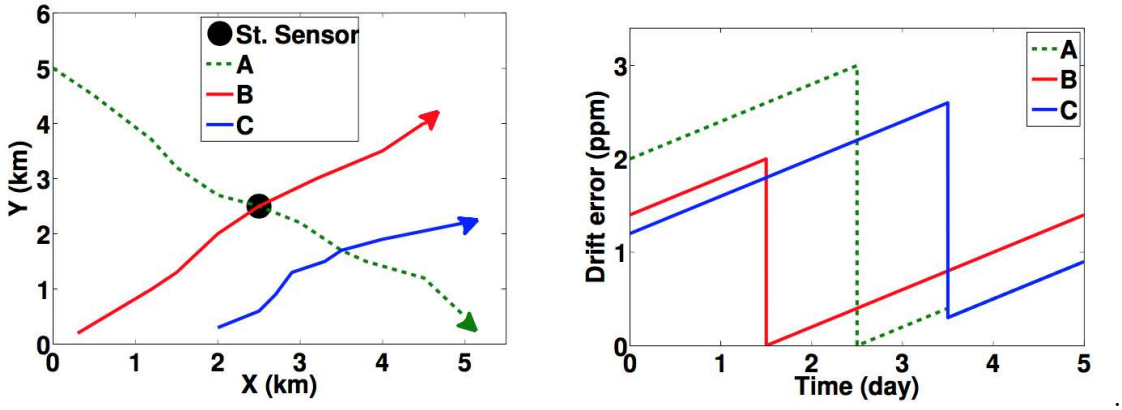
Les types de capteurs embarqués sur les systèmes mobiles étant souvent sujets à un biais de mesure, cette notion de *Rendez-Vous* appliquée à la mesure permet d'imaginer des algorithmes d'étalonnage collaboratif, pour passer d'observations subjectives à des observations objectives, ou d'étalonnage lors du « *Rendez-Vous* » avec un étalon (étalonnage indirect).

En reprenant les illustrations de Xiang et *al.* (2012) pour formaliser ce problème (cf. Figure 1.2), considérons un exemple de trois capteurs mobiles qui souffrent d'un biais de mesure qui s'aggrave avec le temps (une dérive) et d'une station fixe de référence. Les mouvements des capteurs considérés sont tracés Figure 1.2a. Leurs erreurs au cours du temps sont tracées Figure 1.2b.

Bien que les capteurs mobiles souffrent d'une erreur de mesure incompressible, leur rencontre assure qu'ils captent le même signal et permet dans le meilleur des cas de réduire l'erreur de mesure à 0. Lorsque le capteur rouge puis le capteur vert rencontrent la station de référence, au mieux leurs mesures sont parfaitement étalonnées ensuite et l'erreur de mesure est nulle. Lorsque le capteur vert et le capteur bleu se rencontrent, ils peuvent collaborer et espérer obtenir un étalonnage meilleur que leur étalonnage individuel. Leur erreur de dérive est donc au mieux inférieure à l'erreur minimale pour les deux systèmes. Ces algorithmes peuvent être appliqués directement en temps réel pour les systèmes de mesure quelque soit la taille du réseau de capteurs (Hasenfratz et *al.*, 2012; Xiang et *al.*, 2012).

En pratique, deux systèmes se rencontrent s'ils sont dans une même zone de l'espace au même moment. Il nous faut alors définir ce qu'est la distance acceptable pour que deux systèmes soient dans la même zone de l'espace et la durée d'un moment. Il y a donc un compromis à trouver pour respecter l'hypothèse selon laquelle les systèmes mesurent le

même signal et pour maximiser la probabilité que les systèmes soient en *Rendez-Vous*. Nous reviendrons sur cette notion au chapitre 4.



(a) Traces du mouvement des capteurs et de leurs *Rendez-Vous*.

(b) Erreurs de dérive au cours du temps pour trois capteurs mobiles.

FIGURE 1.2 – Illustration du concept d'étalonnage durant un *Rendez-Vous*— d'après Xiang et *al.* (2012).

1.3.3 Création de connaissances

A l'heure actuelle, les processus de création de connaissances agnostiques[2] sont un enjeu pour concevoir des robots humanoïdes et plus généralement des systèmes qui raisonnent comme l'homme (Robertsson et *al.*, 2007) sur la base de modèles réalistes.

Pour illustrer l'un de ces processus, nous choisissons la formalisation proposée par Gregory Piatetsky-Shapiro (cf. Figure 1.3) sous l'appellation *Knowledge discovery in data bases* (KDD) (Piatetsky-Shapiro, 1991; Fayyad et *al.*, 1996) car elle est simple, concrète et suit un raisonnement dialectique des données à la connaissance.

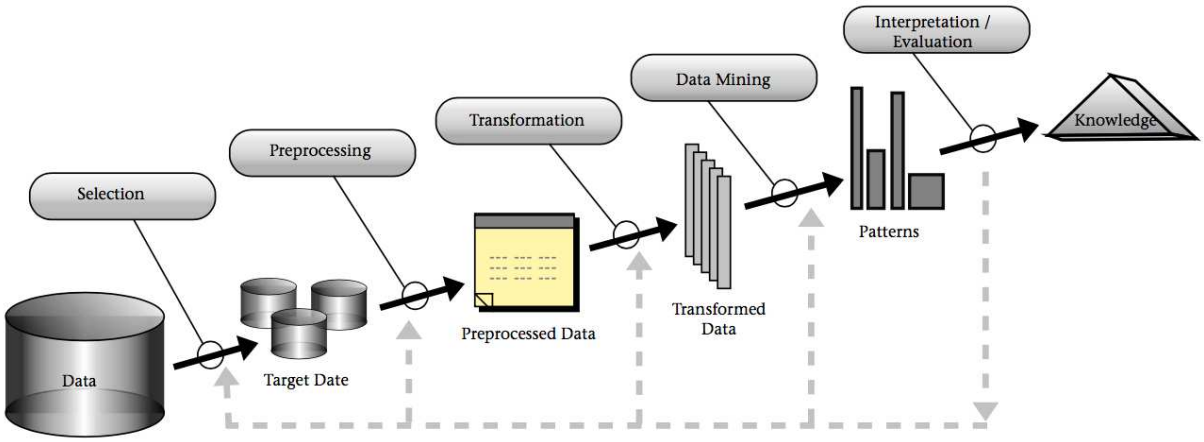


FIGURE 1.3 – Étapes du processus KDD (Fayyad, 1996).

2. A comprendre, « sans a priori ».

Dans ce processus, les données doivent représenter les informations qui sont utiles à la déduction d'une connaissance destinée à un utilisateur. Une fois le jeu de données collecté, il faut se focaliser sur un sous-ensemble du jeu de données initial, le pré-traiter (réduire le bruit de mesure, gérer les valeurs manquantes, ...), puis déterminer de nouvelles représentations des données qui permettent de détecter des propriétés intéressantes. Enfin, il faut confirmer ou infirmer ces propriétés à l'aide des données mises de côté. A chaque étape, il est possible de retourner en arrière une fois le jeu de données mieux compris.

Nous pourrions imaginer d'autres processus. Par exemple, si les données sont produites par des capteurs, nous pouvons ajouter une boucle de rétroaction de la connaissance aux données. Cela pourrait traduire la capacité d'action du système sur son environnement et sur la mesure du signal – et donc sur les données analysées par la suite du processus.

Dans cette logique, nous avons présenté un poster (cf. annexe 3) exposant la démarche que nous avons suivie au cours de cette thèse à ASPLOS2019 [3]. Il met en exergue la proximité entre le cycle de vie d'un projet en science des données et le cycle de prototypage d'un instrument de mesure.

1.4 Problématique de la qualité de l'air en zone urbaine

La pollution de l'air est une modification des caractéristiques naturelles de l'atmosphère, par contamination chimique, physique ou biologique, néfaste pour l'écosystème. De nombreux facteurs influencent la pollution de l'air : le rayonnement solaire et la température accélèrent certaines réactions, au point que des cycles journaliers et saisonniers se distinguent ; le vent transporte et disperse les polluants, ou à l'inverse son absence, favorise le dépôt et l'homogénéisation dans le temps ; la pluie lessive l'air et transfère la pollution au sol et à l'eau... Et, en ville, ce phénomène se complexifie davantage du fait des nombreuses sources d'émission et de la structure de la ville induite par les bâtiments.

Son impact sur la santé humaine (asthme, cancer, bronchite...) (Brunekreef et Holgate, 2002) et l'environnement (pluies acides, destruction de la couche d'ozone, effet de serre...) est préoccupant. En effet, en avril 2018, le Commissariat général au développement durable, une direction du ministère de la Transition écologique et solidaire, publie son rapport sur les « Modes de vie et pratiques environnementales des Français » [4] : la pollution de l'air et le réchauffement climatique sont les deux premières préoccupations environnementales des français. Les « Marches pour le climat » depuis septembre 2018 confirment ces estimations. De plus, l'Organisation Mondiale de la Santé alarme sur le sujet depuis plus de 40 ans (Mage et *al.*, 1996). D'après les derniers rapports de l'organisation mondiale de la santé (OMS), la pollution de l'air tue environ 7 millions de personnes par an à travers le monde, dont 4,2 en raison de la pollution de l'air extérieur, et 90 % des personnes respirent un air chargé en polluants gazeux et particulaires.

3. The 24th ACM International Conference on Architectural Support for Programming Languages and Operating Systems.

4. `https://www.statistiques.developpement-durable.gouv.fr/sites/default/files/2018-10/thema-03-modes-vie-pratiques-environnementales-francais-b.pdf`

Dans certaines régions du monde, la population est prête à payer pour se procurer de l'air moins pollué. Fondée en 2014, l'entreprise canadienne Vitality Air [5] vend de l'air en bouteille, provenant des Rocheuses canadiennes, entre 3,5 et 2300 dollars le litre (cf. Figure 1.4). Elle a commencé par les vendre à Beijing puis s'est élargie au reste du monde, et propose désormais une nouvelle version « optimisée » (sic), des bouteilles d'oxygène pur à 95 %. Plusieurs types de bouteilles existent : parfumées à la bière, énergisantes, anti-stress, avec de la poudre de diamants, signés 2 Chainz (un rappeur et acteur américain)...

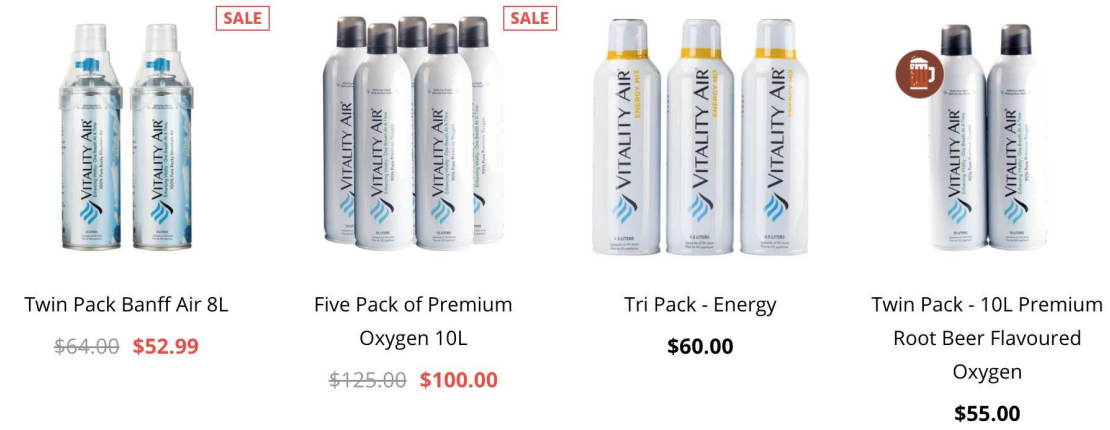


FIGURE 1.4 – Bouteilles d'air et d'oxygène vendues par Vitality Air.

La Figure 1.5 est une photographie de la situation à Hong Kong, en Chine, où une impression du panorama visible dans des conditions idéales est installée pour permettre aux touristes de se prendre en photo.

FIGURE 1.5 – Cette photographie, prise par Alex Hofford, met en exergue le contraste entre la réalité du panorama de Hong Kong (en arrière-plan) et l'idéal installé en trompe-l'œil pour les touristes (au premier plan).

5. https://vitalityair.com/

1.4.1 Législation relative à la surveillance des polluants atmosphériques

Depuis 1987, l'OMS publie des lignes directives concernant la qualité de l'air[6]. *« Ces lignes directrices visent à informer les responsables de l'élaboration des politiques et à fournir des cibles appropriées à toute une série d'actions à mener pour la prévention de la pollution atmosphérique dans les différentes parties du monde. Elles constituent l'évaluation la plus largement reconnue et la plus actuelle des effets de la pollution aérienne sur la santé. Elles préconisent des objectifs de qualité de l'air qui réduisent fortement les risques sanitaires. ».*

Les nouvelles directives (2005) ont vocation à s'appliquer au monde entier. Ces directives proposent un ensemble de valeurs cibles pour les principaux polluants atmosphériques, au-dessus desquels des effets néfastes pour la santé humaine sont clairement identifiés. Néanmoins, les études épidémiologiques montrent qu'en dessous de ces seuils les effets pourraient rester non négligeables.

En France, la loi LAURE de 1996 reconnaît le droit de respirer un air sain. Elle marque le début du cadre réglementaire français en terme de pollution de l'air. Cette loi s'accompagne d'un dispositif de surveillance de la qualité de l'air. Les Associations Agréées Surveillance Qualité de l'Air (AASQA) sont des associations agréées par l'État. Elles se répartissent la surveillance et la prévention en terme de pollution de l'air à l'échelle régionale Elles sont réunies au sein de la Fédération ATMO FRANCE et coordonnées techniquement et scientifiquement par le laboratoire central de surveillance de la qualité de l'air (LCSQA).

Les normes de gestion de qualité de l'air sont, elles, définies à l'échelle européenne. Ces normes, à titre indicatif pour certains polluants et obligatoires pour d'autres, sont retranscrites en France dans le Code de l'environnement. Elles régulent les émissions par pays (directive (EU) 2016/2284 en vigueur) et cadrent la surveillance de l'air ambiant, en indiquant le nombre de capteurs et leurs incertitudes attendues, leurs emplacements et les méthodes de référence (directives 2004/107 et 2008/50/CE en vigueur, révision en cours). Cette surveillance permet de contrôler l'écart aux seuils définis. Les seuils sont de plusieurs types :

- **le seuil d'alerte** est le seuil au-dessus duquel il y a un risque pour la santé humaine à court terme,
- **la valeur limite** est le seuil au-dessus duquel il y a un risque pour la santé humaine à long terme,
- **le niveau critique** est le seuil au-dessus duquel il y a un risque pour l'écosystème,
- **la valeur cible** est la valeur de mesure à atteindre à court terme,
- **l'objectif à long terme** est la valeur de mesure à atteindre à long terme,
- **le seuil d'évaluation** inférieur et supérieur sont des pourcentages de la valeur limite au-dessus desquels les contraintes de collecte se durcissent et l'estimation de la qualité de l'air par modélisation ne suffit plus.

La Table 1.1 (source AirParif) présente les valeurs limites et les seuils d'alerte pour les polluants qui nous intéressent dans ce travail, le dioxyde d'azote (NO_2), le monoxyde de

6. voir le site de l'association Respire `https://www.respire-asso.org/`

carbone (CO), et les particules en suspension (PM_{10}). Ces trois polluants sont sélectionnés pour 2 raisons. Premièrement, ces polluants sont des traceurs de l'émission par le trafic routier. La majeure partie des NO_x (NO et NO_2) sont formés lors de la combustion à haute température en présence d'air (moteur thermique et chauffage, cimenterie). Les PM_{10} sont également largement émises par le trafic routier, notamment via la formation de suie (moteur diesel), l'usure des véhicules (carrosserie, plaquettes de frein...) et la remise en suspension des poussières sur la chaussée. Enfin, le CO est également émis lorsque la combustion ne s'effectue pas en condition stœchiométrique et dépend de la richesse du mélange (rapport entre le nombre de moles d'hydrocarbure et d'oxygène), majoritairement par les véhicules essence. Deuxièmement, ces composés peuvent être mesurés à partir de capteurs portatifs simples et à bas coût (cf. chapitres suivants).

En 2018, ATMO Occitanie (AASQA région Occitanie) indique que pour la métropole de Toulouse, les niveaux relevés en PM_{10} respectent la valeur limite de $40\,\mu g/m^3$. Les valeurs maximales sont enregistrées sur la station en proximité du périphérique avec une moyenne annuelle de $28\,\mu g/m^3$. Le niveau de fond urbain (moyenne en dehors des zones intenses d'émission) est environ de $15\,\mu g/m^3$. La valeur moyenne de NO_2 pour l'agglomération de Toulouse se situe à $17\,\mu g/m^3$, en deçà de la valeur limite. Cependant, la station en bordure de périphérique affiche une moyenne annuelle de $68\,\mu g/m^3$. De même la station située en proximité de l'autoroute A620 dépasse la valeur limite avec une moyenne annuelle de $47\,\mu g/m^3$.

Le cas du CO est un peu particulier. Il existe une valeur de fond en CO liée à la formation de ce composé dans l'atmosphère lors de son cycle naturel. Les concentrations atmosphériques de ce gaz sont assez faibles du fait de son temps de résidence (cf. ci-dessous) relativement long. Cette valeur de fond est située entre 0,05 et 0,15 ppm (rapport de mélange exprimé en partie par millions). En raison de la très haute toxicité de ce composé, les émissions par les véhicules à moteur ont très largement diminué en zone urbaine. Le niveau de CO en ville se situe entre 0,5 et 2 ppm (environ 1 ppm pour l'agglomération de Toulouse). Les concentrations les plus élevées sont malheureusement rencontrées en air intérieur (tabagie, manque de ventilation des locaux, appareil de combustion défectueux...), pouvant mener à des situations dramatiques. Très localement, le niveau de CO peut augmenter à cause de la congestion du trafic. La valeur limite pour le CO donnée dans la Table 1.1 est exceptionnelle en milieu ouvert. La correspondance entre ppm et $\mu g/m^3$ est donnée dans l'annexe 4.

Même si les concentrations moyennes respectent les valeurs limites, certaines stations du fait de leur proximité avec les sources de pollution vont afficher des valeurs dépassant ces limites. Il existe donc une hétérogénéité importante des concentrations au niveau d'une agglomération en fonction de la distribution des sources.

De plus, les polluants étudiés sont des composés réactifs à courte durée de vie. Les dépôts via les précipitations (voie humide) et par sédimentation et interaction avec les surfaces (voie sèche) font décroître les concentrations. Ces dernières évoluent également en fonction des réactions chimiques affectant les composés (destruction chimique).

L'ensemble de ces processus peut être représenté par une équation cinétique de premier ordre (Équation 1.1).

TABLE 1.1 – Valeurs limites et seuils d'alerte définis par l'Union Européenne pour le dioxyde d'azote, le monoxyde de carbone, et les particules PM_{10}.

Polluant	Valeur limite	Seuil d'alerte
NO_2	$40\,\mu g/m^3$ en moyenne annuelle	$400\,\mu g/m^3$ en moyenne horaire pendant 3h consécutives
	$200\,\mu g/m^3$ en moyenne horaire pas plus de 18 h par an	
PM_{10}	$40\,\mu g/m^3$ en moyenne annuelle	$80\,\mu g/m^3$ en moyenne journalière
	$50\,\mu g/m^3$ en moyenne journalière pas plus de 35 jours pas an	
CO	$10\,000\,\mu g/m^3$ maximum journalier de la moyenne sur 8 h	

$$\frac{d\rho_X}{dt} = -k\rho_X \tag{1.1}$$

La durée de vie fait référence au temps de résidence atmosphérique des polluants $\tau = 1/k$.

Certains composés, comme les oxydes d'azote, étant très réactifs, leur concentration va rapidement décroître en s'éloignant de la source d'émission. La Figure 1.6 représente sur une échelle logarithmique le temps de résidence de quelques composés chimiques dans la basse atmosphère [7]. Le CO_2 avec un temps de résidence de l'ordre du siècle n'est pas représenté sur cette échelle. Les NO_x ont un temps de résidence inférieur à la journée. Ce court temps de résidence va donc fortement limiter leur dispersion atmosphérique et donc créer de fortes hétérogénéités à proximité des sources.

Les aérosols (particules) ont un temps de résidence de quelques jours à quelques semaines. Ce temps de résidence varie en fonction des types de sources et de l'altitude à laquelle se situent les particules. Par exemple, les poussières minérales d'origine éolienne émises depuis les zones d'Afrique du Nord ont une durée de vie suffisamment longue pour atteindre les villes d'Europe ou des États-Unis en raison de leur transport en altitude. D'autres phénomènes, comme la pollution d'origine agricole, contribuent également à la charge en particules dans les agglomérations. Les concentrations de particules sont donc susceptibles d'avoir une homogénéité plus importante que les NO_x.

7. http://cerea.enpc.fr/fich/support_cours/POLU1_2012-2013/VET-Ozone-NO2-2013.pdf

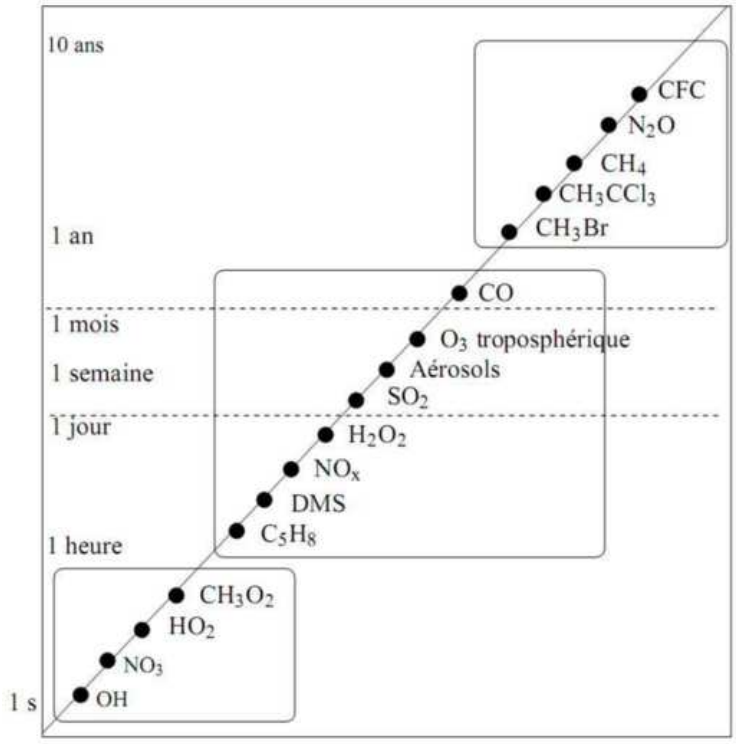


FIGURE 1.6 – Temps de résidence atmosphérique de différents composés chimiques. L'échelle des temps est logarithmique.

1.4.2 Émissions de polluants atmosphériques en zone urbaine

La pollution atmosphérique résulte de l'injection et de l'accumulation dans l'air de différents composés chimiques. Les processus d'émission dits primaires sont ceux qui émettent directement dans l'atmosphère les différents composés. Les processus d'émission dits secondaires sont ceux qui émettent un ensemble de précurseurs qui, par transformation chimique, vont former les polluants.

Les oxydes d'azote (NO_x) sont typiquement des polluants primaires formés lors de la combustion, et en particulier par les moteurs des véhicules. L'ozone, qui se forme par réactions photochimiques à partir des oxydes d'azote et des composés organiques volatils (COV) est un polluant secondaire, qui apparaît lors des journées ensoleillées.

Les particules en suspension dans l'atmosphère sont émises par voie primaire et secondaire. Les PM_{10} (PM pour particulate matter) correspondent à la fraction massique des particules en suspension ayant un diamètre aérodynamique inférieur à $10\,\mu$m. Dans cette fraction massique, nous retrouvons les particules grossières émises par mécanismes d'érosion et d'abrasion ainsi que certaines particules biogéniques comme les pollens. La fraction massique $PM_{2.5}$, qui est incluse dans la fraction PM_{10}, correspond à la masse de particules ayant un diamètre aérodynamique inférieur à $2{,}5\,\mu$m. Dans les $PM_{2.5}$ nous retrouvons des composés primaires comme les suies et des composés secondaires issus de la condensation de précurseurs gazeux.

Les inventaires d'émission regroupent l'ensemble des émissions de polluants pour un territoire et une période donnée. Lorsque l'inventaire est spatialisé, la terminologie change

pour « cadastre des émissions ». L'inventaire est essentiellement construit à partir de données économiques de consommation et d'évolution technologique (procédés de combustion, industrie). La mise à jour des inventaires dépend donc de la disponibilité de ces données. Les sources peuvent être ponctuelles, linéiques ou zonales (diffuses) et d'origine anthropique ou naturelle. L'intensité de l'émission varie suivant le processus en jeu. Typiquement, nous retrouvons un cycle journalier d'émission par le trafic routier, avec une augmentation des émissions pendant les déplacements domicile–travail. A ce cycle journalier se superpose un cycle hebdomadaire lié à la réduction du trafic pendant les week-end.

La nomenclature Corinair, utilisée dans l'Union Européenne, classe les sources de pollution en 11 catégories : trafic routier, plateforme aéroportuaire, trafic ferroviaire et fluvial, résidentiel et tertiaire, industrie manufacturière, chantiers et carrières, extraction, transformation et distribution d'énergie, traitement des déchets, agriculture et enfin émission naturelles. Ce genre de nomenclature est utile pour normaliser ces informations dans les cadastres et les inventaires d'émission. Elles identifient les sources et taux d'émission. Le pas spatio-temporel dépend de la vocation de l'étude (échelle locale, transfrontalière ou internationale) et de la variabilité du polluant. Généralement, la taille des mailles varient de 5 km× 5 km à 100 km× 100 km. Le pas de temps est le jour ou l'année.

Le graphique 1.7 présente les émissions des principaux polluants par secteur d'activité pour l'agglomération de Toulouse Métropole. Comme pour toutes les grandes métropoles, le secteur transport est le principale contributeur aux émissions de NO_x et de particules. Ces émissions proviennent de l'utilisation de véhicules à propulsion thermique. Le secteur résidentiel émet pour 24 % des PM_{10} et 7 % des oxydes d'azote. Ce secteur inclut le chauffage résidentiel basé sur la combustion de fuel fossile et de biomasse. Enfin le secteur industriel contribue également pour 23 % au PM_{10} à travers l'émission de poussières dans les processus de fabrication. Ce secteur est est partie responsable de l'émission de composés organiques volatils (utilisation de solvants par exemple). Ces composés sont également susceptibles de contribuer à la formation de particules fines.

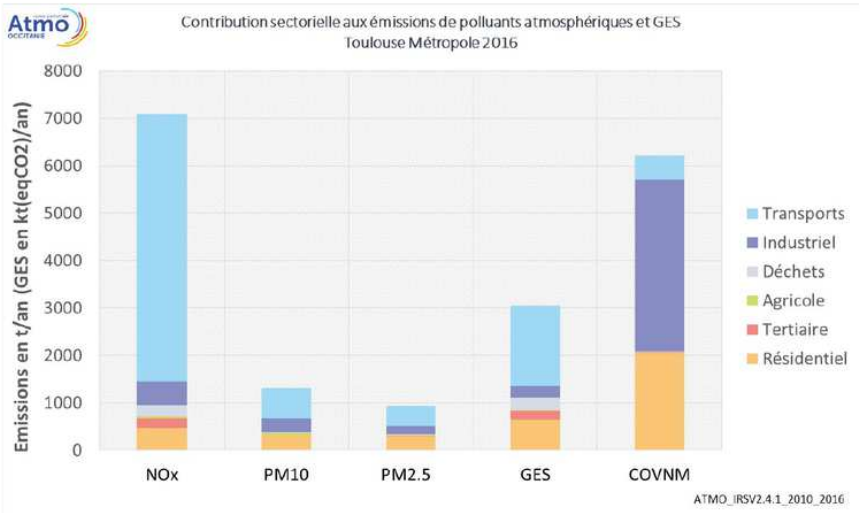


FIGURE 1.7 – Contribution sectorielle aux émissions d'oxydes d'azote (NO_x), de particules PM_{10} et $PM_{2.5}$, de gaz à effet de serre (GES) et de composés organiques volatils non-méthaniques (COVNM) pour Toulouse Métropole en 2016 (source ATMO Occitanie).

1.4.3 Représentations des concentrations à l'échelle de la ville

A partir du cadastre des émissions, il est alors possible d'obtenir une estimation des concentrations atmosphériques en prenant en compte le phénomène de dispersion, de transformation chimique et de dépôts. Dans l'hypothèse de dilution, nous séparons l'équation d'évolution des champs descriptifs de l'écoulement atmosphérique (vitesse du vent, densité, température, humidité) de celle des espèces chimiques. L'évolution de la concentration c_i d'une espèce chimique est alors donnée par une équation de dispersion réactive, de type advection-diffusion-réaction. L'advection correspond au transport dans le champ du vent V, la diffusion au mélange turbulent et la réaction aux modifications physico-chimiques (Sportisse, 2008).

Cette équation prend la forme d'une équation aux dérivées partielles (Équation 1.2) :

$$\frac{\partial c_i}{\partial t} + div(V(x,t)c_i) = div(K_{molec}\nabla c_i) + \chi_i(c_i T(x,t),t) + S_i(x,t) - \Lambda_i c_i \tag{1.2}$$

.

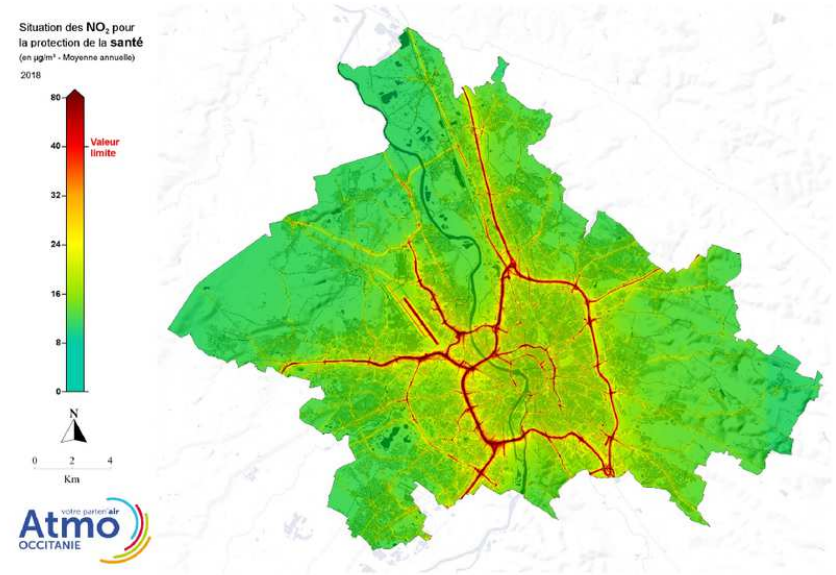
Nous retrouvons les termes liés à l'advection $div(V(x,t)c_i)$, à la diffusion moléculaire $div(K_{molec}\nabla c_i)$, aux paramétrisations physico-chimiques $\chi_i(c_i T(x,t),t) - \Lambda_i c_i$, et aux termes sources $S_i(x,t)$.

Les modèles tridimensionnels qui résolvent cette équation de manière numérique sont généralement appelés *modèles de chimie-transport* (Chemistry–Transport model en anglais, CTM). Les CTM modélisent de manière déterministe les concentrations et nécessitent pour ce faire un grand nombre de variables d'entrée concernant les données météorologiques et les sources de polluants. Il existe une grande diversité de CTM.

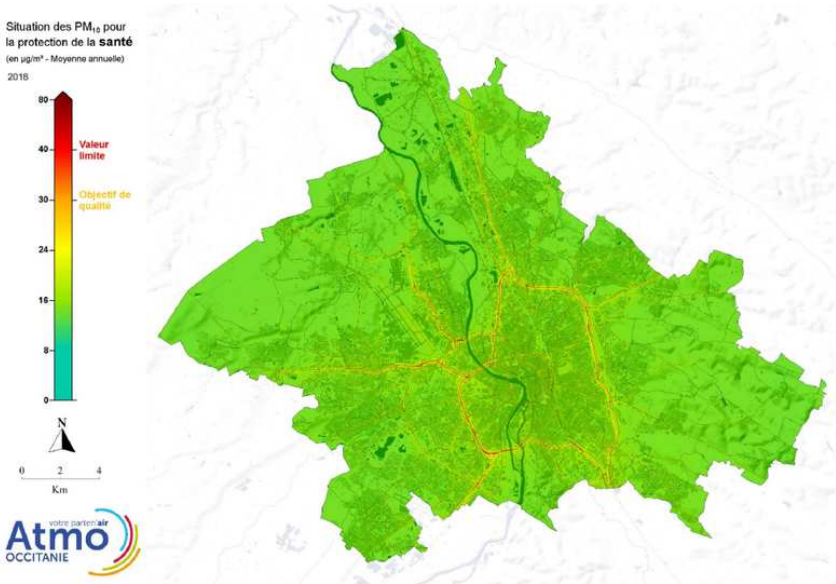
Le modèle national CHIMERE [8] (Menut et *al.*, 2013) permet d'effectuer des simulations à l'échelle régionale (quelques milliers de km) avec une résolution spatiale de quelques km. En dessous de ces échelles, les processus turbulents doivent être explicités, rendant le coût numérique de calcul beaucoup plus important. La simulation numérique à l'échelle urbaine fait appel à d'autre types de modèles spécifiquement adaptés pour représenter les sources urbaines et leur dispersion (e.g. ADMS–urban, Sirane...).

La Figure 1.8 est une représentation graphique (heatmap) de la répartition spatiale des polluants (le NO_2 et les PM_{10} en moyenne annuelle pour 2018) estimée par une simulation numérique pour la ville de Toulouse. La structure du réseau routier se distingue nettement à travers les concentrations de NO_2. Les grands axes de circulation comme le périphérique sont nettement marqués. Le contraste entre la pollution de fond (hors réseau routier) et le réseau routier pour les PM_{10} semble moindre que pour le NO_2, reflétant un temps de résidence plus important permettant une imprégnation plus globale de l'agglomération.

8. `https://www.lmd.polytechnique.fr/chimere/`



Cartographie des concentrations moyennes annuelles de NO_2 sur le territoire de Toulouse Métropole, 2018



Cartographie des concentrations moyennes annuelles de PM10 sur le territoire de Toulouse Métropole, 2018

FIGURE 1.8 – Simulations des concentrations de NO_2 et PM_{10} en moyenne annuelle pour l'agglomération de Toulouse en 2018 — d'après ATMO Occitanie.

Ces heatmaps donnent une vision idéalisée de la pollution atmosphérique pour une période donnée. Elle repose sur des échelles spatiales et temporelles de la pollution déterminées par des équations théoriques. La Figure 1.9 présente ces échelles. Différents mécanismes interviennent dans la construction de la pollution urbaine. La concentration de fond de l'échelle régionale (concentration liée aux conditions synoptiques[9]) est distinguée de la concentration de fond urbaine, qui peut être fortement dépendante de conditions météorologiques locales, comme les inversions thermiques. Sur ces concentrations de fond vient s'ajouter une pollution de proximité générant des points chauds (hot spot) de pollution urbaine. Ces points chauds génèrent une hétérogénéité importante dans

9. À grande échelle, relativement au cadre d'étude.

la ville. Celle-ci est renforcé par la dynamique atmosphérique interne de la ville qui peut s'avérer complexe en fonction de la configuration urbaine, comme par exemple dans le cas de rues encaissées (effet canyon).

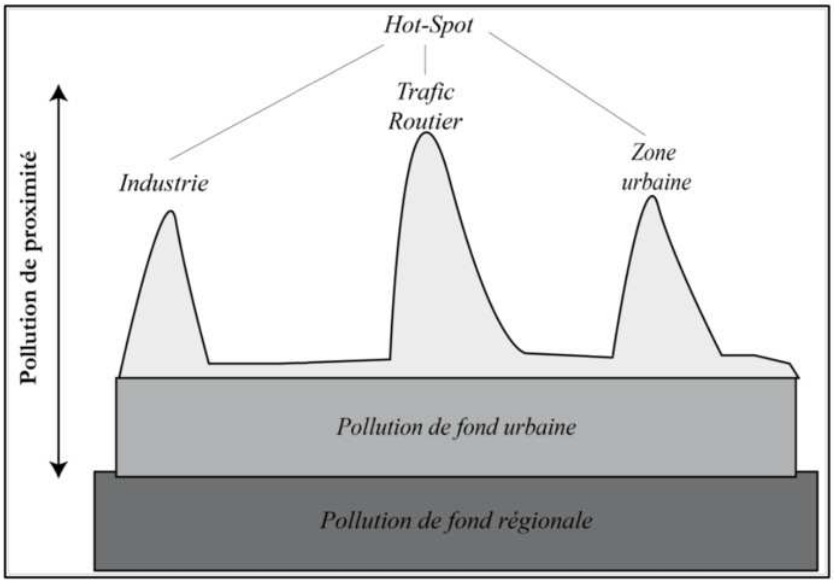


FIGURE 1.9 – Échelles des niveaux de pollution selon les environnements caractéristiques des sources d'émission — d'après Roussel (2006).

La confrontation de ces simulations avec les observations *in situ* des concentrations des polluants permet d'évaluer l'exactitude de ces estimations. Bien que le déploiement de stations de mesure soit un sujet d'étude réccurent dans la littérature, les stations fixes de mesures réellement déployées par ATMO sont principalement reparties de manière à échantillonner les sources d'émission, selon l'environnement caractéristique pris en compte par la simulation. La pertinence des observations locales dans le processus de généralisation et de spatialisation peut donc être questionnée.

Rodriguez et *al.* (2019) étudient la représentativité spatiale locale de ce genre de stations fixes pour le NO_2 et les PM_{10} au regard de l'homogénéité (concentrations quotidiennes ne différant pas de plus de 20 % de la valeur moyenne journalière) et de la sensibilité (forte corrélation des concentrations). Ils en concluent que la localité de la sensibilité d'une station est plus étendue que celle de l'homogenéité, notamment proche des zones de trafic. Les zones représentatives fluctuent grandement au cours de la journée, même pour un même type de station. De plus, la forme et l'entendue des zones représentatives varient également. Les stations de trafic ne sont pas représentatives de la globalité du quartier, mais plutôt de quelques m^2 le long des routes, à la différence des stations de fond urbain, qui peuvent être représentatives jusqu'au km^2.

La Figure 1.10 présente la distribution des stations de mesure pour l'agglomération de la ville de Toulouse. Il existe 4 stations de mesure de fond urbain, 3 en zone de trafic routier et 5 en zone industrielle. Nous remarquons que les stations de mesure en zone de trafic routier sont toutes réparties sur le périphérique. Quand nous comparons la répartition des stations avec la heatmap de NO_2, nous remarquons que les axes routiers secondaires ne sont pas échantillonnés. Les zones résidentielles sont également assez mal représentées.

Les axes de croisement montrent également des zones assez hétérogènes, probablement en relation avec la circulation atmosphérique locale au niveau des intersections (Dobre et *al.*, 2005). Nous comprenons également que la statistique des mesures pour une classe donnée risque de ne pas être représentative en raison du nombre limité de points d'échantillonnage. Le réseau de stations fixes pose donc une question d'ordre méthodologique concernant la représentativité spatiale et temporelle des mesures à des échelles pertinentes pour la ville, à savoir de l'ordre de la centaine de mètres et de l'heure.

Pour répondre à cette question, il existe 2 solutions. La première est de multiplier les points de mesure fixe. La seconde, qui est retenue dans le cadre de cette thèse, est d'utiliser un réseau mobile d'observations.

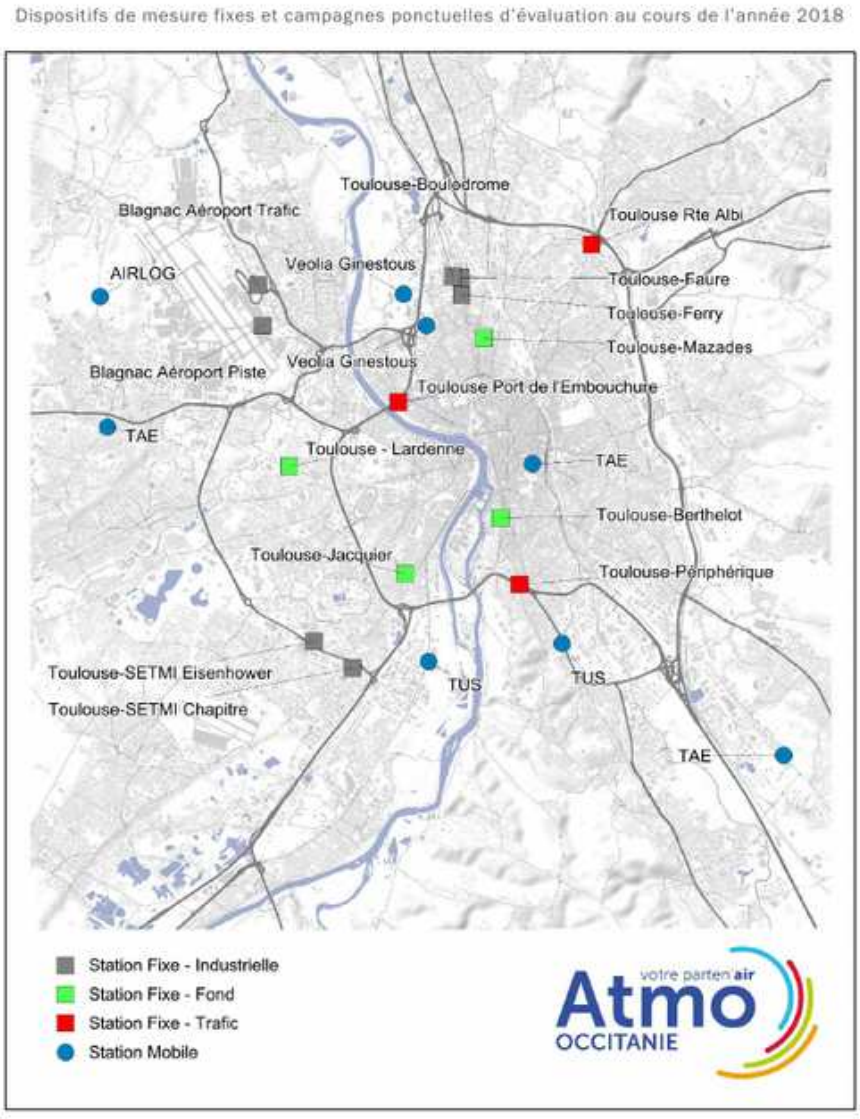


FIGURE 1.10 – Dispositifs de mesure fixes et campagnes ponctuelles d'évaluation au cours de l'année 2018 sur l'agglomération de Toulouse — d'après ATMO Occitanie.

1.5 Vers de nouvelles observations en réseau

Yi et *al.* (2015) distinguent trois types de réseaux de capteurs sans fil (Wireless Sensor Network, WSN) : les réseaux de capteurs statiques (Static Sensor Network, SSN) lorsque les capteurs sont fixes, les réseaux de capteurs participatifs (Community Sensor Network, CSN) lorsque les capteurs sont transportés par des volontaires et les réseaux de capteurs sur véhicule (Vehicle Sensor Network, VSN) lorsque les capteurs sont dédiés à un type de véhicule.

Les SSN ont l'avantage d'être plus simples à mettre en œuvre. En effet, la position du capteur est déterminée à l'avance ce qui permet de s'épargner l'acquisition de la position du système et d'assurer la connectivité du réseau (les données peuvent être relevées a posteriori). De plus, il n'y a généralement pas de contrainte d'alimentation, de poids et de taille du système. Cette simplicité de mise en œuvre permet de disposer de plusieurs capteurs à un même endroit et d'obtenir des données précises et fiables. Néanmoins, le placement des capteurs est crucial. En outre, la densité nécessaire de couverture en capteurs dépend du phénomène étudié. La ville est le siège de nombreuses interactions humaines qui induisent de fortes variabilités en pollution et donc nécessite de nombreux capteurs statiques. Cependant, ces interactions et notamment la mobilité humaine peuvent permettre de pallier cet inconvénient.

Dans une approche classique de mesure de la pollution, les stations de mesure sont fixes (SSN) et ne décrivent que quelques points isolés de l'espace. Le pas de temps de la mesure est sélectionné de manière à couvrir le cycle journalier des différents polluants, qui est modulé par le cycle des émissions et par l'influence de la météorologie et de la photochimie.

1.5.1 Capteurs

Les instruments de mesure sont généralement des instruments dits « en ligne », permettant une mesure quasiment instantanée, à l'inverse des instruments de mesure dits « passifs » à lecture différée (généralement en laboratoire) permettant de connaître uniquement la concentration cumulée sur une période donnée (e.g. canister d'air ou filtre à particules). Nous faisons référence ici uniquement aux techniques en ligne.

La qualité d'un capteur s'estime au travers des caractéristiques suivantes (Menini, 2011) :

- La sensibilité : exprime la variation de la réponse du capteur en fonction de la variation du mesurande (grandeur mesurée par le capteur),
- La sélectivité : exprime l'influence d'autres évènements sur la variation de la réponse du capteur,
- La stabilité : exprime la stationnarité de la ligne de base (i.e., la valeur du capteur en conditions normales et constantes dans le temps),
- La réversibilité : exprime la capacité du capteur à revenir à son état initial après excitation,
- Le temps de réponse : quantifie le temps que met le capteur à réagir à un évènement,
- Le temps de recouvrement : quantifie le temps que met le capteur pour revenir dans la configuration initiale une fois l'évènement disparu,

— La reproductibilité : exprime la capacité du capteur à produire la même réponse pour un même évènement.

Un instrument de mesure, ou une source d'information, désigne un système composé d'un ou plusieurs capteurs et d'un moyen d'afficher, stocker ou communiquer l'information, ainsi que la capacité de faire quelques calculs simples.

Néanmoins, l'émergence de capteurs détecteurs miniatures à bas coût questionne l'approche traditionnelle (Mead et *al.*, 2013; Moltchanov et *al.*, 2015). Ils permettent d'être embarqués sur une plateforme mobile et ainsi couvrir plus d'espace (Kumar et *al.*, 2015).

Un capteur portatif ou micro-capteur est un capteur de petite taille, peu gourmand en énergie. Dans le cas de la pollution atmosphérique, ces capteurs ont la particularité d'être peu sélectifs et de dériver dans le temps, c'est-à-dire que leur ligne de base varie dans le temps. Il est alors nécessaire d'étalonner régulièrement ce type de capteur.

1.5.2 Réseaux participatifs

Le crowdsourcing (néologisme anglais traduit en « fourni par la foule ») désigne le transfert d'un processus de travail vers une main d'œuvre extérieure constituée d'un grand nombre d'intervenants. Par exemple, le crowdfunding est un moyen de financement alternatif qui fait appel à des ressources financières auprès des internautes afin de financer un projet.

En science, le crowdsensing est le transfert de la collecte d'informations. L'idée d'impliquer des individus dans la collecte d'informations pour des études scientifiques n'est pas récente, notamment en écologie (Dickinson et *al.*, 2010). Ce concept repose sur la participation volontaire d'individus ne possédant pas d'expertise particulière du sujet à une démarche scientifique. La participation peut être active ou passive, i.e. sans intervention des individus.

Le développement récent et massif des moyens de communication individuels à largement contribué au développement du crowdsensing et des CSN en général. Parallèlement, le développement de capteurs connectés permet la construction de vastes réseaux. Un des vecteurs les plus efficaces de déploiement du réseau est la population. C'est par exemple le cas de l'utilisation des stations météorologiques individuelles dans les prévisions météorologiques (Meier et *al.*, 2017). La Figure 1.11 issue de Muller et *al.* (2015), présente un diagramme de Venn recoupant les éléments essentiels de cette approche de la donnée scientifique en réseau : les citoyens, le réseau internet et les capteurs « intelligents ».

Le positionnement adopté dans cette thèse est situé à l'intersection centrale. Nous souhaitons un système « smart » qui envoie de manière automatique les données vers la base de données sans intervention du citoyen. Le citoyen participe de manière passive, en intégrant à son activité le capteur, mais sans action spécifique à la collecte de l'information. En effet, de cette manière nous souhaitons nous affranchir des biais sociaux inhérents à la collecte des données par une population active en science participative. Cette ambition a posé des contraintes très importantes sur la conception du capteur (cf. chapitre 3) ; particulièrement en ce qui concerne l'autonomie énergétique et la communication des données en temps réel. Nous verrons au chapitre 4 que la démarche a évolué vers une participation plus active du citoyen.

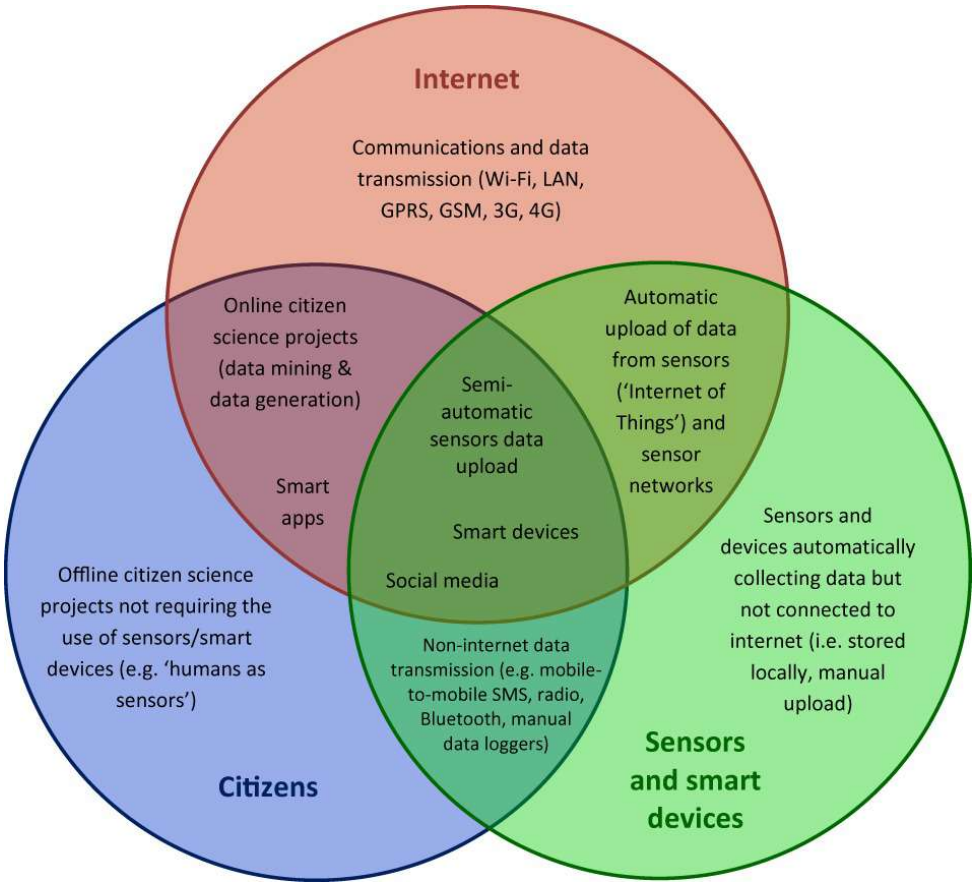


FIGURE 1.11 – Diagramme de Venn montrant les interactions entre le crowdsensing participatif ou non participatif et la technologie — d'après Muller et *al.* (2015).

1.5.3 Plateforme de mobilité

Le choix de la plateforme de transport est crucial pour concilier qualité de la mesure et couverture spatio-temporelle dans le respect des contraintes de déploiement.

D'une part, la plateforme de transport limite l'encombrement et la consommation énergétique du capteur et ainsi la technologie utilisée, de laquelle dépend la précision de la mesure.

D'autre part, la mobilité de la plateforme dépend de la distance moyenne parcourue (évidemment différente entre un piéton et une voiture) et de la fréquence de déplacement (par exemple régulière pour des services publiques, sujette à motivation pour des participants volontaires).

De plus, d'un point de vue pratique, le coût du projet, le temps de maintenance et le temps d'autonomie du système (endurance) en énergie et pour stocker les données sont à prendre en compte. Par exemple, la configuration de la plateforme et le dimensionnement du réseau peuvent impacter le budget et conduire à choisir une famille de capteurs moins chère ou peuvent réduire le nombre de plateformes équipées.

Les systèmes participatifs doivent être relativement simples d'utilisation et peu encombrants. Leur mise en œuvre nécessite une attention plus particulière concernant la source d'alimentation mais peut éventuellement profiter du téléphone portable de l'utilisateur

(date, données GPS, accéléromètre et réseau Internet) et du partage des frais du système. Les données sont moins précises (car l'étalonnage est plus compliqué) et collectées en des positions non imposées. Cependant, statistiquement ces positions couvrent plus finement les zones les plus fréquentées qui sont des zones intéressantes pour estimer l'exposition de la population à la pollution (une des destinations de la modélisation de la pollution de l'air). La vocation de ces données est généralement d'être rendues publiques, ce qui pose un problème de préservation de la vie privée des utilisateurs.

Les réseaux de capteurs sur véhicule ont l'avantage de ne généralement pas être contraints en alimentation, en poids ni en taille du système ; ce qui permet de disposer de plusieurs capteurs et d'obtenir des données précises. De plus, ils couvrent une large zone d'étude, mais au détriment de la résolution temporelle (Wong et *al.*, 2009). Enfin, l'acquisition de la position et la communication des données sont des enjeux majeurs et le prix d'un tel système est élevé.

1.5.4 Mobilité : vie privée *versus* utilité

Les données ouvertes (Open Data en anglais) sont des données librement accessibles que tout un chacun peut utiliser, modifier et rediffuser quelque soit son but.

Dans le contexte du crowdsensing, les données sont collectées par des participants. Ces données peuvent contenir des informations susceptibles de porter atteinte à leur vie privée et cela de manière individuelle ou collective.

Une étude récente suggère qu'aucun jeu de données, quelque soit la méthode d'anonymisation (existante ou imaginable), ne peut suivre les directives du Règlement Général sur la Protection des Données (Rocher et *al.*, 2019). Le recoupement avec d'autres jeux de données publiques est une source de complexité du problème. Par exemple, dans le Massachusetts, la Commission des assurances collectives (GIC), responsable de l'assurance maladie des employés d'État, collecte des informations détaillées (avec plus d'une centaine de méta-données par visite médicale). Ces informations supposées anonymisées ont été diffusées librement. Néanmoins, par association à la base de données des votants de l'État, il est possible de retrouver des dossiers médicaux de personnes connues, tel que celui de William Weld, alors gouverneur du Massachusetts (Sweeney, 2002).

Dans la méthodologie que nous développons, les informations de date et heure, de position géographique et de concentration de polluants sont les paramètres que nous souhaitons récolter a minima. D'autres méta-données portant sur le profil de l'utilisateur pourraient également être intéressantes.

Dans le cas de données de géolocalisation, la suppression des « données personnelles » n'est souvent pas suffisante pour protéger la vie privée. Les méthodes d'exploration de ce type de données s'améliorent constamment et peuvent révéler l'identité et le comportement de la personne les ayant collectées (Sweeney, 2002).

Par exemple, le simple fait de repérer le domicile et le lieu de travail, à l'aide des motifs de déplacement réguliers le matin et le soir, permet de réduire considérablement le panel de personnes envisagées.

Ben Mokhtar et *al.* (2017) étudient l'unicité de la mobilité humaine à faible résolution spatio-temporelle à partir de données de géolocalisation provenant de capteurs GPS, GSM ou Wi-Fi. Les informations spatiales retenues sont l'adresse MAC des points d'accès pour

le Wi-Fi, l'identifiant des antennes pour le GSM ou la rencontre de points d'intérêt pour le GPS (où les points d'intérêt sont déterminés préalablement grâce à l'ensemble des positions GPS). La résolution temporelle est de l'ordre de quelques dizaines de minutes. Malgré cela, les auteurs montrent qu'il suffit de 4 points spatio-temporels pour identifier 97 % des utilisateurs.

Or dans notre cas, dégrader aussi brutalement les données n'est pas envisageable au regard de leur utilité. En effet, l'intérêt de ce type de données, pour la modélisation de la pollution de l'air, est justement d'augmenter la résolution spatiale comparativement à celle des SSN.

Dans le cas où de nombreux utilisateurs sont mis en jeu, une solution pour trouver un compromis, entre vie privée et utilité, est d'étudier le recouvrement des zones spatiales visitées par les utilisateurs (Cerf et *al.*, 2017).

Une autre technique intéressante proposée par Primault et *al.* (2015) est d'échanger les trajectoires de deux utilisateurs au moment d'un *Rendez-Vous* pour complexifier la tâche d'un attaquant. Dans notre cas, cette technique suppose d'avoir préalablement inter-étalonné les capteurs de terrain (pollution, conditions météorologiques...) sinon cela biaiserait l'étalonnage ultérieur du jeu de données et il serait assez simple pour l'attaquant de déterminer ces échanges.

Ainsi, la problématique de la protection de la vie privée est cruciale dans le cas de collecte de données scientifiques via crowdsensing. Avant de pouvoir généraliser la méthodologie que nous développons et diffuser les données collectées, il apparaît nécessaire d'y répondre. Cependant, cet aspect n'est pas abordé dans cette thèse qui se consacre aux mesures et aux méthodes d'analyse.

1.6 Conclusion et consécution des chapitres suivants

Ce premier chapitre expose la vision adoptée selon la science des données et la trilogie qui la régit : collecte de données, analyse des données – via l'apprentissage automatique et les systèmes répartis – et prise de décision.

Le cas d'étude s'oriente autour de la pollution de l'air car il s'agit d'un phénomène environnemental complexe et d'actualité, mais aussi parce que l'émergence de capteurs miniatures et à bas coût permet d'imaginer des systèmes de capture mobiles. A l'échelle urbaine, les réactions physico-chimiques mises en jeu par les polluants sont nombreuses, interdépendantes, hétérogènes et peuvent avoir des temps de vie très courts. Ces systèmes de mesure mobiles semblent alors intéressants pour capter au maximum la variabilité du phénomène.

Néanmoins, le compromis entre couverture spatiale et qualité du capteur embarqué conduit à plusieurs défis : le choix de la plateforme de transport et la vie privée des utilisateurs, l'étalonnage automatique d'une flotte de capteurs et l'exploitabilité des données collectées.

Les contraintes attendues concernant les différents type de réseaux (CSN, VSN et SSN) sont représentées graphiquement sur le diagramme de Kiviat de la Figure 1.12. Ces pondérations ont été estimées après la réalisation d'un état de l'art des projets existants par Yi et *al.* (2015). Nous constatons globalement que les CSN semblent économiques,

ont une bonne couverture spatiale mais ont aussi des données de moins bonne qualité pour un coût de maintenance élevé. Les SSN qui visent une observation de qualité avec une excellente résolution temporelle, souffrent d'un coût élevé et ont une très mauvaise couverture spatiale. Enfin les VSN semblent être un bon compromis.

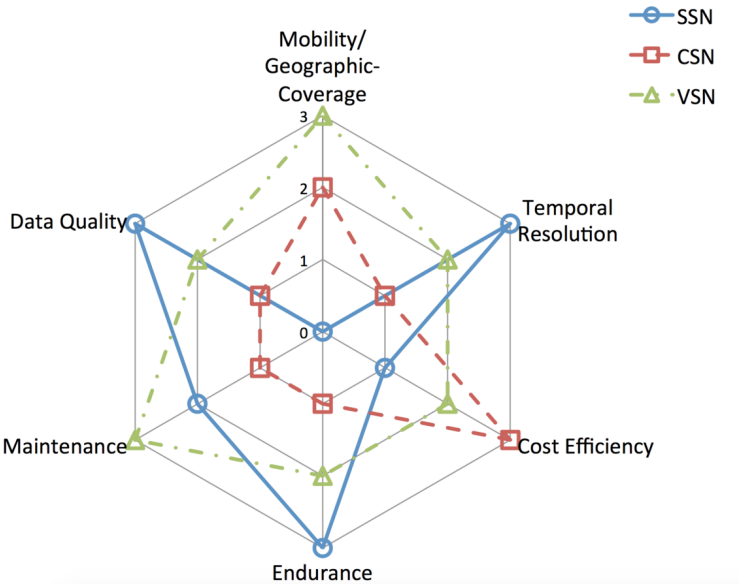


FIGURE 1.12 – Diagramme de Kiviat des contraintes des trois types de réseaux de capteurs — d'après Yi et *al.* (2015).

Cette thèse tire partie des VSN et des CSN pour optimiser le coût et la couverture spatiale.

Bibliographie

ASCHER, M. . Models and Maps from the Marshall Islands : A Case in Ethnomathematics. Historia Mathematica, 1995. DOI : 10.1006/hmat.1995.1030.

BEN MOKHTAR, S. , BOUTET, A. , BOUZOUINA, L. , BONNEL, P. , BRETTE, O. , BRUNIE, L. , CUNCHE, M. , D 'ALU, S. , PRIMAULT, V. , RAVENEAU, P. , RIVANO, H. et STANICA, R. . PRIVA'MOV : Analysing Human Mobility Through Multi-Sensor Datasets. NetMob 2017, 2017. https://hal.inria.fr/hal-01578557.

BENGIO, Y. , COURVILLE, A. et VINCENT, P. . Representation Learning : A Review and New Perspectives. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012. DOI : 10.1109/TPAMI.2013.50.

BRUNEKREEF, B. et HOLGATE, S. T. . Air pollution and health. The Lancet, 2002. DOI : 10.1016/S0140-6736(02)11274-8.

CERF, S. , PRIMAULT, V. , BOUTET, A. , BEN MOKHTAR, S. , BIRKE, R. , BOUCHENAK, S. , CHEN, L. Y. , MARCHAND, N. et ROBU, B. . PULP : Achieving Privacy and Utility

Trade-off in User Mobility Data. SRDS 2017 - 36th IEEE International Symposium on Reliable Distributed Systems, 2017. DOI : 10.1109/SRDS.2017.25.

Davenport, W. . Marshall Islands Navigational Charts. Imago Mundi, 1960. DOI : 10.1080/03085696008592173.

Dickinson, J. L. , Zuckerberg, B. et Bonter, D. N. . Citizen Science as an Ecological Research Tool : Challenges and Benefits. Annual Review of Ecology, Evolution, and Systematics, 2010. DOI : 10.1146/annurev-ecolsys-102209-144636.

Dietterich, T. G. . Ensemble Methods in Machine Learning. Multiple Classifier Systems, 2000. DOI : 10.1007/3-540-45014-9_1.

Dobre, A. , Arnold, S. J. , Smalley, R. J. , Boddy, J. W. D. , Barlow, J. F. , Tomlin, A. S. et Belcher, S. E. . Flow field measurements in the proximity of an urban intersection in London, UK. Atmospheric Environment, 2005. DOI : 10.1016/j.atmosenv.2005.04.015.

Domingos, P. . A few useful things to know about machine learning. Communications of the ACM, 2012. DOI : 10.1145/2347736.2347755.

Donoho, D. . 50 Years of Data Science. Journal of Computational and Graphical Statistics, 2017. DOI : 10.1080/10618600.2017.1384734.

Fayyad, U. . From Data Mining to Knowledge Discovery in Databases. AI Magazine, 1996. DOI : 10.1609/aimag.v17i3.1230.

Fayyad, U. , Piatetsky-Shapiro, G. et Smyth, P. . Knowledge Discovery and Data Mining : Towards a Unifying Framework. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 1996. http://dl.acm.org/citation.cfm?id=3001460.3001477.

Feinberg, R. , Dymon, U. J. , Paiaki, P. , Rangituteki, P. , Nukuriaki, P. et Rollins, M. . 'Drawing the Coral Heads' : Mental Mapping and its Physical Representation in a Polynesian Community. The Cartographic Journal, 2003. DOI : 10.1179/000870403225012943.

Hasenfratz, D. , Saukh, O. et Thiele, L. . On-the-fly calibration of low-cost gas sensors. Wireless Sensor Networks, 2012. DOI : 10.1007/978-3-642-28169-3_15.

Heuschling, X. . Manuel de statistique ethnographique universelle, précédé d'une introduction théorique d'après l'état actuel de la science. Societe typographique belge, A. Wahlen et compagnie Bruxelles, 1847. ISBN 978-2-01-306720-1.

Hombert, M. et Préaux, C. . Recherches sur le recensement dans l'Egypte romaine. Lugduni Batavorum : Brill, 1952. http://lib.ugent.be/catalog/rug01:001811550.

Kranakis, E. , Krizanc, D. et Rajsbaum, S. . Mobile Agent Rendezvous : A Survey. Structural Information and Communication Complexity. Springer Berlin Heidelberg, 2006. DOI : 10.1007/11780823_1. Series Title : Lecture Notes in Computer Science.

Kumar, P. , Morawska, L. , Martani, C. , Biskos, G. , Neophytou, M. , Di Sabatino, S. , Bell, M. , Norford, L. et Britter, R. . The rise of low-cost sensing for managing air pollution in cities. Environment International, 2015. DOI : 10.1016/j.envint.2014.11.019.

Mage, D. , Ozolins, G. , Peterson, P. , Webster, A. , Orthofer, R. , Vandeweerd, V. et Gwynne, M. . Urban air pollution in megacities of the world. Atmospheric Environment, 1996. DOI : 10.1016/1352-2310(95)00219-7.

McMahan, H. B. , Moore, E. , Ramage, D. , Hampson, S. et Arcas, B. A. y. . Communication-Efficient Learning of Deep Networks from Decentralized Data. Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, 2016. https://arXiv.org/abs/1602.05629.

Mead, M. I. , Popoola, O. A. M. , Stewart, G. B. , Landshoff, P. , Calleja, M. , Hayes, M. , Baldovi, J. , McLeod, M. , Hodgson, T. , Dicks, J. , Lewis, A. , Cohen, J. , Baron, R. , Saffell, J. et Jones, R. . The use of electrochemical sensors for monitoring urban air quality in low-cost, high-density networks. Atmospheric Environment, 2013. DOI : 10.1016/j.atmosenv.2012.11.060.

Meier, F. , Fenner, D. , Grassmann, T. , Otto, M. et Scherer, D. . Crowdsourcing Air Temperature from Citizen Weather Stations for Urban Climate Research. Urban Climate, 2017. DOI : 10.1016/j.uclim.2017.01.006.

Menini, P. . Du capteur de gaz à oxydes métalliques vers les nez électroniques sans fil. Habilitation à diriger des recherches, Université Paul Sabatier - Toulouse III, 2011. https://hal.archives-ouvertes.fr/tel-00697471.

Menut, L. , Bessagnet, B. , Khvorostyanov, D. , Beekmann, M. , Blond, N. , Colette, A. , Coll, I. , Curci, G. , Foret, G. , Hodzic, A. , Mailler, S. , Meleux, F. , Monge, J.-L. , Pison, I. , Siour, G. , Turquety, S. , Valari, M. , Vautard, R. et Vivanco, M. G. . CHIMERE 2013 : A Model for Regional Atmospheric Composition Modelling. Geoscientific Model Development, 2013. DOI : https://doi.org/10.5194/gmd-6-981-2013. Publisher : Copernicus GmbH.

Moltchanov, S. , Levy, I. , Etzion, Y. , Lerner, U. , Broday, D. M. et Fishbain, B. . On the feasibility of measuring urban air pollution by wireless distributed sensor networks. Science of The Total Environment, 2015. DOI : 10.1016/j.scitotenv.2014.09.059.

Muller, C. L. , Chapman, L. , Johnston, S. , Kidd, C. , Illingworth, S. , Foody, G. , Overeem, A. et Leigh, R. R. . Crowdsourcing for Climate and Atmospheric Sciences : Current Status and Future Potential. International Journal of Climatology, 2015. DOI : 10.1002/joc.4210.

Oriol, J.-C. . Éléments d'histoire de la statistique. Statistix, 2010. https://hal.archives-ouvertes.fr/inria-00466297.

Piatetsky-Shapiro, G. . Knowledge Discovery in Real Databases : A Report on the IJCAI-89 Workshop. AI Magazine, 1991. DOI : 10.1609/aimag.v11i4.873.

Predd, J. B. , Kulkarni, S. B. et Poor, H. V. . Distributed learning in wireless sensor networks. IEEE Signal Processing Magazine, 2006. DOI : 10.1109/MSP.2006.1657817.

Primault, V. , Mokhtar, S. B. et Brunie, L. . Privacy-Preserving Publication of Mobility Data with High Utility. 2015 IEEE 35th International Conference on Distributed Computing Systems. IEEE, 2015. DOI : 10.1109/ICDCS.2015.117.

Radosavljevic, V. , Vucetic, S. et Obradovic, Z. . Continuous Conditional Random Fields for Regression in Remote Sensing. 19th European Conference on Artificial Intelligence, 2010. DOI : 10.3233/978-1-60750-606-5-809.

Robertsson, L. , Iliev, B. , Palm, R. et Wide, P. . Perception modeling for human-like artificial sensor systems. International Journal of Human-Computer Studies, 2007. DOI : 10.1016/j.ijhcs.2006.11.003.

Rocher, L. , Hendrickx, J. M. et de Montjoye, Y.-A. . Estimating the success of re-identifications in incomplete datasets using generative models. Nature Communications, 2019. DOI : 10.1038/s41467-019-10933-3.

Rodriguez, D. , Valari, M. , Payan, S. et Eymard, L. . On the spatial representativeness of NOX and PM10 monitoring-sites in Paris, France. Atmospheric Environment : X, 2019. DOI : 10.1016/j.aeaoa.2019.100010.

Roussel, I. . Climatologie géographiques et pollutions atmosphériques : quelles synergies ?, 2006. Climat et Société : L'apport Des Géographes-Climatologues, Journées de la climatologie, Nice.

Saukh, O. , Hasenfratz, D. , Walser, C. et Thiele, L. . On rendezvous in mobile sensing networks. Real-World Wireless Sensor Networks, 2014. DOI : 10.1007/978-3-319-03071-5_3.

Sportisse, B. . Pollution atmosphérique : des processus à la modélisation. Springer, 2008. ISBN 978-2-287-74961-2.

Sweeney, L. . k-Anonymity : a model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 2002. DOI : 10.1142/S0218488502001648.

Wong, K.-J. , Chua, C. C. et Li, Q. . Environmental Monitoring Using Wireless Vehicular Sensor Networks. 5th International Conference on Wireless Communications, Networking and Mobile Computing, 2009. DOI : 10.1109/WICOM.2009.5303846.

Xiang, Y. , Bai, L. , Piedrahita, R. , Dick, R. P. , Lv, Q. , Hannigan, M. et Shang, L. . Collaborative calibration and sensor placement for mobile sensor networks. Proceedings of the 11th international conference on Information Processing in Sensor Networks, 2012. DOI : 10.1145/2185677.2185687.

Yi, W. Y. , Lo, K. M. , Mak, T. , Leung, K. S. , Leung, Y. et Meng, M. L. . A Survey of Wireless Sensor Network Based Air Pollution Monitoring Systems. Sensors, 2015. DOI : 10.3390/s151229859.

Chapitre 2

Approche théorique d'un réseau de capteurs mobiles

Revois deux fois pour voir juste, ne vois qu'une fois pour voir beau.
– Henri-Frédéric Amiel

Sommaire

2.1 Introduction

Actuellement les agences de qualité de l'air et les organismes officiels s'appuient essentiellement sur des réseaux statiques de stations de mesure (SSN). Ces stations de mesure sont équipées d'instruments très performants et capables de mesurer une grande gamme de polluants. Cependant, ces stations sont généralement très coûteuses et nécessitent une maintenance importante, ce qui limite le déploiement sur de vastes réseaux. Il en résulte que les variabilités de concentrations de polluants à petite échelle (typiquement la rue) échappent à ces mesures. La complémentarité avec un réseau de stations mobiles est évidente mais n'est pas couramment mis en œuvre.

Nous nous intéressons ici à la variabilité des polluants à l'échelle intra-urbaine et nous proposons de nous appuyer sur les mesures effectuées par un système mobile (i.e. les VSN et certains CSN). Les réseaux mobiles représentent un bon compromis entre résolutions spatiales et temporelles. En effet, les stations mobiles couvrent une grande zone d'espace sans pour autant nécessiter un grand nombre de stations. L'inconvénient majeur de la mobilité est que le temps d'échantillonnage en un point donné est grandement réduit. Il existe donc un challenge pour reconstituer à partir de mesures mobiles une représentation globale de la pollution à l'échelle intra-urbaine et donc pour trouver un support mobile approprié.

En amont d'une expérience *in situ* utilisant les vélos comme support mobile, nous effectuons tout d'abord une étude théorique de restitution de la variabilité spatiale des polluants. L'objectif est d'évaluer la capacité d'un réseau mobile à rendre compte d'une distribution spatiale de polluants, compte tenu du nombre d'éléments dans le réseau et de leur parcours moyen. Nous nous plaçons dans le cas idéal où la distribution spatiale des polluants est connue a priori. Un réseau virtuel de capteurs sur vélo vient échantillonner cette distribution spatiale. La distribution spatiale des polluants est estimée sur l'ensemble du domaine à partir des observations mobiles et de méthodes statistiques *ad hoc*. Les résultats de la prévision à partir des mesures mobiles sont ensuite comparés avec la distribution connue a priori.

Le point de départ de cette étude théorique est une carte de distribution des polluants à l'échelle d'une ville. Afin d'avoir une étude la plus réaliste possible, cette carte est simulée à partir d'un modèle numérique permettant de transporter les espèces chimiques (cf. chapitre 1) à des échelles spatiales compatibles avec les trajectoires des vélos, typiquement une résolution de l'ordre de la dizaine de mètres. De cette carte sont tirées des observations synthétiques, i.e. la concentration de polluants « vue » par le vélo pour une position géographique donnée correspond à la concentration de polluants où se situe le vélo dans la grille du modèle numérique. Chaque trajet de vélo génère donc un ensemble de données synthétiques.

Nous cherchons à estimer, à partir de ces données synthétiques, la carte de répartition spatiale des polluants la plus vraisemblable. La vraisemblance est estimée a posteriori en comparant la carte simulée avec la carte originale servant ainsi de référence. Dans cette étude, la temporalité du phénomène est ici négligée. Nous considèrons que les trajectoires sont instantanées et simultanées. Les méthodes de spatialisation sont basées sur l'utilisation de critères géographiques et sociologiques décrivant la ville. L'hypothèse sous-jacente est donc que sa structure détermine en partie la répartition des polluants à travers celle-ci.

La Figure 2.1 présente la méthode suivie. Durant le pré-traitement, les trajets de vélos générés aléatoirement sont associés à des mesures synthétiques de la pollution de l'air, par correspondance à la grille du modèle de simulation numérique. La variabilité sous maille, c'est-à-dire à une résolution spatiale plus fine que celle du modèle, n'est donc pas représentée. Les variables explicatives géographiques sont principalement extraites depuis OpenStreetMap (OSM).

Durant le traitement, les algorithmes choisis de spatialisation sont appliqués au jeu de données ainsi formé. Le but des algorithmes est de prédire la pollution de l'air sur toute la zone traversée par les vélos fictifs. Durant le post-traitement, les prédictions des algorithmes sont comparées avec la carte de référence à l'aide de métriques classiques : RMSE (Root-mean-square deviation), MAE (Mean absolute error) et le coefficient de corrélation.

Enfin, cette méthode nous permet de caractériser la plateforme de transport au travers du nombre de systèmes et de la période d'échantillonnage. Pour évaluer les performances de chaque algorithme, nous expliquons d'abord les sources principales d'erreurs dans les prédictions puis nous comparons leur capacité à détecter des anomalies spatiales.

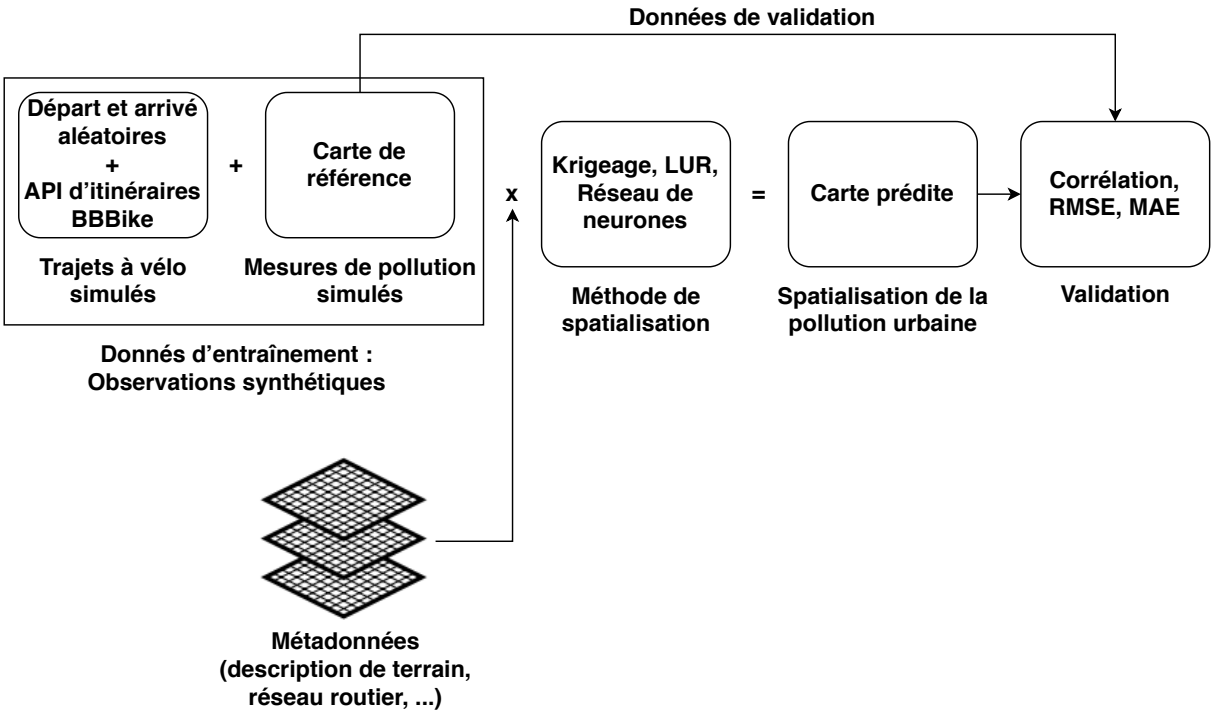


FIGURE 2.1 – Méthode suivie.

2.2 État de l'art

2.2.1 Réseau de capteurs mobiles en zone urbaine

Il existe une littérature assez abondante sur l'utilisation de systèmes portables pour la mesure de la qualité de l'air en réseau. La Table 2.1 présente un échantillon de projets de type CSN et VSN basés sur l'utilisation de capteurs de qualité de l'air – sur la base du travail de Yi et *al.* (2015). De nombreux autres projets visent l'estimation de la pollution de l'air à l'échelle urbaine. Google & Aclima (partenariat entre Google et une startup), par exemple, tente de cartographier la ville de Denver aux USA en disposant des capteurs de pollution sur des Google StreetCar. Le projet BeMap, réalisé par des étudiants de l'EPFL, embarque des capteurs sur des vélos, les déploie temporairement à Sao Paolo et Rio de Janeiro au Brésil, puis récupère les données pour les afficher. Concernant les CSN, le projet hackAir (Kosmidis et *al.*, 2018) est sans doute le plus avancé à l'heure actuelle à l'échelle planétaire. Il donne la possibilité au citoyen de construire facilement son propre module transportable de captation de la pollution et de transmission des données. CitiSense (Bales et *al.*, 2012) et AirVisual (startup) fournissent aux citoyens un boîtier mobile permettant de capter respectivement O_3, CO et CO_2, $PM_{2.5}$, météorologie. Les informations sont collectées et diffusées en l'état sur une application mobile ou un site web. Cependant ces projets, donnent seulement une vision du phénomène de pollution avec une faible granularité.

Nous remarquons que de nombreux modes de transport sont étudiés (voiture, vélo, bus, tramway, taxi, piéton, oiseau...), avec une préférence pour la voiture, principalement car elle assure une source d'énergie fiable. Mais pour l'instant seules de petites flottes de systèmes mobiles (au mieux, de l'ordre de la dizaine de systèmes) sont utilisées. La grande majorité des systèmes mobiles embarquent des micro-capteurs (semi-conducteurs, électrochimique, à absorption infrarouge et à photo-ionisation). En effet, ils font partie de ceux qui consomment le moins d'énergie et sont les moins encombrants. Une comparaison des différents types de capteurs de gaz est présentée au chapitre 3.

Dans l'étude bibliographique établie, il apparaît que le projet suisse OpenSense est celui de type VSN qui a la méthodologie la plus aboutie. Hasenfratz et *al.* (2012) ont mené une première étude de faisabilité basée sur l'utilisation de capteurs d'ozone connectés à un smartphone. L'étalonnage des capteurs est effectué par comparaison avec un capteur fixe de référence lors de *Rendez-Vous* (cf. section 1.3.2). L'impact du déplacement sur la précision de la mesure est également étudié brièvement.

Un appareil de mesure de la pollution atmosphérique beaucoup plus développé (O_3, particules ultrafines, CO, NO_2) a ensuite été installé sur le réseau de tramway de la ville de Zurich (Hasenfratz et *al.*, 2015) permettant la production d'un jeu de données de 3 000 000 de points en 6 mois (Li et *al.*, 2012). En reprenant la taxonomie proposée par Delaine et *al.* (2019), les différents aspects pratiques inhérents à l'étalonnage collaboratif des capteurs mobiles sont abordés par ce groupe de travail : étalonnage entre paire de capteurs (pairwise calibration) ou étalonnage par étude de l'ensemble des capteurs (macro calibration), étalonnage entre capteurs mobiles et/ou à l'aide de stations fixes, étalonnage entre capteurs de même qualité (blind calibration) ou à l'aide d'un instrument de référence.

L'étalonnage par *Rendez-Vous* permet de confronter les valeurs de deux systèmes lors-

qu'ils se trouvent dans une même région de l'espace en même temps. Les *Rendez-Vous* permettent également de détecter des capteurs défectueux. Saukh et *al.* (2014) étudient la définition optimale de la taille de la région et de la durée d'un *Rendez-Vous* par rapport à leur jeu de données. Ils introduisent également la notion de propagation « multi-hop » de l'erreur d'étalonnage par *Rendez-Vous* dans un réseau (Saukh et *al.*, 2015). Cette notion est très importante dans le cadre de vastes réseaux ayant des capteurs sujets à dérive temporelle. Ils montrent que la régression linéaire, généralement utilisée pour déterminer les coefficients d'étalonnage, provoque l'accumulation d'erreurs lors de la multiplication du nombre de *Rendez-Vous*. Une méthode basée sur la régression géométrique est plus adaptée pour limiter ce problème. Les mesures en particules ultra-fines (ultrafine particles, UPF) issues du réseau de tramway sont utilisées pour créer des cartes statistiques (Hasenfratz et *al.*, 2015; Mueller et *al.*, 2016) ou des graphes du réseau routier (Marjovi et *al.*, 2015) de la pollution de l'air à Zurich. Ces résultats ont largement inspiré ce travail et les modèles utilisés sont davantage présentés section 2.2.2.

Ainsi, nous avons choisi, au travers du projet BICLUE, d'embarquer des micro-capteurs sur des vélos. En effet, le vélo est une plateforme de transport intéressante car (i) relativement peu de projets l'étudient, (ii) les distances des trajets sont plus grandes qu'à pied et moins contraintes qu'en voiture, (iii) ne pollue pas (et donc ne biaise pas la collection des données), (iv) couvre naturellement les zones les plus fréquentées et (v) dispose de communautés de cyclistes soucieux de l'air qu'ils respirent. La Figure 2.2 reprend le diagramme de Kiviat présenté section 1.5.3 et positionne notre ambition au regard des six contraintes définies. Comme les CSN, nous souffrons de problèmes de maintenance et de qualité de la donnée au prix de l'avantage financier. Néanmoins, nous nous en distinguons de par la plateforme de transport utilisée, le vélo, qui améliore la mobilité. De plus, l'utilisation d'une dynamo peut permettre de réaliser un système autonome en énergie, donc plus endurant et plus utilisé, et ainsi avec une meilleure résolution temporelle.

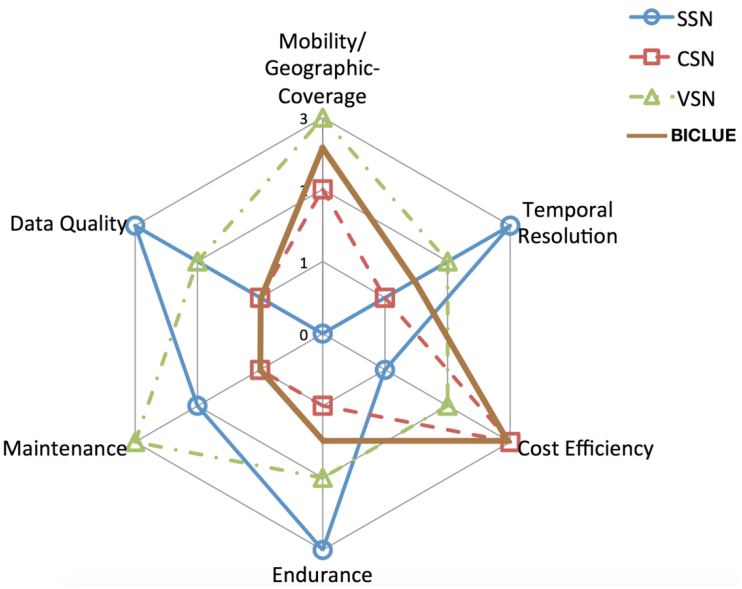


FIGURE 2.2 – Positionnement de notre projet sur le diagramme de Kiviat de Yi et *al.* (2015).

TABLE 2.1 – Principaux projets de réseaux de capteurs mobiles (type CSN et VSN) pour l'étude de la qualité de l'air.

	Référence	Plateforme	Échelle	Type de réseau	Capteurs
CSN	Bales et *al.* (2012)	Piéton	ville, San Diego, USA	8 participants / 1 mois	NO_2, O3 et CO
	Hasenfratz et *al.* (2012)	vélo	ville, Zurich	1 capteur / 2 mois	Semi-conducteur (O3)
	Li et *al.* (2012)	Tramway	ville, Zurich (100 km^2)	10 capteurs mobiles, 2 capteurs fixes / 3 mois	Semi-conducteur (O3), Alphasense (CO), Matter-Aerosol (PM)
	Honicky et *al.* (2008)	Taxi	Ville Accra Ghana	1 capteur / 1 jour	CO, NO_x
	Méndez et *al.* (2011)	Boîtier	NA	1 capteur	Semi-conducteur (CO2, COV), Combustion catalytique (H2), Électrochimique (CO)
VSN	Lo Re et *al.* (2014)	bus	Ville Palerme	NA	Semi-conducteur (CO, NO_2, O3, CO2)
	Völgyesi et *al.* (2008)	Voiture	NA	NA	Semi-conducteur (CO, NO_2, O3)
	Hu et *al.* (2009)	Voiture	NCTU Campus, China	16 véhicules / NA	Absorption infrarouge (CO2)
	Devarakonda et *al.* (2013)	Voiture	Périphérique/ Autoroute	1 véhicule	Semi-conducteur (CO), Analyseur optique (PM)
	Al-Ali et *al.* (2010)	Bus	NA	1 bus	Électrochimique (CO, SO2 , NO_2)
	Wong et *al.* (2009)	Voiture	zone industrielle	1 véhicule	Analyseur optique (PM), Électrochimique (CO, NO_2, NO, COV)

2.2.2 Familles de méthodes statistiques de spatialisation

L'objectif de la méthode de spatialisation est d'obtenir une prévision sur l'ensemble du domaine du niveau des polluants à partir des mesures mobiles.

Lorsque des mesures de la variable explicative sont disponibles, les techniques classiques de régression et d'approximation sont couramment utilisées et éventuellement adaptées à l'aspect géolocalisé du problème. Il s'agit de méthodes modélisant la variable explicative [1] dans l'espace par une fonction continue passant aux abords des dites mesures de la variable explicative. Deux sous-catégories se distinguent : les modèles de proximité et les modèles de type Land-Use Regression (LUR).

Sans doute la plus utilisée (Jerrett et *al.*, 2004) des méthodes de proximité, le Krigeage minimise la variance spatiale du résidu (erreur d'estimation) sans biais. De nombreuses autres versions sont possibles. Janssen et *al.* (2008) proposent une version améliorée du Krigeage en éliminant les tendances locales au regard des données sur l'utilisation des sols. Une autre méthode très populaire pour sa simplicité est l'Inverse Distance Weighting. Sivaraman et *al.* (2013) la comparent au Krigeage. Ce dernier est plus facile à mettre en œuvre, mais le Krigeage ordinaire obtient des résultats plus robustes.

Les LUR sont un ensemble de techniques basées sur des variables explicatives liées aux données de terrain, telles que le type de végétation, l'activité humaine (zone industrielle, résidentielle...), la proximité à des axes routiers, l'intensité du trafic routier, la vitesse du vent, la densité de population..., qui tirent partie des corrélations entre ces variables explicatives. Ce type de modèle a très largement été appliqué (Su et *al.*, 2009; Hasenfratz et *al.*, 2015), comparé (Hoek et *al.*, 2008; Ghassoun et *al.*, 2015) et couplé avec d'autres modèles (Adams et Kanaroglou, 2016; Janssen et *al.*, 2008). Traditionnellement les LUR utilisent des techniques de régression multi-linéaire qui relient la concentration du polluant à des estimateurs spatiaux. Des techniques basées sur des régressions non-linéaires sont également utilisées, telles que les modèles additifs généralisés (Generalized Additive Model, GAM). Les GAM sont appréciés pour leur capacité à capturer des phénomènes non linéaires (Hasenfratz et *al.*, 2015; Mueller et *al.*, 2016), et étendus en les couplant à l'analyse en composantes principales (PCA) pour diminuer la dimension de l'espace des variables explicatives (Li et *al.*, 2017). Ces approches peuvent être combinées, par exemple en modélisant la dérive spatiale du phénomène dans le cas du Krigeage Universel et du Krigeage avec dérive externe. Mercer et *al.* (2011) comparent le Krigeage Universel avec les LUR.

Enfin, les méthodes d'apprentissage statistique, où chaque donnée vient affiner la modélisation, sont de plus en plus en vogue et redorent les méthodes de classification. Les réseaux de neurones (Adams et Kanaroglou, 2016; Kurt et *al.*, 2008) ont été le plus rapidement adoptés. Un design de neural network spécifique aux stations fixes pour la qualité de l'air (Zheng et *al.*, 2015) et une approche pour designer les neural network en fonction d'une simulation à l'aide d'algorithmes génétiques ont été proposés (Niska et *al.*, 2004). D'autres méthodes ont également été testées, telles que les Probabilistic Graph Model (PGM) (Marjovi et *al.*, 2015) ou la logique floue à partir de données discrétisées (Onkal-Engin et *al.*, 2004), voire même développées particulièrement pour le domaine de l'étude de la pollution de l'air (Hsieh et *al.*, 2015).

1. Aussi appelée « méta-donnée » ou « caractéristique » en apprentissage automatique.

L'aspect temporel est moins étudié. Il s'agit souvent d'une variable explicative comme une autre, parfois discrétisée comme le souligne Dons et *al.* (2013), et intégrée au modèle d'estimation. Une simple interpolation est opérée, voire des prédictions successives. D'autres modèles cependant utilisent les tendances des séries temporelles de mesures de la variable explicative collectées par des stations de mesure fixes (Romanowicz et *al.*, 2006; Qi Gan et *al.*, 2011) ou encore se servent des réseaux de neurones (Russo et *al.*, 2013) pour la prédiction temporelle d'estimations spatiales, voire établissent une méthode spécifique (Zheng et *al.*, 2015). De plus, les deep-neural network sont de plus en plus exploités, notamment de type récurrent (ou aussi appelés spatio-temporel).

Dans cette étude comparative nous nous intéressons au Krigeage (modèle de proximité), au GAM (LUR) et aux réseaux de neurones (apprentissage automatique). Nous présenterons ces méthodes d'estimation par ordre de complexité croissante (et interprétabilité décroissante). Le Krigeage capture mieux les phénomènes linéaires alors que les réseaux de neurones approximent aussi bien des phénomènes linéaires que non linéaires, mais sont considérés comme des « boîtes noires ». Ces modèles sont présentés succinctement dans les sections qui suivent.

2.2.2.1 Krigeage

Nous proposons ici une description simplifiée du Krigeage. Davantage de détails sont présentés en annexe 5, notamment sur la base du mémoire de (Baillargeon, 2005).

Le Krigeage est une méthode d'estimation par auto-corrélation spatiale, c'est-à-dire que les prédictions s'effectuent à l'aide de la connaissance de données voisines, et que leur impact sur la prédiction est évalué en fonction de leur corrélation. En d'autres termes, la prédiction s'écrit

$$\hat{Y}(s_0) = \sum_{s_i \in V(s_0)} \lambda_i Y(s_i) \tag{2.1}$$

où s_0 est la position de la prédiction, $V(s_0)$ le voisinage de s_0 et s_i est une position où la variable explicative Y est connue.

Le Krigeage a la particularité de minimiser la variance spatiale du résidu (erreur d'estimation) sans biais, c'est-à-dire qu'il cherche

$$min(Var[\hat{Y}(s_0) - Y(s_0)]) \text{ subject to } E[\hat{Y}(s_0) - Y(s_0)] = 0. \tag{2.2}$$

Cette particularité suppose l'existence mathématique de la variance du résidu et du biais, ce qui ajoute une nouvelle contrainte (2.3). Les hypothèses couramment choisies sont la stationnarité de second ordre ou la stationnarité intrinsèque. Elles sont décrites en annexe 5.

De plus, le modèle du Krigeage généralise l'interpolation linéaire classique en supposant une dépendance spatiale de l'erreur, i.e.

$$Y(s) = \mu(s) + \delta(s) \tag{2.4}$$

où μ représente la structure déterministe pour $E[Y]$ et δ une fonction aléatoire stationnaire, d'espérance nulle et de structure de dépendance supposée connue.

Plusieurs types de Krigeage existent en fonction de la forme de μ choisie, mais le plus commun et le plus simple est le Krigeage ordinaire : $\mu(s) = \mu$ avec μ de constante inconnue.

Enfin, la forme de δ est déterminée à l'aide de l'analyse variographique (présentée en annexe 5), puis (2.4) est réinjectée dans le systèmes d'équations (2.1), (2.2) et (2.3) pour obtenir le système d'équation final.

Pour l'implémentation en R, nous utilisons simplement la fonction autoKrige du paquet automap [2] (Hiemstra et *al.*, 2009), qui sélectionne automatiquement le meilleur ensemble d'hypothèses pour le modèle d'erreur (analyse variographique), puis détermine une approximation de Y sur l'espace convexe minimal recouvrant les données d'apprentissage.

Remarques
— Le Krigeage est une méthode d'interpolation exacte : $\forall s_i, \hat{Y}(s_i) = Y(s_i)$.
— Le Krigeage peut être global ou local : le choix du voisinage de s_0 peut se restreindre à une certaine distance, un certain nombre de voisins ou à l'inverse prendre en compte toutes les données disponibles. Pour des préoccupations de temps de calcul, nous décidons d'appliquer un Krigeage local – avec les cent voisins les plus proches conservés.

2.2.2.2 Modèle additif généralisé

Le modèle de régression linéaire prédit la variable $\hat{Y}$ en utilisant une combinaison linéaire des variables explicatives X_i tel que

$$\hat{Y} = \sum_i \beta_i X_i. \tag{2.5}$$

Ainsi, la variation des valeurs d'une variable explicative induit une variation proportionnelle pour la variable prédite. Elle ne convient donc pas pour tout type de distribution de variable à prédire. Elle est plutôt adaptée aux distributions normales.

Le modèle linéarisé généralisé (Generalized Linear Model, GLM) est une généralisation de la régression linéaire à d'autres distributions, en utilisant une fonction de liaison g telle que

$$g(\hat{Y}) = \sum_i \beta_i X_i. \tag{2.6}$$

De nombreuses fonctions de liaison ont été étudiées dans la littérature, notamment pour estimer les distributions habituelles (Poisson, Gamma, Multinomial...).

Le modèle additif généralisé (Generalized Additive Model, GAM) est une combinaison du GLM et du modèle additif (Addive Model, AM) qui généralise la régression linéaire aux phénomènes non linéaires. Le AM s'écrit

$$\hat{Y} = \sum_i f_i(X_i) \tag{2.7}$$

avec f_i une fonction lisse appartenant à une classe connue (e.g. spline, exponentiation).

2. `https://cran.r-project.org/web/packages/automap/automap.pdf`

Pour former un modèle AM, un algorithme de réadaptation alternant adaptation de la prédiction Y et adaptation de la variable f_i est généralement utilisé.

Finalement, le GAM s'écrit sous la forme

$$g(\hat{Y}) = \sum_i f_i(X_i). \tag{2.8}$$

En raison des conclusions de Mueller et *al.* (2016); Hasenfratz et *al.* (2015), nous choisissons la fonction logarithme en tant que fonction de liaison g et des splines d'ordre trois pour la classe des f_i.

Pour l'implémentation en R, nous utilisons le paquet mgcv[3] (Wood, 2017), comme Mueller et *al.* (2016).

En pratique, pour former notre modèle LUR, nous devons avoir plus de points de données que de paramètres du modèle (environ 40 dans notre cas). De plus, seules les variables explicatives continues bénéficient des fonctions lisses f_i (s(.,k=3) en R) ; les variables explicatives catégorielles sont traitées comme dans le GLM (simple pondération des catégories).

Enfin, pour prédire à partir du modèle, nous devons supprimer toutes les catégories non observées des variables explicatives catégorielles, i.e. toutes les catégories non présentes dans le jeu d'entraînement. En effet, si une variable catégorielle dispose d'une catégorie non apprise, le modèle ne saura pas comment se comporter avec elle. Nous ne la prenons donc pas en compte pour la prédiction.

2.2.2.3 Réseau de neurones artificiels

Un neurone artificiel est conceptuellement inspiré par un neurone biologique et son fonctionnement. Dans un réseau, un neurone transfère une sortie en fonction de ses entrées, pondérée par un poids synaptique évoluant au cours de l'apprentissage (plasticité synaptique), l'ensemble du réseau définissant une fonction des variables explicatives.

De nombreuses architectures de réseaux sont étudiées. Cependant, la plus robuste, documentée et mise en œuvre est sans aucun doute le perceptron multicouche (multilayer perceptron, MLP) (Jain et *al.*, 1996). De plus, Adams et Kanaroglou (2016) ont déjà choisi cette architecture pour étudier l'estimation de la pollution atmosphérique (PM_{10}, NO_2) à l'aide de mesures mobiles. Nous faisons les mêmes choix conceptuels, à savoir :

— un neurone d'entrée par variable explicative (cf. Tables 2.2),
— la fonction logistique en tant que fonction d'activation,
— une seule couche cachée, de taille indéterminée allant de 5 à 25 neurones.

L'indétermination de la couche cachée permet de considérer plusieurs réseaux de neurones différents et de sélectionner celui qui fournit les meilleurs résultats. Pour faire varier cet hyperparamètre (le nombre de neurones de la couche cachée), nous utilisons le package caret[4] (Kuhn, 2008). Ce package sert à faire varier le jeu d'hyperparamètres et sélectionner le meilleur, mais délègue la mise en œuvre du modèle d'apprentissage (ici, le MLP) à proprement parlé. Pour cela, nous utilisons le même package R que Bergmeir et Benítez

3. https://cran.r-project.org/web/packages/mgcv/index.html
4. https://cran.r-project.org/web/packages/caret/index.html

(2012), à savoir RSNNS [5].

En pratique, les réseaux de neurones ne gèrent pas les variables explicatives catégorielles. Il nous faut alors les binairiser i.e. les décomposer en autant de variables binaires que de catégories. Concrètement, après binarisation, nous avons une centaine de variables explicatives binaires. Les variables explicatives continues sont centrées-réduites pour leur donner le même poids a priori dans le réseau.

La variable prédite (la concentration du polluant) est également centrée-réduite. Cela permet d'obtenir de bien meilleurs résultats. Pour que la prédiction ait un sens, i.e. indique une concentration dans une gamme de valeurs réelles, nous effectuons l'opération inverse sur la prédiction. Pour cela, nous enregistrons la moyenne et la variance de la variable prédite du jeu d'entraînement.

Enfin, comme pour le GAM, nous devons supprimer toutes les catégories non observées pour la prédiction.

Un schéma de la forme du réseau de neurones entraîné est présenté en annexe 6.1.

2.3 Génération des observations synthétiques

2.3.1 Zone d'étude

Notre cas d'étude se focalise sur la prévision des PM_{10} et du NO_2 à l'échelle de la ville de Marseille. Le choix des PM_{10} et du NO_2 permet de tester les méthodes de spatialisation sur des polluants primaires mais ayant des durées de vie différentes (voir 1). Le choix de Marseille est totalement pragmatique. En effet, ATMO Sud qui est l'association agréée pour la surveillance de la qualité de l'air (AASQA) de la région Provence-Alpes-Côte d'Azur, a rapidement répondu à notre sollicitation concernant la mise à disposition de cartes à haute résolution de la pollution de l'air. Les demandes d'information auprès de ATMO Occitanie concernant la région de Toulouse n'ont jamais pu aboutir.

Marseille (43°17′47″ N, 5°22′12″ E) est la seconde plus grande ville de France en terme de population, avec plus de 860 000 habitants, et troisième en terme de taille, avec une surface d'environ 240 km^2. Elle est localisée dans le Sud de la France au bord de la mer Méditerranée. La moitié de sa surface est dans un territoire naturel non constructible et elle abrite le plus grand port de France et le deuxième en Méditerranée. Elle possède trois autoroutes principales, une au nord, une au Sud et une le long de la côte, ainsi que des tunnels. Elle dispose également de deux sites industriels classés SEVESO – une directive européenne identifiant les sites industriels présentant des risques d'accidents majeurs. L'aéroport est situé à 25 km à l'Ouest. Marseille est la seconde ville la plus embouteillée de France après Paris. Elle se situe au 53^{e} rang mondial des villes congestionnées par le trafic routier.

Notre zone d'étude est un rectangle focalisé sur le centre-ville allant du (43°14′9″ N, 5°20′42″ E) au (43°21′0″ N, 5°28′48″ E) et couvre une superficie de 140 km^2.

5. https://cran.r-project.org/web/packages/RSNNS/index.html

2.3.2 Extraction des variables explicatives de la ville depuis OSM

La difficulté d'acquisition des variables explicatives pour une zone d'étude conduit à une hétérogénéité des bases de données (Hoek et *al.*, 2008) et à une faible résolution spatio-temporelle (Marjovi et *al.*, 2015; Hoek et *al.*, 2008). Nous choisissons OpenStreetMap [6] pour extraire les variables explicatives de la zone d'étude parce que c'est le principal service de cartographie collaboratif libre de droits. Ainsi, nous choisissons d'opter ce service qui assure comparabilité et reproductibilité des études, au détriment de variables explicatives dynamiques, c'est-à-dire qui varient dans le temps.

Un fichier OSM représente une région du monde. Il est composé d'objets élémentaires d'un modèle conceptuel du monde physique : des nœuds, des voies et des relations. Les nœuds sont définis par leur longitude et leur latitude et peuvent représenter des points d'intérêt ou servir à la définition de voies. Les voies représentent des routes ou des zones d'espace. Les relations sont utilisées pour relier des objets entre eux. Tout objet peut être enrichi d'informations spécifiques, à l'aide d'attributs clé-valeur.

OSM a son propre format de données conçu pour faciliter la contribution gratuite des utilisateurs. Néanmoins ce format de données n'est pas toujours simple à utiliser. Nous avons donc eu recours à la base de données de GeoFabrik qui propose un ensemble de graphes (shapefiles) pré-traités [7]. Nous y avons adjoint la couche de topographie de l'institut national de l'information géographique et forestière (IGN) à une résolution de 75 mètres, car il s'agit d'une information disponible dans OSM mais non fournie par GeoFabrik.

Nous reformons le jeu de données de manière à ne pas avoir trop de catégories différentes pour une même variable explicative ou pas assez de données pour une catégorie. Par exemple, les nombreuses catégories de la variable explicative « points d'intérêt » (supermarché, restaurant, hôpital, pharmacie...) sont regroupées en 9 classes (shopping, loisirs, santé...).

De plus, certains shapefiles fournis par GeoFabrik comportent des données manquantes. En certaines zone de l'espace, aucune information n'est disponible. Par exemple, une habitation n'étant pas un point d'intérêt, il n'y aura pas d'information sur ce shapefile à cette position. En outre, tous les points d'intérêt ne sont pas forcement répertoriés dans OSM. Les méthodes utilisées (Krigeage, GAM et MLP) ne gèrent pas les données manquantes ; ils attendent des variables explicatives définies sur toute la zone d'étude. Nous introduisons alors une catégorie « NA » pour les données manquantes d'une variable explicative catégorielle ; pour une variable explicative continue, nous remplaçons les données manquantes par la médiane des valeurs de cette variable.

Nous calculons aussi de nouvelles variables explicatives telles que la distance la plus proche à la route principale ou la densité de bâtiments. Enfin, les shapefiles sont projetés dans le système de référence de coordonnées EPSG 2154, approximativement cartésien pour la France, puis centrés-réduits et rasterisés à une résolution de 25 mètres. Pour la rasterisation, nous utilisons la fonction rasterize du package raster [8] en R. Pour les variables

6. https://www.openstreetmap.org/
7. https://download.geofabrik.de/osm-data-in-gis-formats-free.pdf
8. https://cran.r-project.org/package=raster

catégorielles, si plusieurs catégories sont présentes pour une même cellule du raster, c'est la dernière rencontrée par l'algorithme qui est conservée. Ainsi, certaines informations disparaissent. Par exemple, lorsque deux voies de type différents se croisent (prenons une piste cyclable et une autoroute, à deux altitudes différentes), un seul type de voies est conservé à la cellule du raster correspondant au croisement et une discontinuité pour l'une des deux voies est observée. Cette solution n'est donc pas optimale mais réduit considérablement le temps de mise en œuvre et le temps de calcul. Ce format correspond à la résolution spatiale de nos données de pollution simulées (cf. ci-dessous).

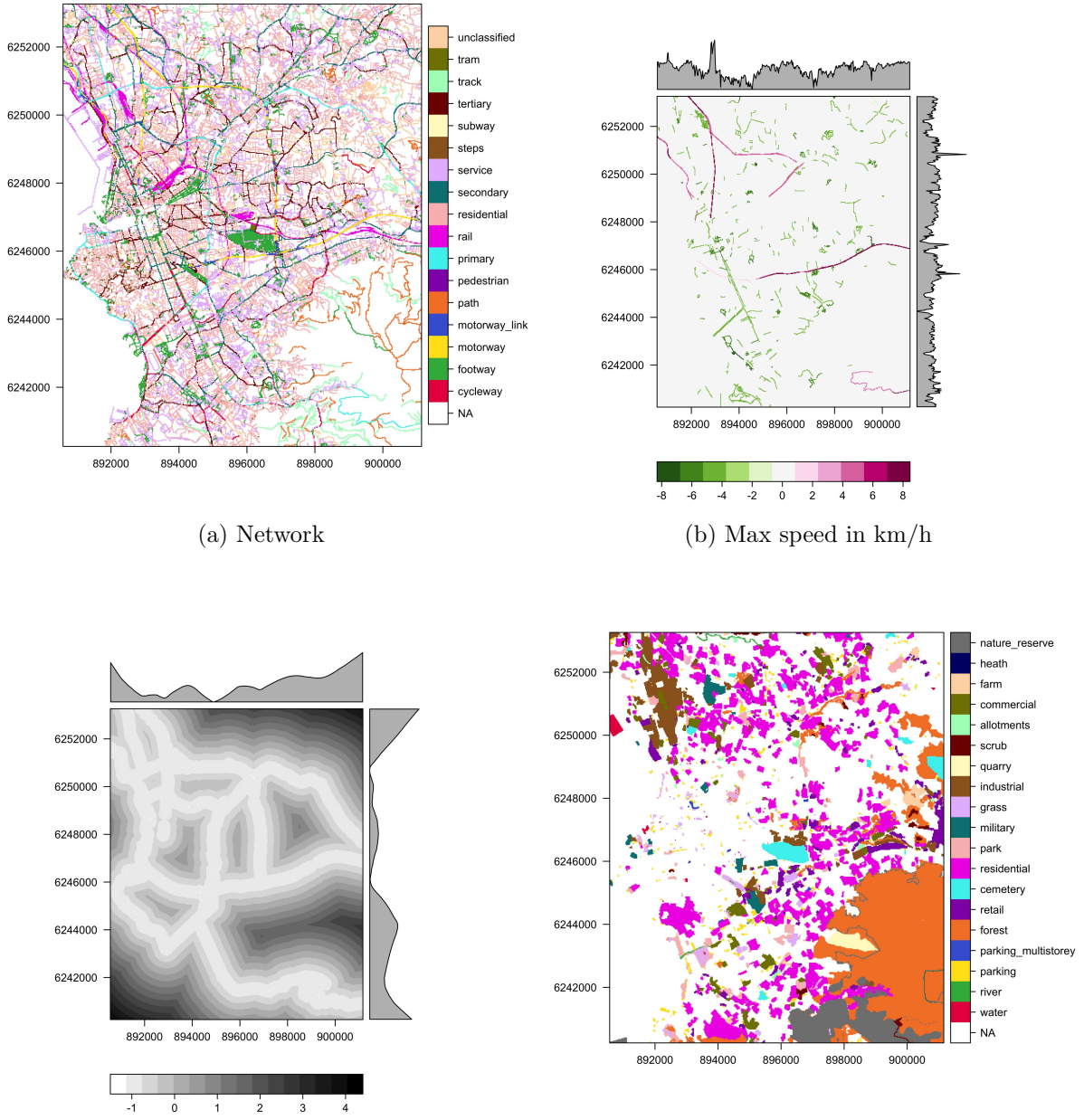
Les variables explicatives extraites sont regroupées dans la Table 2.2.

Table 2.2 – Description des variables explicatives continues ou catégorielles extraites de la base GeoFabrik et de l'IGN.

Variables continues		
Variable explicative	Description	Note
Position	Coordonnées spatiales	
Altitude	Altitude par rapport au niveau de la mer (base IGN)	IGN
Distance to main roads	Distance aux routes annotées 'motorway', 'trunk' ou 'primary' dans OSM	Calculée à partir roads
Buildings_a	Densité de bâtiments	Calculée à partir buildings_a
Maximum Speed	Limitations de vitesse	Extraites de roads
Variables catégorielles		
Network	Réseau routier et ferroviaire	Fusion de roads et railways
Transport	Infrastructures de transport (arrêt de bus, terminal de ferry, centrale de taxis...)	
Landuse_a	Utilisation des terres	Fusion de water_a, transport_a, traffic_a, natural_a et landuse_a
Traffic	Informations du réseau routier (feux de circulation, signalisation...)	
POIs	Points d'intérêt de la ville classés par classes principales (points)	Fusion de pois et pofw
POIs_a	Points d'intérêt de la ville classés par classes principales (surface)	Fusion de pois_a et pofw_a
Tree	Présence d'un ou plusieurs arbres dans la ville	Calculée à partir de natural

Le rendu graphique des couches utilisées est présenté aux Figures 2.3 et 2.4. La ville est asymétrique avec le bord de mer à l'Ouest ; cernée par des massifs (Figure 17d). Une zone boisée se trouve au Sud-Est. Nous identifions aisément la vieille ville correspondant à la plus grande densité de bâtiments (Figure 17c) avec un accès sur le port. La carte d'utilisation des sols donne des informations essentiellement sur les zones cultivées et de forêts (Figure 16d) mais rien en centre-ville. Les axes rapides de circulation (Figure 16b

et 16a) sont les pénétrantes A50 à l'Ouest, A7 et A55 au nord et la D559 au Sud. La distance aux principaux axes de circulation (cf. Table 2.2) est présentée Figure 16c.



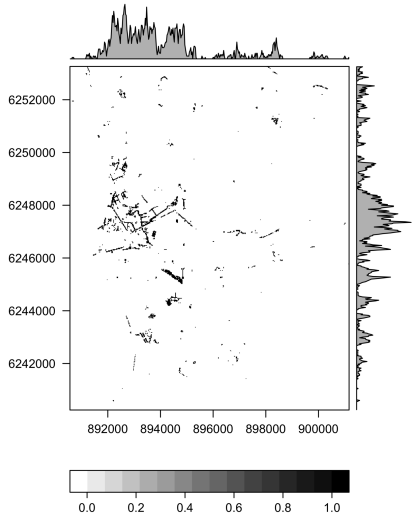
(a) Network

(b) Max speed in km/h

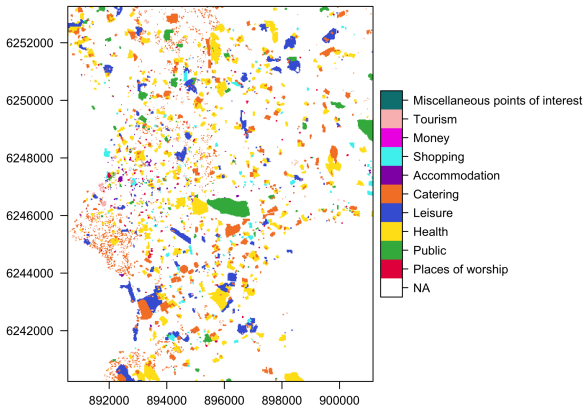
(c) Distance to main road

(d) Landuse

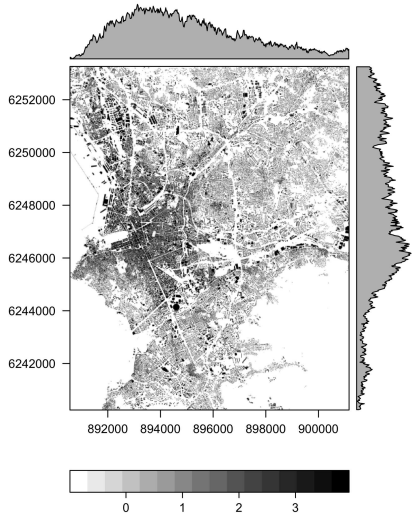
FIGURE 2.3 – Variables explicatives extraites de la base de données de OpenStreetMap et pré-traitées représentant (a) le réseau routier, (b) la limite de vitesse autorisée, (c) la distance aux principaux axes de circulation, et (d) la carte d'occupation du sol pour la ville de Marseille.



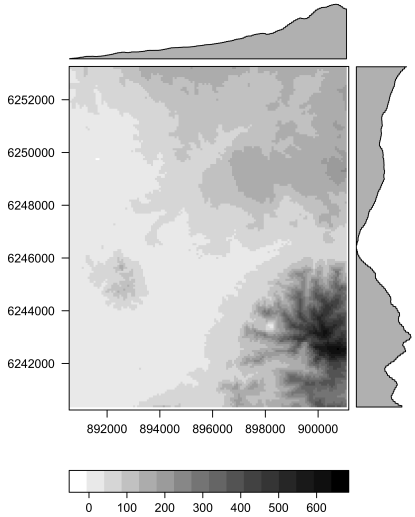
(a) Tree



(b) Point of interest Area



(c) Building Area



(d) Altitude in meters

FIGURE 2.4 – Même légende que Figure 2.3 pour (a) la présence d'arbres, (b) les zone de points d'intérêt, (c) la densité de construction et (d) l'altitude du MNT (issue de la base de données de l'IGN).

2.3.3 Simulation des trajets à vélo

Puisque nous neutralisons la composante temporelle dans cette étude, nous n'avons pas besoin de simuler un modèle de mobilité ni les schémas récurrents de déplacements. Il suffit de sélectionner au hasard des trajets fictifs de cyclistes dans la zone d'étude, sans fournir aucune information sur la chronologie, et de vérifier la cohérence spatiale du jeu de trajets simulés.

Les trajets sont simulés en définissant des emplacements de départ et de destination. Nous choisissons les lieux de départ autour des principaux points attractifs de la ville, à savoir les plages et le vieux port. Les points de départ sont choisis aléatoirement selon une densité gaussienne centrée sur ces points attractifs. La Figure 2.5 présente la carte de la ville, la distribution de la densité des emplacements de départ et les emplacements de départ réellement générés. Les lieux de destination sont sélectionnés en fonction d'une distance à vol d'oiseau et d'un cap, eux-mêmes sélectionnés de manière aléatoire. Le cap est choisi de manière uniforme entre 0° et 360° et la distance à vol d'oiseau, nécessairement inférieure à la longueur réelle du trajet, suit une répartition normale centrée sur 2,45 km d'écart-type 4,5 km.

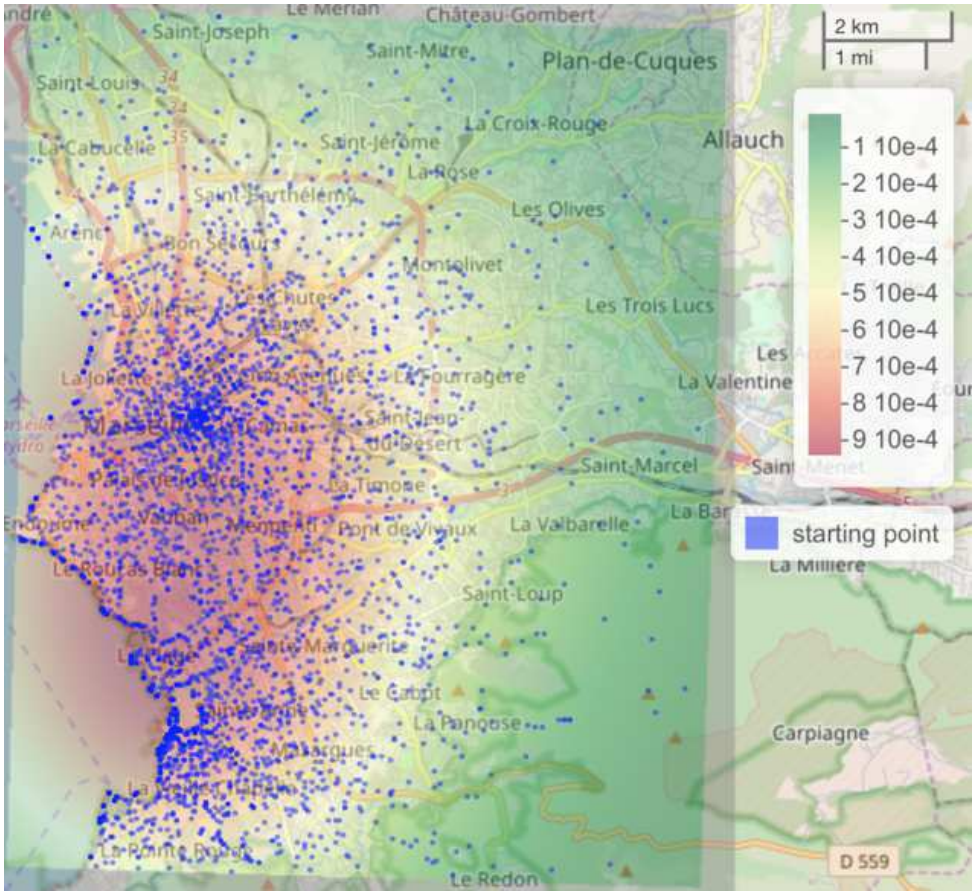


FIGURE 2.5 – Points de départs générés et distribution spatiale de génération.

Afin d'obtenir un parcours réaliste, nous faisons appel à une interface de programmation applicative (API) de planification d'itinéraires cyclistes, nommée BBBike[9]. Cette API utilise OSM pour le calcul des parcours et retourne une série de coordonnées GPS pour chaque changement de direction, parfois annotées par le nom de la rue.

Le nombre maximal de trajets par jour est fixé à 4500. Ce nombre correspond approximativement à la fraction estimée des travailleurs qui font quotidiennement le trajet

9. `https://www.bbbike.org`

maison–travail–maison (1,3 % [10] des 363 939 travailleurs [11] à Marseille en 2015).

A la Figure 2.6 nous montrons une comparaison entre la distance moyenne des trajets de notre simulation et celle évaluée par l'institut français de la statistique et des études économiques (INSEE) pour la France. Nous remarquons une sous-estimation des petits trajets (< 1 km) et une sur-estimation des trajets intermédiaires (entre 6 km et 8 km). Ce décalage dans les distributions correspond au choix fait de simuler des trajets domicile–travail.

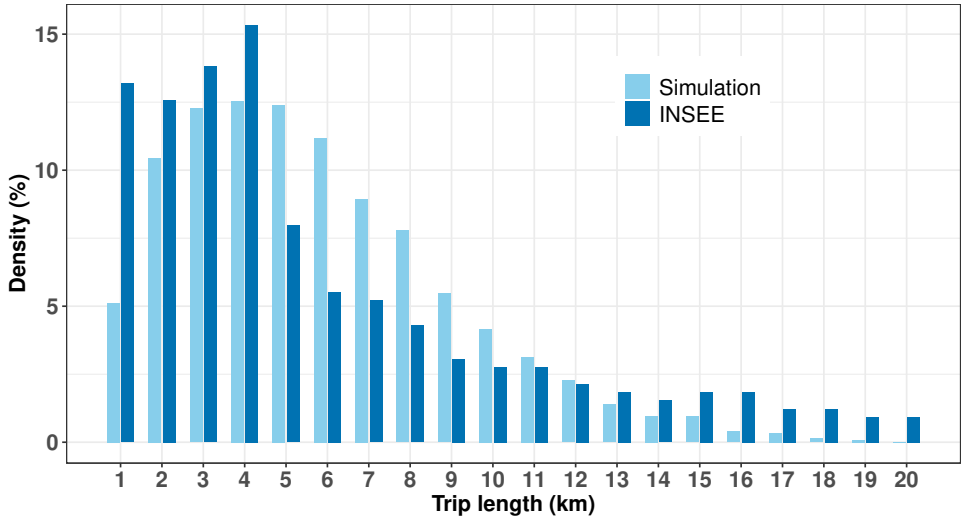


FIGURE 2.6 – Histogramme (à une résolution de 1 km) de la longueur des trajets à vélo pour les 4500 trajets simulés sur Marseille et pour l'ensemble des français selon les statistiques de l'INSEE.

La Figure 2.7 représente la couverture spatiale des trajets à vélo ainsi simulés pour un ensemble maximal de 4500 vélos. Nous pouvons observer une couverture homogène de la ville par les trajets en raison du choix aléatoire de la direction de déplacement.

10. https://www.insee.fr/fr/statistiques/2553852
11. https://www.insee.fr/fr/statistiques/2011101?geo=COM-13055

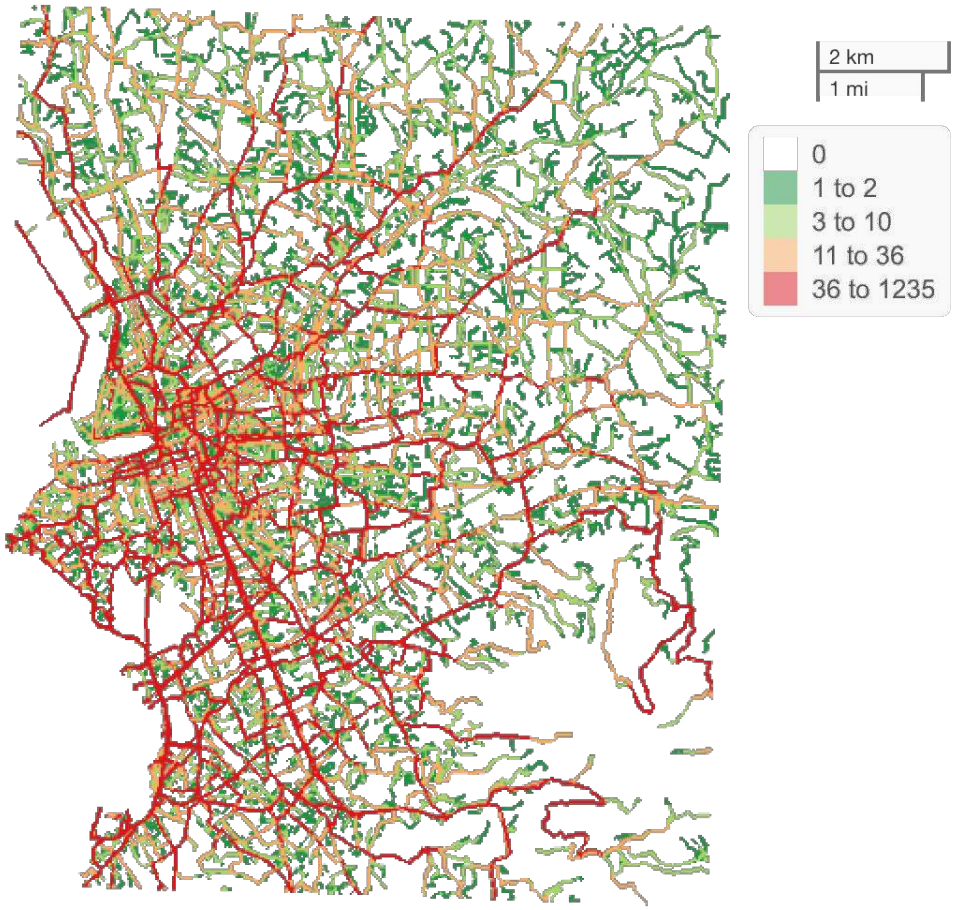


FIGURE 2.7 – Carte des fréquences de passages de nos vélos simulés. Résolution spatiale 25 m × 25 m.

Afin d'estimer le réalisme des lieux visités, nous comparons le nombre de passages des vélos simulés et des vélos réels en certains endroits, sur la base des données fournies par l'association « Vélo en Ville [12] » . Cette association a organisé en 2017 et 2018 des campagnes de comptage de vélos en certains points de la ville de Marseille durant les périodes de pointe, entre 8h et 9h et de 16h30 à 18h30. La Table 2.3 comptabilise le nombre de vélos aux différentes intersections lors des enquêtes de 2017 et 2018. Nous constatons un biais entre notre simulation et le comptage, en partie dû à la période relative courte du comptage. Le nombre de trajets est surestimé d'un facteur 1,2 (Michelet/vélodrome) à 5 (Corniche/hélice). Le décalage semble être plus important pour les sites situés sur les axes de circulation loin du centre-ville. Cependant il existe une corrélation significative entre le comptage et la simulation (coefficient Pearson R=0,72) indiquant que la simulation est assez réaliste en terme de couverture.

12. `http://www.velosenville.org`

TABLE 2.3 – Comptages du nombre de vélos à différentes intersections à Marseille, issus des observations réelles fournies par « Vélo en ville » et de notre simulation.

Lieu de comptage	2017	2018	Simulation
Baille/Lodi	294	-	764
National/Guibal	175	-	487
Prado/Castellane	639	727	1297
Chave/Eugène Pierre	88	-	332
Joliette/République	178	-	623
Rome/Saint Louis	388	-	214
Vieux Port/Canebière	522	667	965
République/Sadi Carnot	262	-	416
Corniche/hélice	157	-	819
Michelet/vélodrome	438	506	638

Le type de voies empruntées par les simulations sont représentés à la Figure 2.8. Nous remarquons que certains types de voies ne sont logiquement pas échantillonnés (*rail*, *subway*, *tram*). Le type *motorway* est échantillonné, mais de manière marginale. La comparaison de notre simulation avec les catégories pour l'ensemble de la carte montre que les catégories *cycleway* (pistes cyclables) et *tertiary* (routes inter-quartiers et routes reliant la banlieue et la ville) sont naturellement sur-représentées par rapport à l'ensemble de la carte. Ainsi, notre plateforme de transport simulée semble suivre les contraintes de déplacement d'une flotte de vélos circulant à travers la ville.

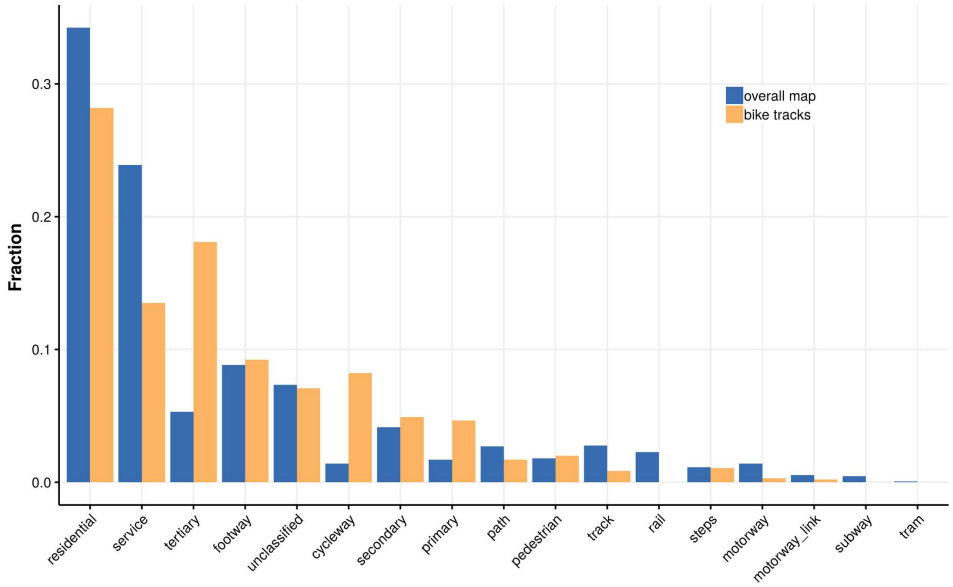


FIGURE 2.8 – Distribution des types de voies pour toute la zone d'étude et pour nos 4500 trajets simulés.

2.3.4 Observations synthétiques à partir d'un modèle numérique de qualité de l'air

Pour simuler les observations qui peuvent être faites par les systèmes mobiles tout en se concentrant sur le problème d'estimation spatiale, nous supposons que nos capteurs sont idéaux (tous identiques, sans bruit, sans biais, sans retard) : à période d'échantillonnage spatial fixe, nous attribuons des valeurs de mesure le long du trajet conformément à une carte de référence de la pollution, supposée décrire parfaitement l'environnement. Afin d'avoir une description aussi réaliste que possible de la répartition spatiale de la pollution atmosphérique à l'échelle de la ville, nous utilisons des analyses fournies par l'AASQA ATMO Sud. Ces analyses sont basées sur des simulations numériques à différentes échelles impliquant le modèle CHIMERE (voir chapitre 1), le modèle ADMS-URBAN [13], et les mesures fournies par les stations de référence (SSN). Les cartes fournies correspondent à la valeur maximale journalière en NO_2 et PM_{10}. La résolution spatiale des cartes est de 25 m × 25 m. Nous utilisons un jeu de 14 cartes journalières, du 12 au 15 février 2018.

La Figure 2.9 présente un exemple de trajet pour la journée du 14 février 2018. Le trajet est représenté sur la carte présentant la simulation de la concentration de NO_2 pour cette journée. Le vélo simulé se dirige de l'est vers l'Ouest en direction de la côte. Les concentrations de NO_2 simulées vont de 20 μg/m^3 à 100 μg/m^3 sur une distance de 6 km. Nous observons une augmentation de la concentration lors de la traversée du centre-ville sur le type de voies *tertiary* puis une décroissance lente quand le vélo se déplace vers l'Ouest. Les valeurs les plus faibles sont observées dans le quartier résidentiel et sur les zones identifiées comme piétonnes (*footway*).

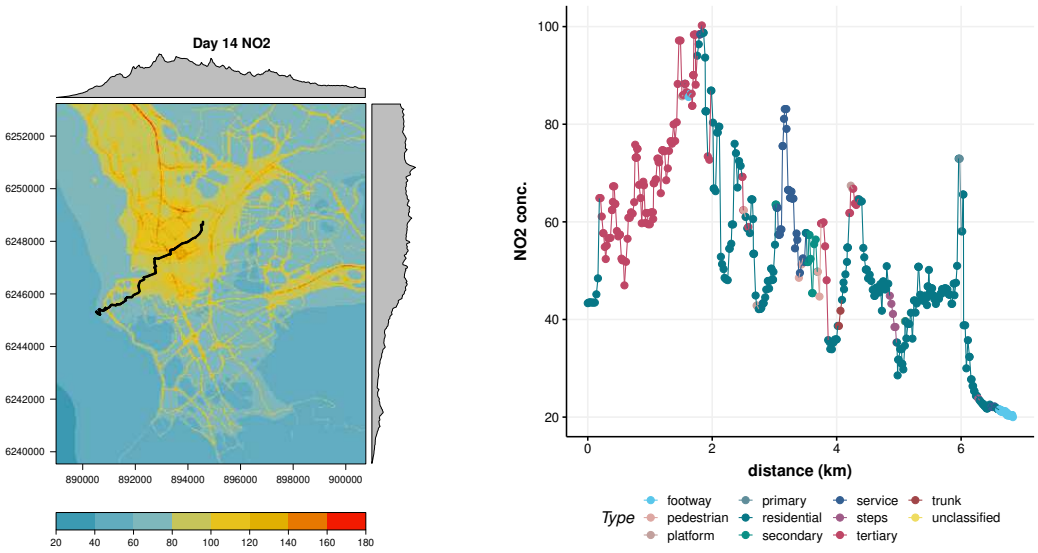


FIGURE 2.9 – Exemple de trajet de 6 km dans la ville de Marseille le 14 février (à gauche) carte de NO_2 et (à droite) évolution du NO_2 en fonction de la distance parcourue et du type de voies.

Comme nous avons pu le voir à la Figure 2.8, les vélos échantillonnent correctement

13. https://www.cerc.co.uk/environmental-software/ADMS-Urban-model.html

l'ensemble des différentes structures de la ville. Nous retrouvons également une certaine variabilité des concentrations lors des trajets. Les concentrations vont en moyenne de 12 à 111 pour le NO_2 et de 10 à 37 pour les PM_{10}. Les concentrations ont une variabilité spatiale qui est en relation avec le type de voies. La Figure 2.10 présente la statistique (boxplot) des valeurs observées en polluants pour l'ensemble des trajets simulés (4500). Nous remarquons que les concentrations de polluants sont plus élevées sur les axes circulant (e.g. *motorway*, *primary*). L'amplitude est d'environ un facteur 2 entre les zones les plus polluées et les zones les moins polluées pour les PM_{10} et d'un facteur 4 pour le NO_2. La statistique pour cette sélection est similaire à celle de la carte entière (hormis les types qui ne sont pas échantillonnés) ce qui indique que notre échantillonnage n'est pas biaisé en ce qui concerne les valeurs de polluants.

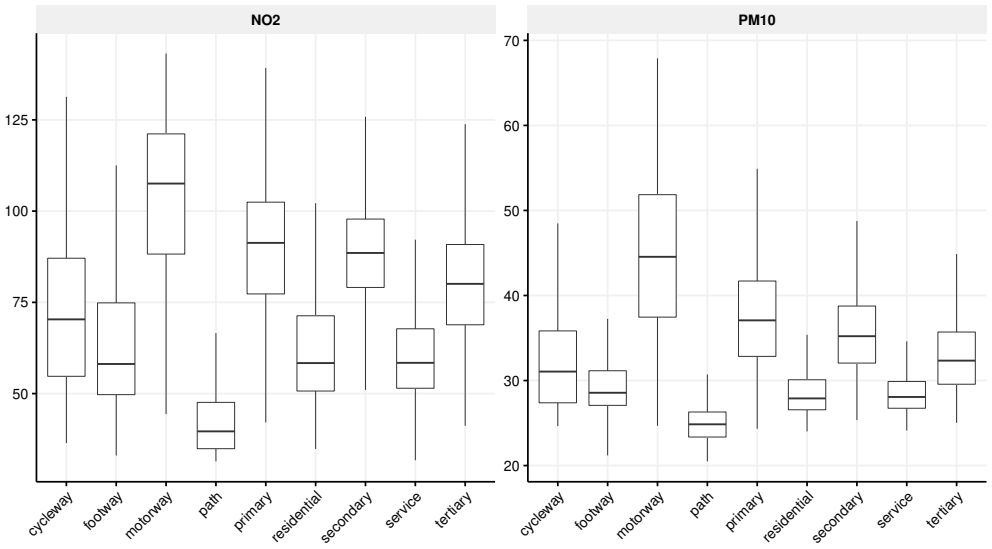


FIGURE 2.10 – Boxplot des concentrations en NO_2 et PM_{10} en fonction du type de voies pour un ensemble de 4500 trajets.

2.4 Spatialisation des observations mobiles

Les prédictions issues des méthodes de spatialisation dépendent du jeu d'observations fourni. Or ce jeu de mesures dépend du nombre de trajets et de la distance d'échantillonnage. Nous souhaitons minimiser la taille du jeu de mesures de pollution géolocalisées à collecter tout en préservant la qualité de la prédiction.

De manière à tester l'impact de la taille du réseau de capteurs mobiles et de la fréquence à laquelle sont faites les mesures, nous effectuons une simulation de Monte-Carlo pour un nombre variable de trajets avec une distance d'échantillonnage variable. Nous faisons varier d'abord le nombre de trajets puis l'échantillonnage. Pour chaque couple (nombre de trajet, échantillonnage), nous considérons 280 évènements, supposés indépendants et identiquement distribués (i.i.d.), obtenus en synthétisant 20 jeux de trajets différents par jour pour les 14 jours. Les cartes obtenues à partir des méthodes de spatialisation sont comparées avec les cartes de référence.

Nous restreignons notre zone d'étude à l'enveloppe convexe formée par les 4500 trajets synthétisés, notée par la suite C_{4500}. Cela permet de minimiser les zones inaccessibles aux vélos (mer, parc naturel...) et donc difficilement prédictibles.

Pour évaluer la qualité de la carte prédite, nous comparons la valeur prédite à la valeur de la carte de référence, en chaque point de C_{4500} privée des positions du jeu de mesures (qui ont servi à l'entraînement du modèle), à l'aide de métriques classiques pour évaluer les modèles de régression :

- l'erreur quadratique moyenne, $RMSE_Z(X,Y) = \sqrt{\frac{\sum_{i\in Z}(X_i - Y_i)^2}{card(Z)}}$ avec $card$ la fonction cardinale sur l'ensemble Z,
- l'erreur absolue moyenne, $MAE_Z(X,Y) = \frac{\sum_{i\in Z}|X_i - Y_i|}{card(Z)}$
- le coefficient de corrélation de Pearson, $cor_Z(X,Y) = \frac{Cov_Z(X,Y)}{\sigma_Z(X)\sigma_Z(Y)}$ avec Cov_Z la covariance et σ_Z l'écart-type dans Z.

La moyenne et l'écart-type des métriques pour les événements de notre simulation de Monte-Carlo informent respectivement sur la précision et la confiance à apporter à la prédiction. En effet, nous pouvons considérer que le minimum de la moyenne de l'erreur correspond à l'erreur systématique de la méthode de spatialisation et l'écart-type de l'erreur correspond à l'incertitude liée à la réalisation du tirage aléatoire des trajets (dont souffre également la réalisation d'un jeu de données réel). Pour une meilleure lisibilité, nous choisissons de représenter l'écart-type relatif – i.e. l'écart-type divisé par la moyenne – au lieu de l'écart-type.

Nous cherchons alors à déterminer des points d'inflexion à partir desquels de plus grands jeux de données améliorent peu la qualité de la prédiction.

Formellement reformulé, avec les notations J l'ensemble des jours étudiés ($card(J) = 14$), R_j la carte de référence associée au jour j, $T_{n,s}$ les jeux de n trajets synthétisés échantillonnés tous les s mètres ($card(T_{n,s}) = 20$) et $P_{j,t,ms}$ la carte prédite par la méthode de spatialisation ms avec les mesures du jeu de trajets t au jour j ; nous étudions pour un polluant donné (NO_2 ou PM_{10}) :

$$erreur_{Monte-Carlo}(n,s,ms) = mean_{j\in J\times t\in T_{n,s}}(metric_{C_{4500}\setminus t}(\text{Préd}_{j,t,ms} - \text{Réf}_j))$$

et

$$incertitude_{Monte-Carlo}(n,s,ms) = (\frac{sd}{mean})_{j\in J\times t\in T_{n,s}}(metric_{C_{4500}\setminus t}(\text{Préd}_{j,t,ms} - \text{Réf}_j))$$

en faisant varier n puis s pour $(ms, metric) \in \{Krigeage, GAM, MLP\} \times \{RMSE, MAE, cor\}$.

En outre, pour évaluer l'impact du nombre de trajets et de l'échantillonnage sur le jeu de mesures, nous le caractérisons au travers de trois taux de couverture. Nous définissons le taux de couverture d'un jeu de trajets échantillonnés $t \in T_{n,s}$ pour une variable V comme la proportion entre les valeurs observées et les valeurs observables dans C_{4500}. Concrètement, pour une variable catégorielle V, il s'agit de la proportion du nombre de catégories rencontrées par rapport au nombre de catégories qui peuvent être rencontrées : $\frac{card_t(V)}{card_{C_{4500}}(V)}$. Pour une variable continue, il s'agit du rapport entre l'étendue [14] des valeurs

14. Différence entre la valeur maximale et la valeur minimale.

observées par rapport à l'étendue des valeurs observables : $\frac{max_t(V)-min_t(V)}{max_{C_{4500}}(V)-min_{C_{4500}}(V)}$.

Les variables V considérées sont chacune des colonnes du jeu de données : les variables explicatives et les mesures du polluant considéré. Néanmoins, nous écartons les variables explicatives Position_x et Position_y au profit de la couverture spatiale, décrite par le nombre de cellules visitées du raster, qui est davantage indépendante du référentiel.

2.4.1 Sensibilité au nombre de trajets

Nous faisons varier le nombre de trajets dans notre simulation de Monte-Carlo à une période d'échantillonnage fixe de 100 mètres. Les nombres de trajets choisis (2, 4, 6, 8, 10, 40, 80, 120, 250, 500, 1500, 3000, 4500) sont resserrés vers les petites valeurs d'une part, car plus réalistes et rapidement calculables et d'autre part, car les prédictions varient plus fortement pour celles-ci.

Nous représentons les variations de l'$erreur_{Monte-Carlo}(., 100, .)$ pour chaque métrique (RMSE, MAE et cor) et chaque polluant (NO_2 et PM_{10}) et de l'$incertitude_{Monte-Carlo}(., 100, .)$ pour la RMSE et le NO_2 (car les tendances sont identiques) en fonction du nombre de trajets pour les trois méthodes de spatialisation sélectionnées. Au total, nous prédisons donc $13 * 2 * 3 * card(J \times T_{n,s}) = 21\,840$ cartes.

Nous observons que le réseau de neurones a une erreur relativement constante quelque soit le nombre de trajets. Cela est dû au fait que la carte est prédite sur des valeurs centrées-réduites et que nous avons manuellement biaisé et dispersé les cartes selon les mêmes biais et écart-type que ceux du jeu d'entrée. Cette opération n'est pas satisfaisante.

Nous représentons alors les variations des métriques pour les cartes centrées-réduites afin de pouvoir comparer les trois méthodes. La Figure 2.11 représente ces variations. Pour une meilleure lisibilité, l'échelle des abscisses des courbes est logarithmique.

Nous remarquons que, au regard du coefficient de corrélation de Pearson, les prévisions sont meilleures pour le NO_2 que pour les PM_{10}. Pour le LUR et le réseau de neurones (les deux modèles utilisant des variables explicatives de la ville), ce coefficient semble suivre la même tendance et stagner pour le NO_2 à partir de 80 trajets.

Les tendances de la RMSE et de la MAE étant similaires, nous nous concentrons sur la RMSE. Quelque soit la méthode de spatialisation, les prévisions sont meilleures pour le NO_2 que pour les PM_{10}. De plus, pour le NO_2, l'erreur décroît rapidement en fonction du nombre de trajets. Pour les deux modèles utilisant des variables explicatives, elle stagne dès l'ordre de la dizaine de trajets. Pour le Krigeage, elle décroît de façon quasi-logarithmique. Ceci s'explique par le fait que le Krigeage tire profit localement de toute information supplémentaire alors que le LUR et le réseau de neurones tentent de construire un modèle global en fonction des variables explicatives.

En outre, le Krigeage ne prédit que dans l'enveloppe convexe des trajets, donc une zone de l'espace généralement plus réduite que C_{4500}. L'erreur sur C_{4500} serait donc plus forte pour les faibles nombres de trajets.

Ainsi, les méthodes de spatialisation utilisant des variables explicatives obtiennent de meilleurs résultats que le Krigeage pour peu de trajets, et la tendance s'inverse ensuite, à partir d'une centaine de trajets pour le NO_2 et à partir de 500 trajets pour les PM_{10}.

Les résultats du LUR et surtout du réseau de neurones pourraient être encore améliorés. En effet, les variables catégorielles contiennent environ 10 catégories ou plus et

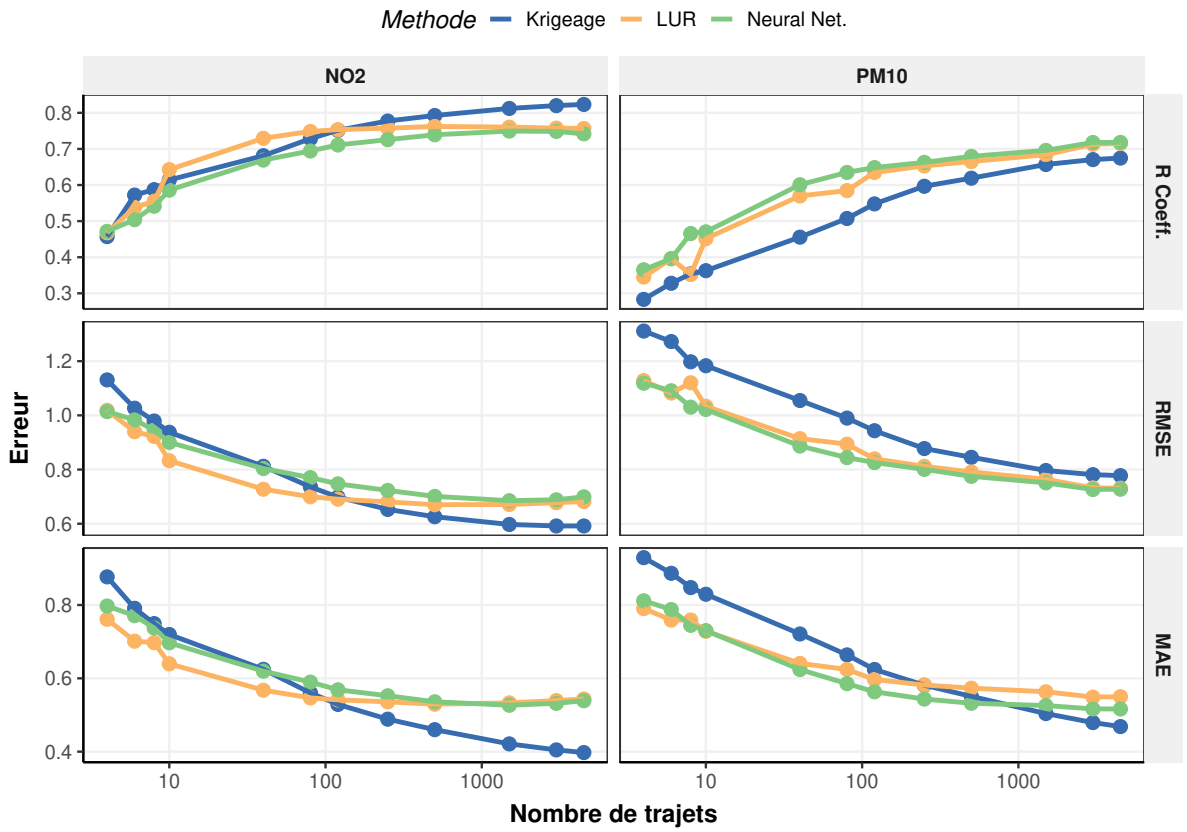


FIGURE 2.11 – Comparaison entre la carte de référence et la carte prédite en fonction du nombre de trajets, à un échantillonnage tous les 100 mètres. Échelle logarithmique.

certaines zones de la ville ne sont pas décrites par celle-ci. Chaque catégorie peut donc être rencontrée, dans le meilleur des cas, sur 10 % de la zone d'étude. Pour le réseau de neurones, chaque neurone de la couche de binarisation ne peut être activée qu'environ 10 % du temps. Il aurait donc été plus judicieux d'opérer une sélection des catégories avant application des méthodes pour s'assurer de leur représentativité.

Pour sa part, le Krigeage a l'avantage d'être rapide en temps de calcul mais diffuse spatialement la pollution estimée. Cette diffusion nivelle la concentration et sous-évalue les pics mais permet d'observer la tendance de fond locale. Concernant le LUR, les variations locales sont plus marquées car directement dépendantes des variables explicatives et l'étendue des valeurs prédites est représentative. Les variations des prédictions par le réseau de neurones semblent représentatives, mais pas l'étendue des valeurs.

En conclusion, le LUR semble plus adapté à notre cas d'utilisation : un jeu de données produit à partir de quelques dizaines de trajets.

De plus, puisque nos trajets sont issus d'une base de 4500 trajets, pour plus de $4500/card(T_{.,100}) = 225$ trajets, les 20 jeux de trajets générés possèdent des trajets en commun. Pour moins de 225 trajets, nous nous sommes assurés que ce n'est pas le cas. Nous n'étudions alors l'incertitude des prédictions que pour les nombres de trajets inférieurs à 225.

Un point d'inflexion se distingue nettement à partir d'une quarantaine de trajets pour les trois modèles. Puisque les tendances entre le NO_2 et les PM_{10} sont strictement identiques, nous ne représentons que celle du NO_2, Figure 2.12.

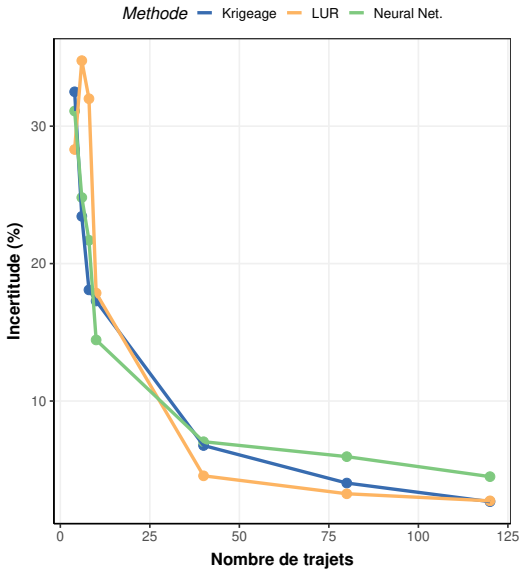


FIGURE 2.12 – Écart-type relatif des simulations pour le coefficient de corrélation en fonction du nombre de trajets.

Ainsi, nous en déduisons qu'une quarantaine de trajets est suffisante pour une prédiction robuste et de précision acceptable au regard de la meilleure précision accessible. C'est pourquoi, le LUR semble être une méthode à privilégier.

Les taux de couverture du jeu de données d'entrée sont tracés en annexe 6.2. Nous y représentons ce palier de 40 trajets par une barre verticale rouge. Nous observons que ces taux de couverture seuls ne sont pas déterminants pour la qualité de la prédiction. En particulier, la couverture spatiale n'est ni représentative de l'erreur ni de l'incertitude de prédiction. Nous en déduisons que la localité des mesures est plus déterminante que le nombre et que la notion de données linéiques (les trajets) importe plus que la notion de données ponctuelles.

2.4.2 Sensibilité à la fréquence d'échantillonnage

Nous faisons varier l'échantillonnage dans notre simulation pour le nombre de trajets fixé précédemment : 40. Pour rappel, la période d'échantillonnage est définie en mètre car aucune temporalité n'est associée aux mesures. Elle correspond à la distance entre deux mesures le long d'un trajet. Les périodes choisies sont 25, 50, 75, 100, 200, 300, 400, 500 et 1000 mètres. Pour un vélo à 15 km/h, leurs équivalences en temps sont 6, 12, 18, 24, 48, 72, 96, 120 et 240 secondes. Pour une meilleure lisibilité, l'échelle des abscisses des courbes présentées est logarithmique.

La Figure 2.13 représente les variations de l'$erreur_{Monte-Carlo}(40, ., .)$ pour chaque métrique (RMSE, MAE et cor) et chaque polluant (NO_2 et PM_{10}) en fonction de la

distance d'échantillonnage. Les cartes étudiées sont à nouveau les cartes centrées-réduites. Au total, nous prédisons $9 * 2 * 3 * card(J \times T_{n,s}) = 15\,120$ cartes.

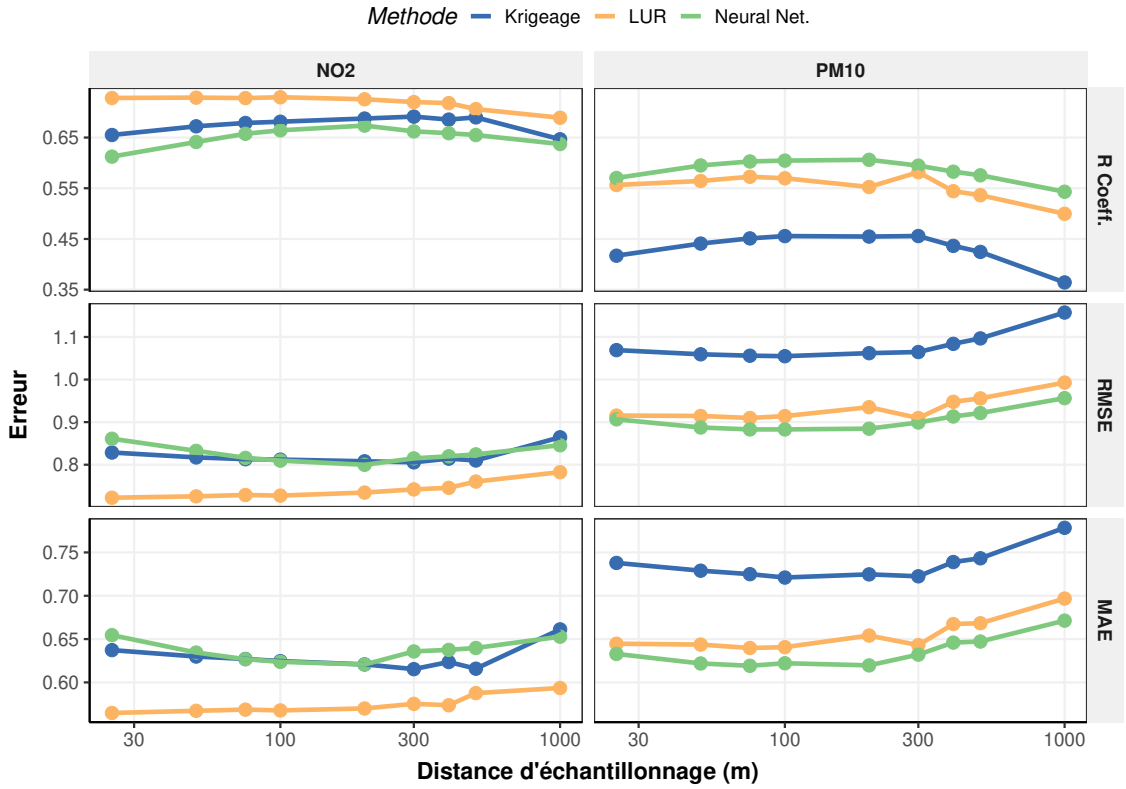


FIGURE 2.13 – Erreur de simulation en fonction de la distance d'échantillonnage. Le nombre de trajets est fixé à 40. Échelle logarithmique.

Premièrement, bien que les résultats soient meilleurs pour le NO_2 que pour les PM_{10}, les tendances des erreurs sont les mêmes pour les deux polluants et ce pour les trois métriques.

Deuxièmement, les variations de l'erreur ne sont pas flagrantes. Elles sont davantage marquées pour les PM_{10}.

Troisièmement, le LUR se distingue du Krigeage et du réseau de neurones. Pour ces deux derniers, nous observons que l'erreur diminue avec la distance d'échantillonnage entre 25 et 100 mètres puis augmente progressivement après 200 mètres.

La diminution de l'erreur entre 25 et 100 mètres de distance d'échantillonnage peut paraître surprenante. En supposant que ces modèles reposent fortement sur les données locales pour prédire, ceci s'explique facilement. En effet, nous supprimons du jeu de données de test les données d'entraînement, qui auraient été bien prédites pour de faibles distances d'échantillonnage. Cette hypothèse est confirmée pour le Krigeage de par son principe. Pour le réseau de neurones, nous confirmons cette intuition avec l'analyse de la section 2.5.

Concernant le LUR, l'erreur croit progressivement avec la distance d'échantillonnage, à peu près de façon logarithmique. Nous en déduisons que toute donnée supplémentaire

est bonne à prendre et que sa prédiction repose davantage sur les données explicatives que le réseau de neurones. Pour le réseau de neurones, nous supposons qu'il s'agit à nouveau de la couche de binarisation, composée de trop de neurones, qui est en cause.

Quatrièmement, pour les plus grandes distances d'échantillonnage, le réseau de neurones suit davantage la tendance du LUR que du Krigeage. Ceci est plus marqué pour le NO_2 que pour les PM_{10}. Nous en déduisons que lorsque les données sont moins précises localement, le réseau de neurones exploite davantage les variables explicatives.

A nouveau, les tendances de l'incertitude sont similaires pour les deux polluants. La Figure 2.14 présente l'$incertitude_{Monte-Carlo}(40, ., .)$ en fonction de la distance d'échantillonnage pour le NO_2.

Nous remarquons que les courbes du LUR et du réseau de neurones sont inversées et oscillent. L'oscillation signifie probablement que $card(T_{40,.}) = 20$ n'est pas suffisant pour cette étude de sensibilité. Néanmoins les tendances sont observables.

Pour le LUR, l'incertitude augmente avec la distance d'échantillonnage ce qui paraît cohérent. L'incertitude est acceptable pour les distances inférieures à 200 mètres, puis augmente plus fortement.

Pour le Krigeage, l'incertitude diminue très faiblement à faible distance d'échantillonnage puis augmente significativement. Ceci est lié à l'explication fournie ci-dessus concernant les données d'entraînement exclues du jeu de test. Les distances d'échantillonnage optimales semblent être entre 200 et 400 mètres.

Pour le réseau de neurones, l'incertitude diminue fortement avec la distance d'échantillonnage puis semble stagner, voire commencer à augmenter. Nous supposons qu'elle suit la même tendance que le Krigeage (pour les mêmes raisons qu'énoncées pour l'analyse de l'erreur) et que cela aurait été observable pour des distances d'échantillonnage plus importantes.

Ainsi, nous fixons la période d'échantillonnage à 200 mètres.

Les taux de couverture du jeu de données d'entrée sont tracés en annexe 6.3. Le choix de l'échantillonnage à 200 mètres est représenté sur les courbes sous forme d'une barre verticale rouge.

Nous remarquons que cette distance d'échantillonnage correspond approximativement à l'intersection entre la tangente à l'origine du taux de couverture spatiale et l'axe des abscisses. Pour un jeu de données fixé, l'analyse du taux de couverture théorique pourrait donc servir à déterminer la distance d'échantillonnage optimale.

Nous en déduisons que pour un jeu de données formé, le taux de couverture a bien une influence sur la qualité de la prédiction. Ainsi, cette notion de taux de couverture ponctuelle n'est pas à oublier mais à considérer après la continuité des points entre eux.

Ces deux études de sensibilité confirment donc l'intérêt de mesures mobiles issues de réseaux de type VSN comparativement aux mesures issues de réseaux de type SSN.

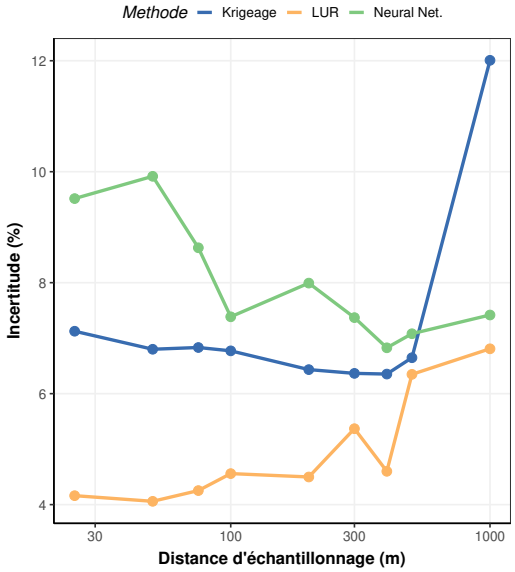


FIGURE 2.14 – Écart-type relatif des simulations pour le coefficient de corrélation en fonction de la distance d'échantillonnage. Le nombre de trajet est fixé à 40. Échelle logarithmique.

Cette conclusion rejoint celle de Alvear et *al.* (2016). Les auteurs ont réalisé une étude sur la base de données réelles montrant que l'orientation du capteur et la période d'échantillonnage ont beaucoup moins d'impact sur les cartes prédites par Krigeage que le trajet emprunté.

De plus, nous tenons à attirer l'attention sur le fait qu'il paraît inutile, voire contre productif, d'échantillonner à des distances plus faibles en vue d'une validation croisée des prédictions. En effet, la validation croisée serait biaisée car toutes les données appartiennent aux mêmes trajets. Or nous avons vu que la notion de trajet (continuité des données) est plus importante que le simple taux de couverture, au regard de la qualité de la prédiction. Les prédictions des jeux de données d'un même jeu de trajets auront ainsi tendance à être identiques même, si la prédiction aurait été différente pour des jeux de trajets différents.

2.5 Analyse de la spatialisation

2.5.1 Cartes prédites par spatialisation

Les Figures 2.15 et 2.16 présentent la carte de référence du jour 14 et les cartes prédites par les trois méthodes de spatialisation, respectivement pour le NO_2 et pour les PM_{10}.

Nous notons que la méthode de spatialisation a un comportement similaire pour le NO_2 et le pour PM_{10}.

Le réseau routier, très marqué dans les cartes de référence, apparait nettement pour les cartes prédites par les deux méthodes qui reposent sur les variables explicatives (LUR et réseau de neurones). Les autoroutes sont globalement bien restituées.

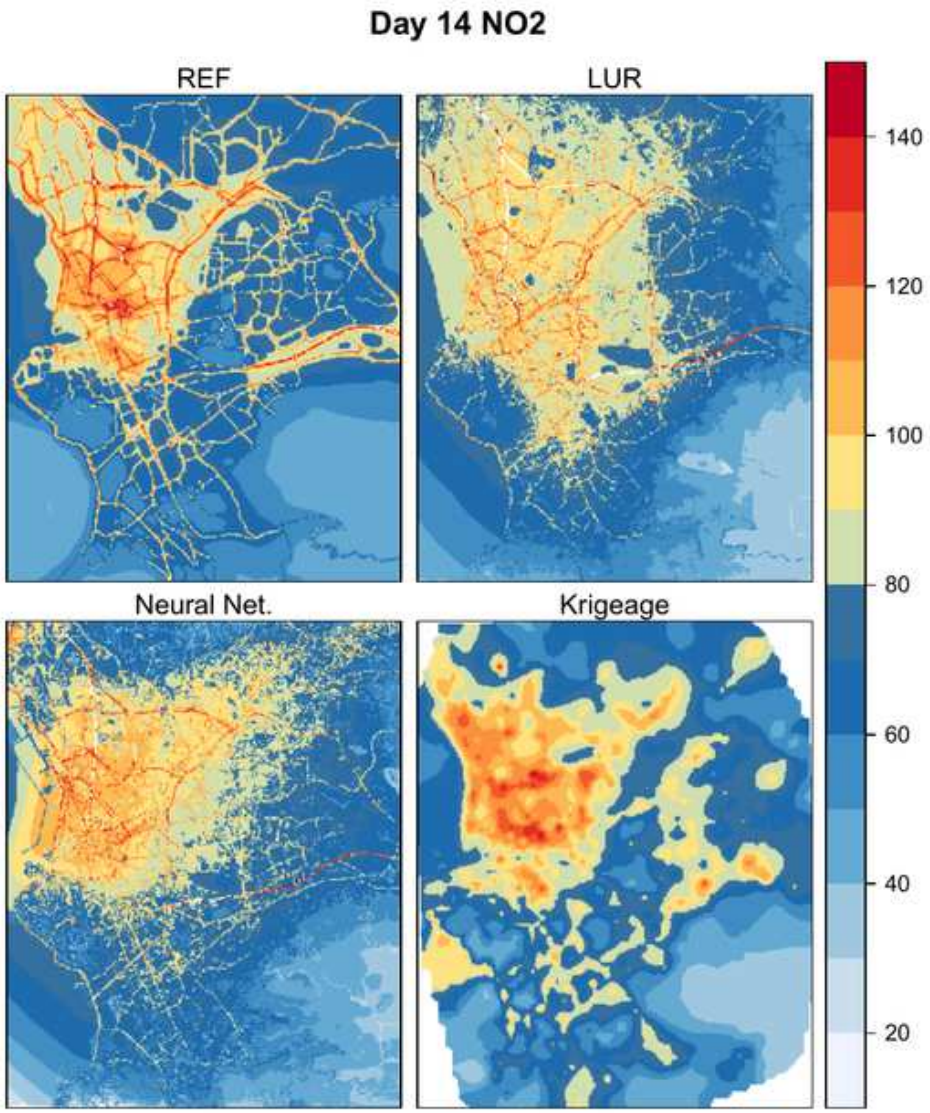


FIGURE 2.15 – Carte de référence pour le NO_2 pour le jour 14 et prédiction correspondante par les différents modèles.

De façon générale, nous observons l'influence des variables explicatives, notamment catégorielles, au travers des variations brutales en concentration de polluants. Pour le LUR, ils définissent presque des niveaux homogènes. Pour le réseau de neurones, nous observons une plus grande variabilité locale.

En outre, nous remarquons que le réseau de neurones reproduit plus fidèlement la tendance de fond de la carte. Ceci est plus marqué pour le NO_2.

Ainsi, le LUR semble généraliser davantage à l'aide des variables explicatives, notamment catégorielles. Le réseau de neurones, lui, semble généraliser davantage à l'aide des variables explicatives continues, dont particulièrement les coordonnées géographiques. Ceci confirme l'analyse de la section précédente concernant l'influence de la localité dans les prédictions par le réseau de neurones.

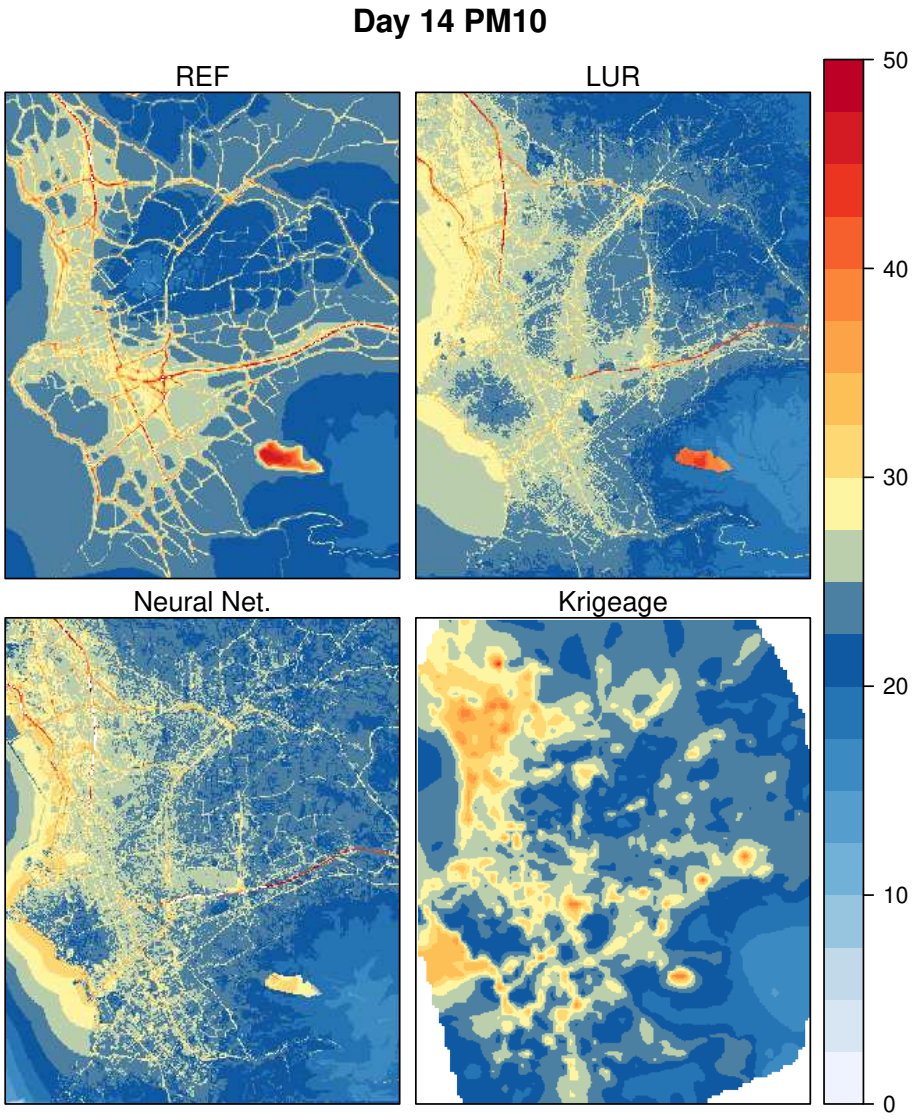


FIGURE 2.16 – Carte de référence pour les PM_{10} pour le jour 14 et prédiction correspondante par les différents modèles.

Le Krigeage nivelle complètement la concentration. Ceci était prévisible puisque son principe est de minimiser la variance spatiale du résidu d'estimation. De plus, cet estimateur est supposé sans biais. Effectivement, il permet bien de retrouver la tendance de fond et de déterminer le centre-ville, les zones résidentielles et le parc naturel (en bas à droite, où le niveau de pollution est le plus faible). Cette méthode souffre néanmoins de la présence des routes dont la variabilité est très forte localement.

Il semble donc intéressant de coupler ces prédictions ; d'utiliser le Krigeage pour déterminer les tendances de fonds et ajuster les prédictions en certaines catégories précises à l'aides de méthodes utilisant les variables explicatives.

2.5.2 Sources d'erreur de spatialisation

Pour analyser les sources d'erreur de spatialisation, nous étudions la qualité des prédictions en fonction des variables explicatives de la ville.

Pour une variable catégorielle, nous nous focalisons sur l'erreur moyenne de la prédiction pour chaque catégorie. Concrètement, en R nous utilisons la fonction mask du package raster.

Pour une variable continue, nous étudions l'erreur moyenne de prédiction en fonction de celle-ci. Par simplicité dans la présentation des résultats et par homogénéité avec les variables catégorielles, nous discrétisons la variable continue en 10 intervalles de même taille. Les variables continues sont donc désormais assimilables à des variables catégorielles. Concernant les variables décrivant les coordonnées GPS, cette approche n'a pas de sens puisque ces deux variables sont intrinsèquement liées.

Cette approche semble pertinente car quelque soit la méthode de spatialisation, quelque soit le polluant, ce sont les mêmes zones qui sont les mieux et les moins bien prédites.

Ces résultats sont présentés sous forme de top 10 des catégories les mieux et moins bien prédites (cf. Figure 2.17). Nous renommons les labels pour une meilleure lisibilité.

Nous observons que beaucoup des zones bien prédites sont des points particuliers à proximité des routes et liés au trafic : station essence, station de taxi, parking, rond-pond, parking à vélo, stop, métro. Ensuite, les jardins partagés et la broussaille arrivent dans notre classement. Ce sont des espaces relativement étendus qui ont une concentration en polluants relativement homogènes. Enfin, les pistes cyclables, qui sont très échantillonnées, sont bien prédites.

Concernant les catégories où la prédiction est la moins bonne, le réseau routier dans sa globalité arrive en tête. Nous notons également que les zones et routes résidentielles, très représentées sur la zone d'étude et dans le réseau routier, sont aussi mal prédites. Ces catégories doivent être trop larges pour réellement apporter une information intéressante. La proximité des routes est surtout mal prédite par le Krigeage. Ceci s'explique simplement par le fait que cette méthode nivelle les concentrations de pollution. Les basses altitudes sont également mal prédites. La majorité des basses altitudes correspond à la mer ; il n'est donc pas surprenant que cette catégorie ait un niveau de pollution erroné. Effectivement, nous supposons que le LUR et le réseau de neurones ont « appris » le niveau de la pollution à partir de quelques basses altitudes en ville, notamment près du port. Enfin, les zones de la ville dont les variables explicatives ne possèdent aucune information et les zones presque jamais échantillonnées (forêt, routes privées, réserve naturelle, zone industrielle) ont de mauvais résultats.

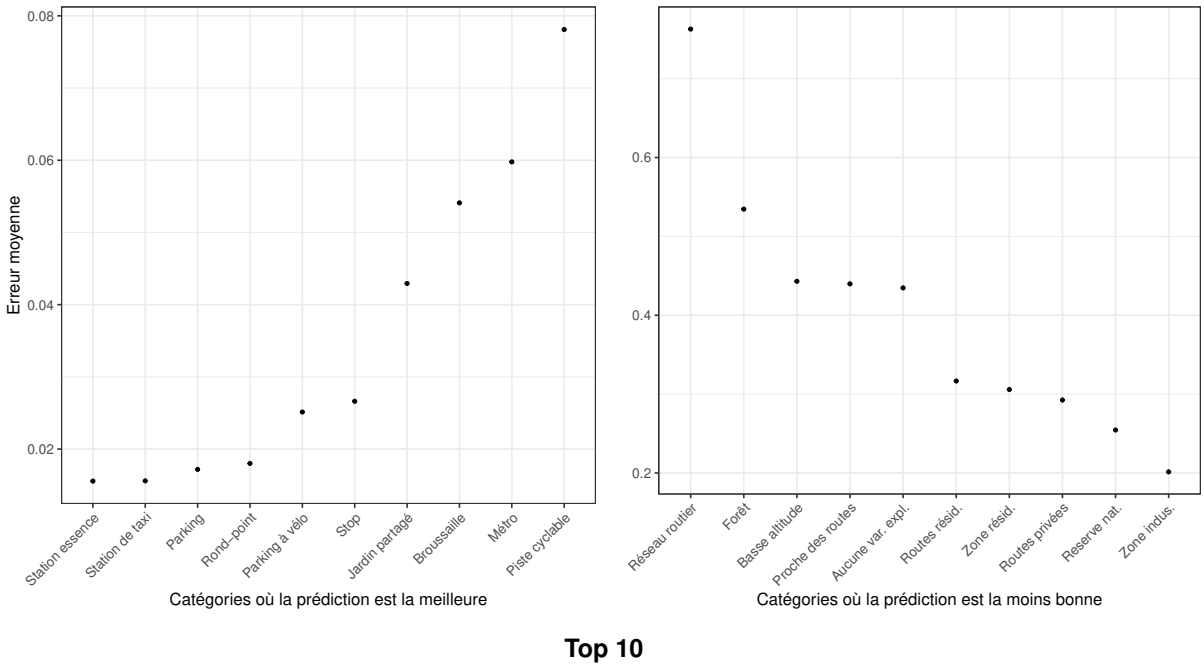


FIGURE 2.17 – Top 10 des catégories des variables explicatives de la ville les mieux et les moins bien prédites par les méthodes de spatialisation.

2.5.3 Détection d'une perturbation spatiale

Pour tester la capacité de détection de chaque modèle, nous perturbons le signal d'entrée (décrit par la carte de référence).

Nous choisissons trois types de perturbation :

— un bruit blanc (distribution gaussienne centrée en $0\,\mu g/m^3$ de variance $1\,\mu g/m^3$)
— une perturbation linéique, le long d'une route de type résidentiel située entre les deux zones attractives (centre-ville et plages) afin d'assurer la présence de trajets,
— une perturbation sphérique au centre-ville, de 2,5 km de diamètre.

L'intensité de la perturbation linéique est d'environ deux fois la valeur maximale de la carte (précisément $200\,\mu g/m^3$ pour le NO_2 et $100\,\mu g/m^3$ pour les PM_{10}).

L'intensité de la perturbation sphérique est d'environ quatre fois la valeur maximale de la carte (précisément $400\,\mu g/m^3$ pour le NO_2 et $200\,\mu g/m^3$ pour les PM_{10}) et de décroissance exponentielle jusqu'à environ deux fois la valeur maximale de la carte en bord de perturbation (précisément $200\,\mu g/m^3$ pour le NO_2 et $100\,\mu g/m^3$ pour les PM_{10}).

Nous avons choisi ces types de perturbation car :

— le premier peut être assimilable à un bruit blanc de mesure du capteur,
— le second permet d'envisager l'impact du parcours des vélos sur une route fortement non représentative,
— le troisième correspond à une source de pollution dont l'émission est particulièrement élevée par rapport au reste de la zone d'étude.

Pour observer l'impact sur une méthode de spatialisation d'une perturbation de la carte de référence, nous étudions la différence entre les cartes prédites à l'aide de deux

jeux de mesures issus d'un même jeu de trajets, l'un synthétisé à partir de la carte de référence avec perturbation et l'autre synthétisé à partir de la carte de référence sans perturbation.

Pour les paramètres déterminés (40 trajets échantillonnés à 200 mètres), nous effectuons cette reconstitution pour les 20 jeux de données différents sur les 14 jours étudiés. Les résultats moyens de la reconstitution des perturbations sont présentés ci-après.

Nous observons des résultats similaires pour le NO_2 et les PM_{10}. Nous présentons ceux pour le NO_2. La Figure 2.18 permet de visualiser la perturbation linéique et les perturbations reconstituées à l'aide des méthodes de spatialisation. Les autres Figures, correspondant aux perturbations de type bruit blanc et sphérique, sont en annexe 6.4.

Premièrement, concernant le bruit blanc, nous observons que toutes les méthodes y sont robustes. Les écarts-types des perturbations reconstituées sont de l'ordre du dixième de celui de la perturbation de référence pour les deux méthodes utilisant des variables expliquées. Celui du Krigeage est du même ordre de grandeur.

Le Krigeage retrouve un bruit blanc, mais corrélé spatialement. Nous notons la présence d'effets de bord linéiques. Nous n'y voyons pas d'explication.

Le LUR et le réseau de neurones propagent le bruit blanc sur des catégories. Ainsi, nous retrouvons visuellement la mer et la structure de la ville. Le LUR semble répartir spatialement le bruit blanc entre les catégories. Le réseau de neurones souffre d'effets de bord aux extrémités de la carte. Malgré cela, le bruit est clairement négligeable puisque le maximum de bruit introduit est de $0{,}15\,\mu g/m^3$ et $0{,}8\,\mu g/m^3$ respectivement pour le LUR et pour le réseau de neurones.

Pour les perturbations sphériques et linéiques, les conclusions sont proches.

Le Krigeage permet de bien déterminer la localisation de la perturbation mais minimise son intensité. Pour la perturbation linéique de $200\,\mu g/m^3$, la perturbation reconstituée atteint son maximum en $11{,}3\,\mu g/m^3$; pour la perturbation sphérique de $200\,\mu g/m^3$ à $400\,\mu g/m^3$, la perturbation reconstituée atteint son maximum en $180\,\mu g/m^3$. La perturbation sphérique est ainsi bien mieux reconstituée. Ceci s'explique par le fait qu'une hypothèse du Krigeage est de supposer l'erreur spatiale isotrope (indépendante de la direction). Nous observons même que la perturbation linéique est reconstituée par de petites sphères.

Pour le LUR, la perturbation sphérique est attribuée à la route qui la traverse. En conséquence, son niveau est plus haut que le reste de la carte ($116\,\mu g/m^3$). Cependant, nous ne retrouvons pas significativement la perturbation mais nous observons, de l'ordre de la dizaine de $\mu g/m^3$, sa diffusion sur environ un tiers de la zone d'étude. Dans le cas d'une perturbation linéique, c'est le type de voies de la perturbation qui est rehaussé d'environ $11{,}3\,\mu g/m^3$. Cela introduit donc une erreur sur toute la zone d'étude. Nous supposons que l'intensité de la perturbation est moyennée sur toute la zone d'étude, d'où l'impact relativement faible.

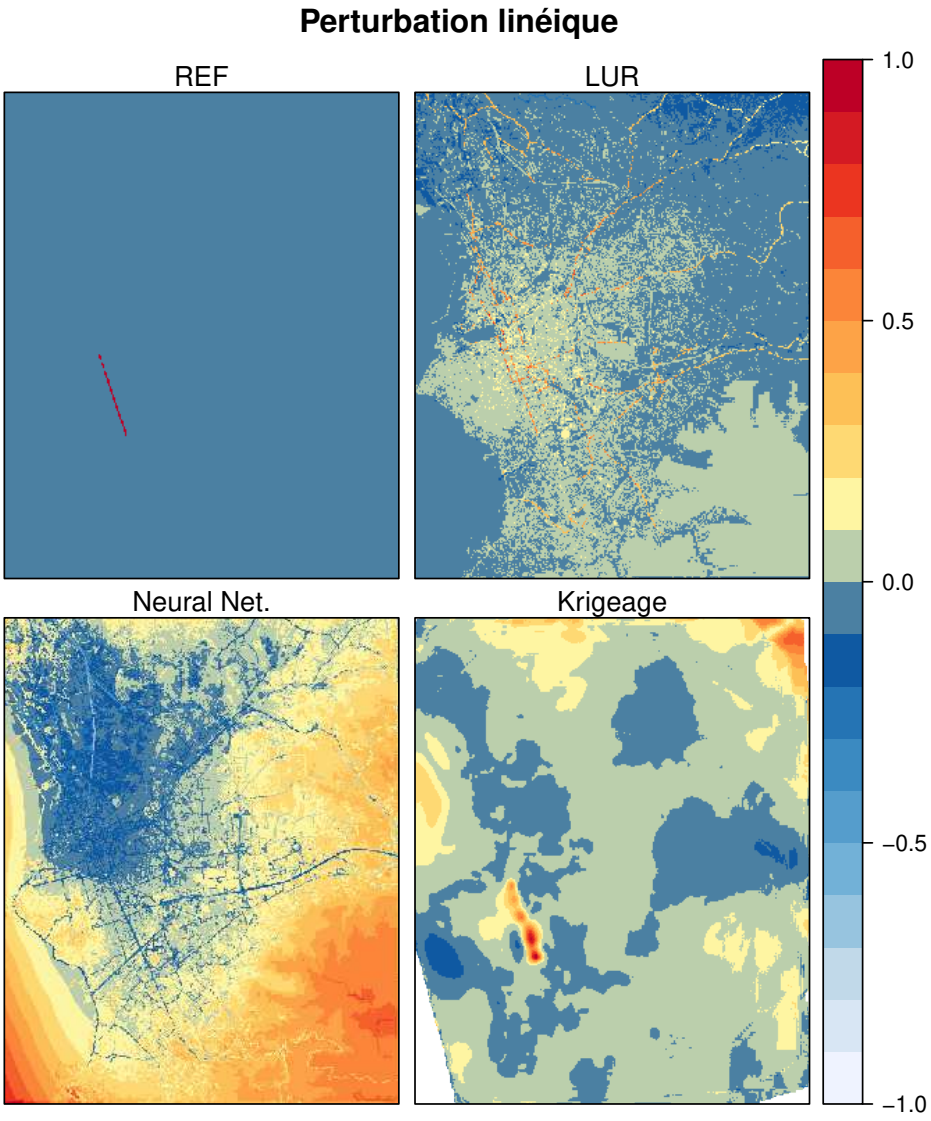


FIGURE 2.18 – Perturbation linéique introduite sur la carte de référence (REF) et détection de la perturbation par les méthodes de spatialisation. Application pour le NO_2, résultats moyennés pour les 14 jours étudiés. Les perturbations sont normalisées par leur maximum ($222\,\mu g/m^3$ pour la référence, $11{,}3\,\mu g/m^3$ pour le LUR, $14{,}7\,\mu g/m^3$ pour le réseau de neurones, $23{,}8\,\mu g/m^3$ pour le Krigeage).

Pour le réseau de neurones, la perturbation reconstituée est inversée par rapport à la perturbation introduite. Il est difficile d'expliquer cette inversion. Nous supposons que c'est parce que le réseau de neurones introduit une variabilité globale plus faible, et que la moyenne de la carte est impactée par la perturbation. En conséquence, la zone la plus concentrée en polluant est sous-évaluée et les autres zones sont sur-évaluées. Cependant, ces différences locales de niveaux de pollution permettent de mettre en évidence que le réseau de neurones s'appuie fortement sur les coordonnées géographiques pour établir la prédiction.

En définitive, le Krigeage est plus adapté pour retrouver une perturbation de type sphérique. La détection d'une route qui a un niveau de pollution particulier est plus difficile. Les résultats du LUR souligne l'importance de nombreux points de mesures pour ne pas biaiser l'ensemble de la carte sur la base des observations de quelques SSN.

2.6 Conclusion

Ce chapitre a présenté une analyse de l'apport d'un réseau de capteurs mobiles sur vélo pour estimer globalement le niveau de polluants dans la ville. L'analyse s'est basée sur une simulation des niveaux de polluants et des trajets d'un ensemble de vélos pour la ville de Marseille.

Les trajets des vélos ont été simulés en considérant une distribution gaussienne de la longueur des trajets et une répartition aléatoire des directions. Les zones de convergences ont été supposées correspondre aux zones les plus attractives (centre-ville et plages) et les trajets à vélo ont été élaborés en utilisant un déplacement idéal. Comparativement avec les données disponibles, nos vélos fictifs semblent assez représentatifs des déplacements dans la ville. La couverture de la ville est importante et variée d'un point de vue de la représentativité spatiale et de la diversité des zones traversées. Ce résultat est bien entendu à nuancer compte-tenu de la grande variété des déplacements individuels à vélo. Il manque par exemple des trajets de loisir dans les zones extra-urbaines ou des trajets de navette péri-urbaine qui apporteraient des informations sur le niveau moyen de pollution dans la ville par rapport à son environnement.

Les observations synthétiques ont été générées à partir d'une estimation assez réaliste de la répartition spatiale du NO_2 et des PM_{10} dans la ville. Cette estimation reste une simulation et représente un état moyen de la répartition des polluants. Dans le cadre d'observations réelles, nous aurions beaucoup plus de variabilité en raison de phénomènes très locaux (émissions ou météorologie) que ne peut pas représenter le modèle numérique. Les résultats de notre simulation ont des tendances similaires pour ces deux polluants.

Le Krigeage tend vers une méthode optimale d'interpolation lorsque la densité de mesure est suffisamment importante. Cependant, cette collecte est coûteuse et sa performance est faible lorsque les mesures sont très peu nombreuses. Cette technique semble toutefois utile pour déterminer la tendance de fond locale.

Nous avons testé des modèles d'estimation non linéaire dont l'hypothèse de base est la possibilité de prédire les niveaux de concentrations à partir des éléments qui structurent la ville (route, points d'intérêt...). Nos résultats montrent qu'à partir d'un nombre réduit de vélos (40) et d'un échantillonnage tous les 200 m, il est possible de restituer des niveaux de concentrations cohérents avec les simulations de référence. Les conclusions faites sur ces méthodes sont récapitulées Table 2.4.

Nous avons pu également analyser l'impact d'une hétérogénéité locale (surexposition d'une zone en particulier) et d'un faible bruit de mesure. Les méthodes sont robustes au bruit de mesure, mais en dehors du Krigeage, il semble qu'elles soient assez peu résistantes et propagent cette hétérogénéité au reste de la carte.

Nous avons fait l'hypothèse que la mesure était instantanée sur l'ensemble du domaine de manière à se focaliser uniquement sur les méthodes de spatialisation. Il est probable

que cette hypothèse introduit un biais dans l'estimation du nombre de trajets nécessaires pour une estimation globale de la pollution. En effet, les polluants ont un cycle journalier et l'heure de mesure va donc influencer l'estimation finale. Ceci sera particulièrement marqué pour des polluants secondaires comme l'ozone, par exemple, qui ont un cycle photochimique. Toutefois, nous observons au chapitre 4 que le cycle journalier du polluant émis par les automobiles (tel que le NO_2) correspond au cycle journalier d'utilisation de certaines bases de cyclistes (vélo-taffeurs en particulier).

TABLE 2.4 – Comparaison du Krigeage, du LUR et du réseau de neurones pour l'estimation de la pollution de l'air en ville. Les signes + et – caractérisent la performance de la méthode d'estimation au regard d'un critère.

Critère	Krig.	LUR	NN
Estim. concentration de fond	+++	+	++
Estim. sources	–	++	+
Estim. de l'étendue	+	++	– –
Taille du réseau	+	+++	++
Echantillonnage spatiale	–	+	+
Détection perturbation	++	+/–	– –

Bibliographie

ADAMS, M. D. et KANAROGLOU, P. S. . Mapping real-time air pollution health risk for environmental management : Combining mobile and stationary air pollution monitoring with neural network models. Journal of Environmental Management, 2016. DOI : 10.1016/j.jenvman.2015.12.012.

AL-ALI, A. R. , ZUALKERNAN, I. et ALOUL, F. . A Mobile GPRS-Sensors Array for Air Pollution Monitoring. IEEE Sensors Journal, 2010. DOI : 10.1109/JSEN.2010.2045890.

ALVEAR, O. , ZAMORA, W. , CALAFATE, C. T. , CANO, J.-C. et MANZONI, P. . EcoSensor : Monitoring environmental pollution using mobile sensors. 2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2016. DOI : 10.1109/WoWMoM.2016.7523519.

BAILLARGEON, S. . Le krigeage : revue de la théorie et application à l'interpolation spatiale dedonnées de précipitations. Mémoire de maîtrise, Université Laval, 2005. http://hdl.handle.net/20.500.11794/18036.

BALES, E. , NIKZAD, N. , QUICK, N. , ZIFTCI, C. , PATRICK, K. et GRISWOLD, W. . Citisense : Mobile Air Quality Sensing for Individuals and Communities. Design and deployment of the Citisense mobile air-quality system. Proceedings of the 6th International Conference on Pervasive Computing Technologies for Healthcare, 2012. DOI : 10.4108/icst.pervasivehealth.2012.248724.

Bergmeir, C. et Benítez, J. M. . Neural Networks in R Using the Stuttgart Neural Network Simulator : RSNNS. Journal of Statistical Software, 2012. DOI : 10.18637/jss.v046.i07.

Delaine, F. , Lebental, B. et Rivano, H. . *In Situ* Calibration Algorithms for Environmental Sensor Networks : A Review. IEEE Sensors Journal, 2019. DOI : 10.1109/JSEN.2019.2910317.

Devarakonda, S. , Sevusu, P. , Liu, H. , Liu, R. , Iftode, L. et Nath, B. . Real-time air quality monitoring through mobile sensing in metropolitan areas. Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing, 2013. DOI : 10.1145/2505821.2505834.

Dons, E. , Poppel, M. V. , Kochan, B. , Wets, G. et Panis, L. I. . Modeling temporal and spatial variability of traffic-related air pollution : Hourly land use regression models for black carbon. Atmospheric Environment, 2013. DOI : 10.1016/j.atmosenv.2013.03.050.

Ghassoun, Y. , Ruths, M. , Löwner, M.-O. et Weber, S. . Intra-urban variation of ultrafine particles as evaluated by process related land use and pollutant driven regression modelling. Science of The Total Environment, 2015. DOI : 10.1016/j.scitotenv.2015.07.051.

Hasenfratz, D. , Saukh, O. , Sturzenegger, S. et Thiele, L. . Participatory air pollution monitoring using smartphones. Proceedings of the 2nd International Workshop on Mobile Sensing, 2012. https://tik.ethz.ch/file/b6c2122d089d2ef88348a74ddf2906dc/HSST2012.pdf.

Hasenfratz, D. , Saukh, O. , Walser, C. , Hueglin, C. , Fierz, M. , Arn, T. , Beutel, J. et Thiele, L. . Deriving high-resolution urban air pollution maps using mobile sensor nodes. Pervasive and Mobile Computing, 2015. DOI : 10.1016/j.pmcj.2014.11.008.

Hiemstra, P. H. , Pebesma, E. J. , Twenhöfel, C. J. W. et Heuvelink, G. B. M. . Real-time automatic interpolation of ambient gamma dose rates from the Dutch radioactivity monitoring network. Computers and Geosciences, 2009. DOI : 10.1016/j.cageo.2008.10.011.

Hoek, G. , Beelen, R. , Hoogh, K. d. , Vienneau, D. , Gulliver, J. , Fischer, P. et Briggs, D. . A review of land-use regression models to assess spatial variation of outdoor air pollution. Atmospheric Environment, 2008. DOI : 10.1016/j.atmosenv.2008.05.057.

Honicky, R. , Brewer, E. A. , Paulos, E. et White, R. . N-smarts : networked suite of mobile atmospheric real-time sensors. Proceedings of the second ACM SIGCOMM workshop on Networked systems for developing regions, 2008. DOI : 10.1145/1397705.1397713.

Hsieh, H.-P. , Lin, S.-D. et Zheng, Y. . Inferring Air Quality for Station Location Recommendation Based on Urban Big Data. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015. DOI : 10.1145/2783258.2783344.

Hu, S.-C. , Wang, Y.-C. , Huang, C.-Y. et Tseng, Y.-C. . A Vehicular Wireless Sensor Network for CO2 Monitoring. Proceedings of IEEE Sensors, 2009. DOI : 10.1109/ICSENS.2009.5398461.

Jain, A. K. , Mao, J. et Moidin Mohiuddin, K. . Artificial Neural Networks : A Tutorial. Computer, 1996. DOI : 10.1109/2.485891.

Janssen, S. , Dumont, G. , Fierens, F. et Mensink, C. . Spatial interpolation of air pollution measurements using CORINE land cover data. Atmospheric Environment, 2008. DOI : 10.1016/j.atmosenv.2008.02.043.

Jerrett, M. , Arain, A. , Kanaroglou, P. , Beckerman, B. , Potoglou, D. , Sahsuvaroglu, T. , Morrison, J. et Giovis, C. . A review and evaluation of intraurban air pollution exposure models. Journal of Exposure Analysis and Environmental Epidemiology, 2004. DOI : 10.1038/sj.jea.7500388.

Kosmidis, E. , Syropoulou, P. , Tekes, S. , Schneider, P. , Spyromitros-Xioufis, E. , Riga, M. , Charitidis, P. , Moumtzidou, A. , Papadopoulos, S. , Vrochidis, S. , Kompatsiaris, I. , Stavrakas, I. , Hloupis, G. , Loukidis, A. , Kourtidis, K. , Georgoulias, A. et Alexandri, G. . hackAIR : Towards Raising Awareness about Air Quality in Europe by Developing a Collective Online Platform. ISPRS International Journal of Geo-Information, 2018. DOI : 10.3390/ijgi7050187.

Kuhn, M. . Building Predictive Models in *R* Using the **caret** Package. Journal of Statistical Software, 2008. DOI : 10.18637/jss.v028.i05.

Kurt, A. , Gulbagci, B. , Karaca, F. et Alagha, O. . An online air pollution forecasting system using neural networks. Environment International, 2008. DOI : 10.1016/j.envint.2007.12.020.

Li, J. J. , Faltings, B. , Saukh, O. , Hasenfratz, D. et Beutel, J. . Sensing the Air We Breathe - the Opensense Zurich Dataset. Proceedings of the National Conference on Artificial Intelligence, 2012. https://dl.acm.org/citation.cfm?id=2900775.

Li, S. , Zhai, L. , Zou, B. , Sang, H. et Fang, X. . A Generalized Additive Model Combining Principal Component Analysis for PM2.5 Concentration Estimation. ISPRS International Journal of Geo-Information, 2017. DOI : 10.3390/ijgi6080248.

Lo Re, G. , Peri, D. et Vassallo, S. D. . Urban Air Quality Monitoring Using Vehicular Sensor Networks. Advances onto the Internet of Things, 2014. DOI : 10.1007/978-3-319-03992-3_22.

Marjovi, A. , Arfire, A. et Martinoli, A. . High Resolution Air Pollution Maps in Urban Environments Using Mobile Sensor Networks. International Conference on Distributed Computing in Sensor Systems, 2015. DOI : 10.1109/DCOSS.2015.32.

Mercer, L. D. , Szpiro, A. A. , Sheppard, L. , Lindstrom, J. , Adar, S. D. , Allen, R. W. , Avol, E. L. , Oron, A. P. , Larson, T. , Liu, L.-J. S. et Kaufman, J. D. . Comparing universal kriging and land-use regression for predicting concentrations of gaseous oxides of nitrogen (NOx) for the Multi-Ethnic Study of Atherosclerosis and Air Pollution (MESA Air). Atmospheric Environment, 2011. DOI : 10.1016/j.atmosenv.2011.05.043.

Mueller, M. D. , Hasenfratz, D. , Saukh, O. , Fierz, M. et Hueglin, C. . Statistical modelling of particle number concentration in Zurich at high spatio-temporal resolution utilizing data from a mobile sensor network. Atmospheric Environment, 2016. DOI : 10.1016/j.atmosenv.2015.11.033.

Méndez, D. , Pérez, A. J. , Labrador, M. A. et Juan José Marrón. P-Sense : A participatory sensing system for air pollution monitoring and control. IEEE International Conference on Pervasive Computing and Communications Workshops, 2011. DOI : 10.1109/PERCOMW.2011.5766902.

Niska, H. , Hiltunen, T. , Karppinen, A. , Ruuskanen, J. et Kolehmainen, M. . Evolving the neural network model for forecasting air pollution time series. Engineering Applications of Artificial Intelligence, 2004. DOI : 10.1016/j.engappai.2004.02.002.

Onkal-Engin, G. , Demir, I. et Hiz, H. . Assessment of urban air quality in Istanbul using fuzzy synthetic evaluation. Atmospheric Environment, 2004. DOI : 10.1016/j.atmosenv.2004.03.058.

Qi Gan, W. , Koehoorn, M. , Davies, H. W. , Demers, P. A. , Tamburic, L. et Brauer, M. . Long-Term Exposure to Traffic-Related Air Pollution and the Risk of Coronary Heart Disease Hospitalization and Mortality. Environmental Health Perspectives, 2011. DOI : 10.1289/ehp.1002511.

Romanowicz, R. , Young, P. , Brown, P. et Diggle, P. . A recursive estimation approach to the spatio-temporal analysis and modelling of air quality data. Environmental Modelling & Software, 2006. DOI : 10.1016/j.envsoft.2005.02.004.

Russo, A. , Raischel, F. et Lind, P. G. . Air quality prediction using optimal neural networks with stochastic variables. Atmospheric Environment, 2013. DOI : 10.1016/j.atmosenv.2013.07.072.

Saukh, O. , Hasenfratz, D. et Thiele, L. . Reducing multi-hop calibration errors in large-scale mobile sensor networks. Proceedings of the 14th International Conference on Information Processing in Sensor Networks, 2015. DOI : 10.1145/2737095.2737113.

Saukh, O. , Hasenfratz, D. , Walser, C. et Thiele, L. . On rendezvous in mobile sensing networks. Real-World Wireless Sensor Networks, 2014. DOI : 10.1007/978-3-319-03071-5_3.

Sivaraman, V. , Carrapetta, J. , Hu, K. et Luxan, B. G. . HazeWatch : A participatory sensor system for monitoring air pollution in Sydney. 38th Annual IEEE Conference on Local Computer Networks - Workshops, 2013. DOI : 10.1109/LCNW.2013.6758498.

Su, J. G. , Jerrett, M. , Beckerman, B. , Wilhelm, M. , Ghosh, J. K. et Ritz, B. . Predicting traffic-related air pollution in Los Angeles using a distance decay regression selection strategy. Environmental Research, 2009. DOI : 10.1016/j.envres.2009.06.001.

Völgyesi, P. , Nádas, A. , Koutsoukos, X. et Lédeczi, A. . Air Quality Monitoring with SensorMap. International Conference on Information Processing in Sensor Networks, 2008. DOI : 10.1109/IPSN.2008.50.

Wong, K.-J. , Chua, C. C. et Li, Q. . Environmental Monitoring Using Wireless Vehicular Sensor Networks. 5th International Conference on Wireless Communications, Networking and Mobile Computing, 2009. DOI : 10.1109/WICOM.2009.5303846.

Wood, S. N. . Generalized Additive Models : An Introduction with R. Chapman & Hall, 2017. ISBN 978-1-4987-2833-1.

Yi, W. Y. , Lo, K. M. , Mak, T. , Leung, K. S. , Leung, Y. et Meng, M. L. . A Survey of Wireless Sensor Network Based Air Pollution Monitoring Systems. Sensors, 2015. DOI : 10.3390/s151229859.

Zheng, Y. , Yi, X. , Li, M. , Li, R. , Shan, Z. , Chang, E. et Li, T. . Forecasting Fine-Grained Air Quality Based on Big Data. Proceedings of the 21th SIGKDD conference on Knowledge Discovery and Data Mining, 2015. DOI : 10.1145/2783258.2788573.

Chapitre 3

Conception d'un système embarqué pour la pollution de l'air en zone urbaine

Les choses ne changent pas. Change ta façon de les voir, cela suffit.
– Lao Tseu

Sommaire

3.1 Introduction

Dans le chapitre 1, nous avons présenté les enjeux de la mesure de la pollution de l'air à l'échelle urbaine. En bref, la mobilité du système de mesure promet une meilleure couverture spatiale de la zone étudiée, mais conditionne la famille de capteurs choisie et donc la précision de la mesure. En effet, l'encombrement, le prix, la consommation énergétique et les besoins de maintenance diffèrent en fonction de la famille.

Dans le chapitre deux, nous avons établi un état de l'art des différents projets articulés autour de la mesure de la pollution de l'air en ville à l'aide d'un réseau de capteurs mobiles. Nous en avons conclu qu'un compromis satisfaisant semble être l'utilisation de micro-capteurs low-cost embarqués sur des vélos. Ensuite, nous avons développé une méthode théorique pour dimensionner le nombre de capteurs nécessaires pour couvrir et modéliser une ville en fonction d'un polluant.

Dans ce chapitre, nous comparons les différents types de micro-capteurs au travers de leur fonctionnement et expliquons la raison pour laquelle nous choisissons d'étudier les capteurs à Métal-Oxyde Semi-conducteur, et en particulier le MiCS-4514. Ensuite, nous présentons le prototypage de notre système autour ce capteur, de l'analyse du besoin au produit final, et le retour de notre expérience sur les aspects les plus difficiles à mettre en œuvre. Enfin, nous évaluons les performances individuelles de nos capteurs (capteur MiCS-4514 et capteur de pression, température, humidité) en laboratoire, puis intercomparons plusieurs systèmes de mesure entre eux *in situ*.

3.2 Micro-capteurs low-cost de pollution de l'air extérieur

Un micro-capteur low-cost est défini au niveau national par le LCSQA [1] comme un capteur de prix inférieur au dixième du prix d'un capteur de référence et de poids, de dimensions, d'encombrement permettant la portabilité par un individu [2].

3.2.1 Comparaison des familles de micro-capteurs

Les micro-capteurs low-cost de pollution de l'air en ville reposent sur quatre familles de capteurs : les semi-conducteurs, les électrochimiques, ceux à absorption infrarouge et ceux à photo-ionisation.

Les capteurs semi-conducteurs reposent sur l'interaction entre une couche sensible semi-conductrice dopée (excédent d'électrons – type N, ou de trous – type P) et l'air ambiant à une température donnée. Cette interaction conduit à l'échange d'électrons et modifie la conductivité de la couche sensible. La mesure de la concentration d'un gaz s'effectue alors via celle de la résistivité d'une couche sensible donnée, à température donnée atteinte à l'aide d'un chauffage. Ces capteurs disposent d'une très bonne sensibilité, mais d'une faible sélectivité (qui peut être améliorée grâce à l'utilisation de membrane

1. Pour rappel, LCSQA signifie Laboratoire Central de Surveillance de la Qualité de l'Air.
2. `https://www.atmoSud.org/sites/paca/files/atoms/files/180518_biblio_microcapteur_2018.pdf`

spécifique), dérivent (par modification de la couche sensible initiale) et sont sensibles à l'humidité. Néanmoins pour pallier le problème de dérive, la couche sensible peut être « nettoyée » en la chauffant intensément ce qui lui permet de retourner vers son état dopé initial.

Les capteurs électrochimiques utilisent les réactions d'oxydo-réduction qui s'opèrent par électrolyse avec l'air ambiant. La variation de concentration en gaz est alors mesurée via celle du courant entre une électrode de travail et une contre-électrode. Le flux d'air au contact des ions mobiles de l'électrolyte est contrôlé par une ouverture capillaire et une barrière hydrophobe et est filtré par une membrane spécifique. Ces capteurs peuvent atteindre une précision de l'ordre du ppm et sont relativement stables dans le temps, mais souffrent d'une dérive (par modification de la charge initiale de l'électrode de travail) et d'une faible sélectivité (qui peut être améliorée grâce à l'utilisation de plusieurs membranes spécifiques).

Les capteurs à absorption infrarouge transmettent un rayonnement électromagnétique infrarouge sur le gaz (ou les particules) et exploitent l'atténuation du flux lumineux (loi de Beer-Lambert). Cette atténuation peut être analysée par image directe ou par son intensité et la forme de la diffusion de la lumière. Ces capteurs sont peu sensibles aux conditions expérimentales (température, humidité), mais ne fonctionnent qu'avec certains gaz. Par exemple, le CO_2 est très bien isolé, mais ce n'est pas le cas pour les hydrocarbures.

Les capteurs-détecteurs à photo-ionisation ionisent les molécules de gaz à l'aide d'une lampe à rayonnement UV, dans une enceinte fermée et équipée de deux électrodes soumises à une forte différence de potentiel. Cela permet de collecter sur la cathode les ions formés et ainsi induire un courant proportionnel à la concentration du gaz. Ces capteurs ne peuvent pas détecter tous les gaz, mais ont une réponse rapide, relativement linéaire et extrêmement sensible (de l'ordre du ppb – rapport de mélange exprimé en partie par milliards) pour les COVs. Néanmoins, ils nécessitent une maintenance régulière.

La Table 3.1 – sur la base du travail de Yi et *al.* (2015) et de Menini (2011) – compare ces familles de capteurs.

Ainsi, bien qu'ils souffrent de sélectivité et dérivent, les capteurs semi-conducteurs et électrochimiques sont ceux qui permettent de cibler le plus de gaz. Ces technologies semblent donc prometteuses pour développer des approches généralisables. Enfin, nous choisissons de nous focaliser sur les capteurs semi-conducteurs, et plus particulièrement les capteurs à métal-oxyde semi-conducteur (MOx), car c'est une technologie développée dans notre laboratoire par l'équipe Microsystèmes d'Analyse (MICA), sous la responsabilité de Philippe Menini.

TABLE 3.1 – Comparaison des familles de micro-capteurs low-cost.

	Semi-conducteurs	Électrochimique	Absorption infrarouge	Photo-ionisation
Gaz cibles et applications	environ 150 gaz dont COV, NO_x, NH3, composés soufrés, Pollution automobile	environ 20 gaz dont COV,NO_x, NH_3, composés soufrés	Hydrocarbures, O_2, CO_2	COV, O_2, CO_2
Linéarité	linéaire à température opérationnelle	linéaire à température ambiante	non linéaire	relativement linéaire
Sensibilité	bonne, améliorée par l'utilisation d'un filtre	moyenne, améliorée par l'utilisation d'un filtre	très bonne en générale faible pour les hydrocarbures	bonne
Sélectivité	mauvaise	bonne	très bonne	mauvaise
Stabilité	mauvaise	très mauvaise	bonne	bonne
Consommation	bonne	moyenne	mauvaise	mauvaise
Coût	très faible	faible	cher	cher
Maintenance	faible	faible	très faible	fréquente
Temps de réponse	20 s à 90 s	<50 s	<20 s	<3 s
Temps de vie	10+ ans	1-2 ans	3–5 ans	environ 6000 h
Portabilité/intégrabilité	très bonne	mauvaise	mauvaise	très bonne

3.2.2 Les capteurs à Métal-Oxyde Semi-conducteur

Les capteurs à Métal-Oxyde Semi-conducteur sont des capteurs semi-conducteurs dont la couche sensible est un oxyde métallique qui réagit avec les oxydes présents dans l'atmosphère.

Dufour (2013) étudie la relation entre la puissance consommée et la température de la couche sensible d'un capteur MOx à l'aide d'une simulation et de données réelles obtenues par une caméra à infrarouge. La Figure 3.1 présente ses résultats. Nous observons que la puissance consommée est de l'ordre de quelques milliwatts, ce qui peut être généré par un cycliste via une dynamo. Ceci confirme notre choix pour cette technologie.

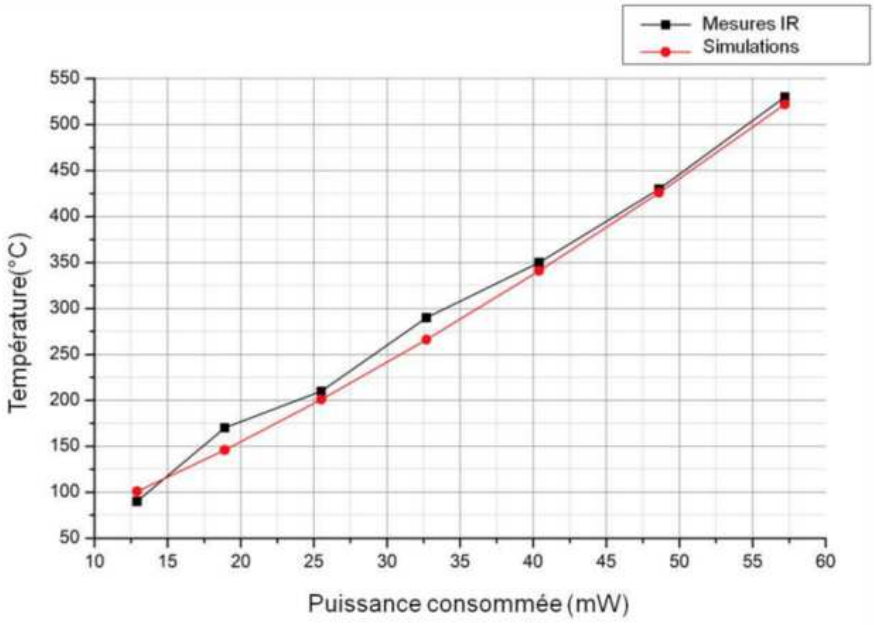


FIGURE 3.1 – Comparaison entre les simulations et les mesures IR de la température maximale à la surface du capteur en fonction de la puissance consommée — d'après Dufour (2013).

Le principe de fonctionnement de ce type de capteur est rappelé au travers de la Figure 3.2 extraite de Korotcenkov et Cho (2017). A une température donnée, le gaz cible échange des électrons avec la couche sensible d'oxyde métallique dopée et modifie ainsi la résistivité de la couche sensible. Deux électrodes aux extrémités de la couche sensible servent à mesurer sa résistivité et ainsi à représenter les variations en concentration de gaz.

Il existe deux types de dopages : N en cas d'excédent d'électrons et P et en cas d'excédent de trous. Au LAAS-CNRS dans l'équipe MICA, les oxydes métalliques les plus utilisés sont le SnO_2, et le ZnO pour le type N et le CuO pour le type P. Les réactions sont similaires pour les types N et P mais sont inversées : un gaz oxydant pour un oxyde métallique de type N est réducteur pour un type P. Nous nous focalisons sur le type N pour le NO_2 et le CO.

Le NO_2 se transforme en NO en présence de l'O_2 présent dans l'air. La couche sensible perd des électrons lors de la réaction, NO_2 est donc oxydant pour le type N. En consé-

quence, la réponse du capteur, c'est-à-dire la résistance de la couche sensible, augmente en présence de NO_2. A l'inverse, le CO se transforme en CO_2 en présence de l'O_2 présent dans l'air. La couche sensible gagne des électrons lors de la réaction, CO est donc réducteur pour le type N. La réponse du capteur diminue en présence de CO.

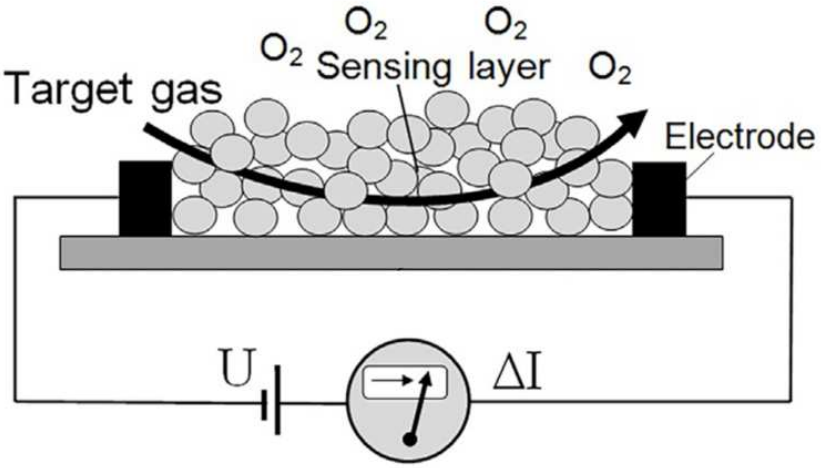


FIGURE 3.2 – Principe de fonctionnement des MOx — d'après Korotcenkov et Cho (2017).

Néanmoins, la couche sensible n'est pas exclusive à un certain gaz. Korotcenkov et Cho (2017) étudient plusieurs oxydes métalliques à différentes températures de chauffage pour déterminer leurs sensibilités en fonction des gaz. La Figure 3.3 extraite de Angelis et Minnaja (1991) représente la réponse différentielle relative pour le SnO_2 en fonction de la température pour certains gaz afin d'illustrer le comportement de ces capteurs et le problème de sélectivité.

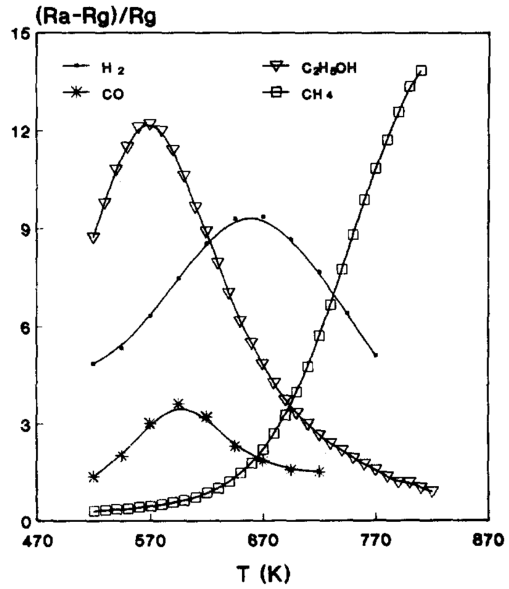


FIGURE 3.3 – Réponse différentielle relative pour le SnO_2 en fonction de la température — d'après Angelis et Minnaja (1991).

En outre, concrètement, pour que la couche sensible atteigne la température donnée à laquelle la réaction cible se produit, le capteur dispose d'un système de résistance chauffante dont la température est contrôlée à l'aide de deux électrodes. Ces nouvelles électrodes sont séparées électriquement des électrodes de mesure à l'aide d'une couche d'isolation.

De nombreuses technologies sont développées pour assurer un chauffage homogène de la couche sensible à une consommation minimale. La forme de la plateforme accueillant la résistance chauffante ainsi que celle de la résistance chauffante (quadrillage, spirale...) sont très étudiées. Les deux structures de plateformes les plus courantes sont les plateformes sur substrat (MOx classique) et les plateformes sur membrane (MOx microhotplate). La Figure 3.4 extraite de l'état de l'art de Liu et *al.* (2018)) présente ces deux types de plateformes.

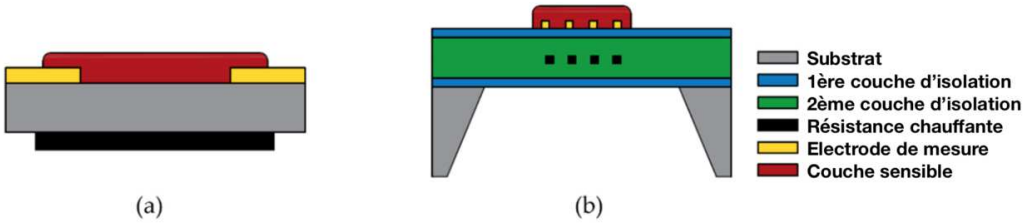


FIGURE 3.4 – Schémas vus de côté a) d'un capteur MOx classique, b) d'un capteur MOx microhotplate — d'après Liu et *al.* (2018).

Les plateformes chauffantes sur substrat sont plus simples : le substrat sert de support isolant ; la couche sensible est d'un côté, la résistance chauffante de l'autre. Les plateformes chauffantes sur membrane sont composées d'un substrat qui supporte une membrane sur laquelle est déposée la résistance chauffante, une couche de passivation pour isoler électriquement et la couche sensible. L'avantage de ce type de structure est de ne pas chauffer au travers du substrat : elle nécessite donc moins de courant et la diffusion de la chaleur est plus homogène.

Pour une couche sensible donnée, Korotcenkov (2007) distingue deux modes de fonctionnement du chauffage. Le premier à basse température ($2\,00 - 500\,°C$) est très sensible à l'humidité et a un temps de réponse plus long, le seconde à haute température ($4\,00 - 700\,°C$) est peu sensible pour le gaz cible et est moins fiable.

Enfin, les erreurs de mesure attendues sont soit liées à une accumulation d'électrons sur la couche sensible, ce qui entraîne une dérive, soit liées à la structure de la couche sensible. Des défauts nanoscopiques ponctuels ou linéiques de la couche sensible peuvent engendrer des biais entre les différents capteurs d'une même série de production.

3.2.3 Le capteur MiCS-4514

Nous choisissons d'étudier le capteur MiCS-4514, un capteur MOx de NO_2 et CO. Le capteur MiCS-4514 est un capteur de pollution de l'air dédié aux applications automobiles (par exemple, pour déclencher l'ouverture et la fermeture du circuit d'air) et ne prétend pas mesurer précisément la concentration des polluants ambiants. Ce capteur a été développé de manière à identifier le type de véhicule par rapport à son émission. En

effet, le diesel émet plus de NO_2 que les véhicules à essence. Parallèlement les essences émettent davantage de CO.

Ces conditions de fonctionnement sont adaptées à notre utilisation en ville : l'humidité acceptable est comprise entre 5 % et 95 %, la température acceptable est comprise entre −30 et 85 °C.

La consommation du capteur est principalement due au chauffage des éléments sensibles. La puissance nominale pour le capteur de CO est de 83 mW afin de chauffer la couche sensible à 360 °C et la puissance nominale pour le capteur de NO_2 est de 43 mW afin de chauffer la couche sensible à 220 °C. Ce capteur chauffe donc à basse température et nous nous attendons à ce qu'il soit sensible à l'humidité.

Dans un air à approximativement 20 °C, les couches sensibles chauffent en environ 30 secondes puis le capteur à un temps de réponse d'environ 10 secondes. De plus, deux modes d'utilisation sont proposés : chauffage continu et chauffage pulsé. Dans le cas du mode pulsé, le signal est un signal créneau de période de l'ordre de la minute. Il permet d'économiser de l'énergie en ne chauffant pas en permanence et de favoriser la réinitialisation des propriétés de la couche sensible par recuit. Néanmoins, le mode continu permet d'effectuer des mesures à environ 1 Hz, soir 60 fois plus fréquentes environ.

Pour interpréter la réponse du capteur, la valeur de la résistance de la couche sensible R_s est généralement normalisée par la valeur de référence R_0 de l'air ambiant en absence de pollution.

La Figure 3.5 présente les relations qui existent entre la réponse du capteur et la concentration réelle de plusieurs gaz. Nous observons une meilleure sélectivité (moins d'interférences avec d'autres gaz) pour le NO_2 que pour le CO. Cependant, elle concerne des températures nominales pour les résistances chauffantes différentes de celles nouvellement recommandées par le fournisseur.

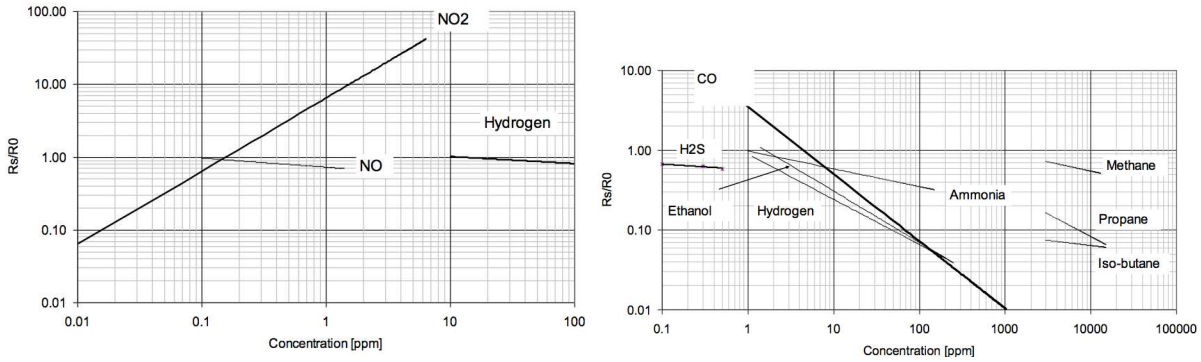


FIGURE 3.5 – Sélectivité du MiCS-4514 à 25 °C, 50 % RH.

Celui-ci fournit de nouvelles courbes d'étalonnage, présentées Figure 3.6, uniquement pour les concentrations en gaz cibles, le NO_2 et le CO. Dans la section 3.5.1, nous tentons de retrouver ces courbes par analyse en laboratoire du capteur. Si tel est le cas, nous pourrons les réutiliser pour estimer la concentration réelle en polluants.

Selon la fiche technique, l'intervalle de détection du CO est compris entre 1 et 1000 ppm, celui du NO_2 est compris entre 0,05 et 10 ppm.

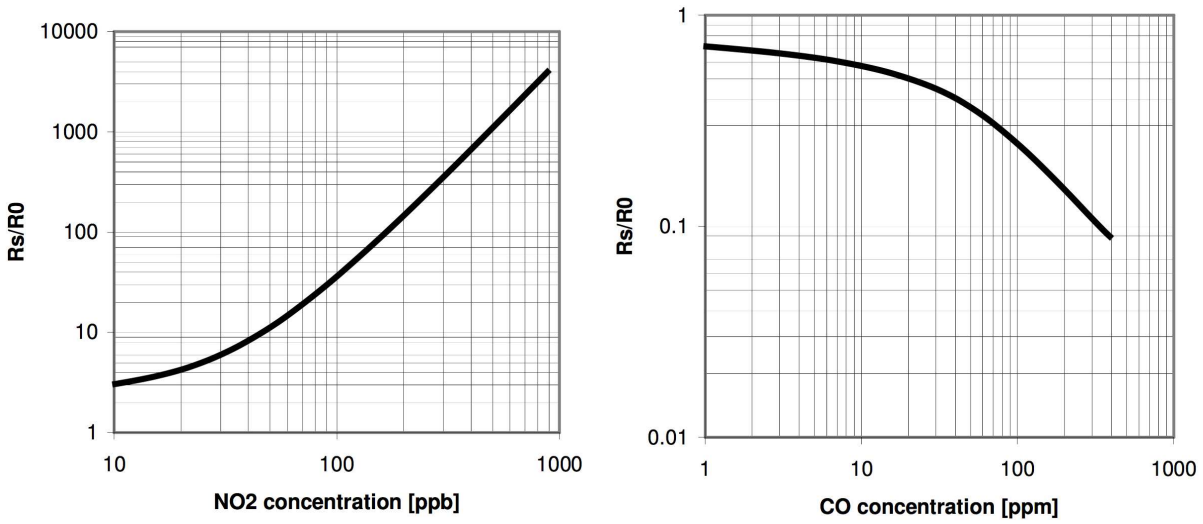


FIGURE 3.6 – Courbes d'étalonnage fournies par MiCS-4514 à 25 °C, 40 % RH.

3.3 Prototypage

3.3.1 Analyse du besoin

Formellement, notre besoin est de caractériser la pollution de l'air en ville. Nous retenons comme solution d'équiper une flotte de vélos de MiCS-4514. La fonction principale de cette flotte de vélos est de collecter des mesures caractéristiques de la pollution automobile. Pour cela, le capteur nécessite un système de stockage, de communication et d'alimentation.

De plus, dans le cas idéal, de nombreux vélos de la ville sont équipés et partagent leurs mesures. Pour ce faire, il faut que le système soit robuste (pour résister à n'importe quel type d'utilisateur) et que son utilisation soit très simple. Le cycliste doit se servir du vélo sans avoir à intervenir sur l'instrument, sauf éventuellement brancher un câble après usage.

En outre, en milieu urbain, si nous voulons laisser le système sur le vélo pour minimiser l'effort d'utilisation, il faut qu'il soit discret, pour éviter les vols et la dégradation, et qu'il résiste aux intempéries.

De surcroît, la mobilité contraint l'encombrement et le poids du système. Elle impose également de déterminer la position afin d'enregistrer une mesure. Enfin, elle perturbe les mesures à cause de la variation d'intensité du flux d'air sur le capteur.

Le projet dans lequel s'inscrit cette solution et le nombre de vélos équipés visé contraignent le coût et le temps de développement. Les contraintes ainsi identifiées et les solutions retenues sont résumées dans la Table 3.2.

En analyse du besoin, une fonction complémentaire est une fonction qui facilite, améliore, ou complète le service rendu. Nous ciblons trois fonctions complémentaires. La première est d'enrichir les mesures (position, pollution) à l'aide d'autres mesures. En l'occurrence, l'utilisation d'un GPS couplé à un accéléromètre peut permettre de mieux reconstruire le trajet et les données météorologiques sont cruciales pour l'interprétation

des mesures des MOx, notamment parce que les MOx sont sensibles à l'humidité.

La deuxième est la validation de la spatialisation de l'information à l'aide d'un réseau mobile, en s'épargnant l'étalonnage du capteur (nécessaire pour les MiCS-4514). Dans un premier temps, cette étude peut être effectuée grâce à un capteur réputé fiable et précis pour un phénomène statique (structure de la ville à l'aide du GPS (Joo et *al.*, 2015) ou de l'accéléromètre) et dans un second temps pour un phénomène dynamique (météorologie). La troisième est de pouvoir connecter facilement un nouveau capteur pour qu'il réutilise le système développé (alimentation, communication, stockage). Cela permet d'une part de tester un capteur *in situ*, et d'autre part de généraliser une méthode de traitement à d'autres types de capteurs voire d'informations. Nous pourrions imaginer utiliser les mêmes méthodes de spatialisation pour le niveau de bruit ou d'éclairage urbain que pour la pollution de l'air.

Table 3.2 – Contraintes du projet et solutions retenues.

	Contrainte	Solution
Système	stockage	carte micro-SD
	communication	micro-USB, BLE, LoRa
	alimentation	dynamo ou batterie
Utilisateurs	robustesse	boîtier de protection
	facilité d'utilisation	solution de bout en bout
Milieu urbain	discrétion	boîtier noir
	résistance aux intempéries	boîtier étanche
Mobilité	poids et encombrement	micro-capteurs
	acquisition de la position	récepteur GPS
	variation du flux d'air sur le capteur	chambres de tranquillisation
Projet	coût	capteurs low-cost
	temps de développement	réduction des ambitions

3.3.2 Réalisation du prototype

L'instrument de mesure a été réalisé en collaboration avec une société de service, *Epurtek*[3]. Elle est spécialisée dans la transition écologique, et plus particulièrement dans les énergies renouvelables, le traitement des déchets, et le traitement de l'eau. Cette startup toulousaine de quatre associés conseille, coordonne et développe des projets orientés autour d'outils d'acquisition de données de procédés de l'environnement.

Nous leur avons délégué la conception et la production du circuit imprimé, ainsi que le développement de fonctionnalités logicielles bas niveau pour l'utilisation des composants matériels. Cette collaboration a été initiée avant le début de cette thèse.

De notre côté, nous nous sommes particulièrement concentrés sur la compréhension du fonctionnement des capteurs, l'évaluation des performances des capteurs embarqués, la fabrication d'un boîtier pour accueillir le circuit imprimé et l'ordonnancement des tâches – pour la gestion de l'acquisition des données des capteurs, pour la gestion de l'énergie et pour la gestion du stockage des données et de la communication. Néanmoins, des problèmes techniques ont nécessité que nous poursuivions le développement des fonctions bas

3. `https://epurtek.fr/`

niveau et effectuions quelques modifications matérielles. Ces modifications ont été réalisées conjointement avec Christophe Zanon, ingénieur au LAAS-CNRS, pendant environ un an.

3.3.2.1 Architecture matérielle

Pour répondre à la problématique énergétique et à celle du temps de production, un compromis est trouvé entre une solution qui n'utilise que des composants sur étagères, à la manière d'une Raspberry Pi, et d'une solution *à partir de zéro*. En effet, le prototypage de la part d'*Epurtek* s'est articulé autour du MSP430, un microcontrôleur à très basse consommation d'énergie réalisé par Texas Instruments (TI), le même que celui de la Raspberry Pi mais à usage dédié et donc épuré. Il dispose de plusieurs modes de fonctionnement (marche, veille, arrêt) afin de réduire la consommation. Le choix des autres capteurs que le MiCS-4514 a été également guidé par la consommation et la taille. Les capteurs supplémentaires sont un récepteur GPS, un accéléromètre et un BME280, multi-capteur de pression (barométrique absolue), température, humidité.

Ainsi, le système est composé d'un microcontrôleur à très basse consommation, le MSP430, qui permet d'effectuer des calculs simples (multiplicateur 32bits sans système d'exploitation) et de communiquer avec différents composants par interruptions. L'architecture matérielle est schématisée Figure 3.7.

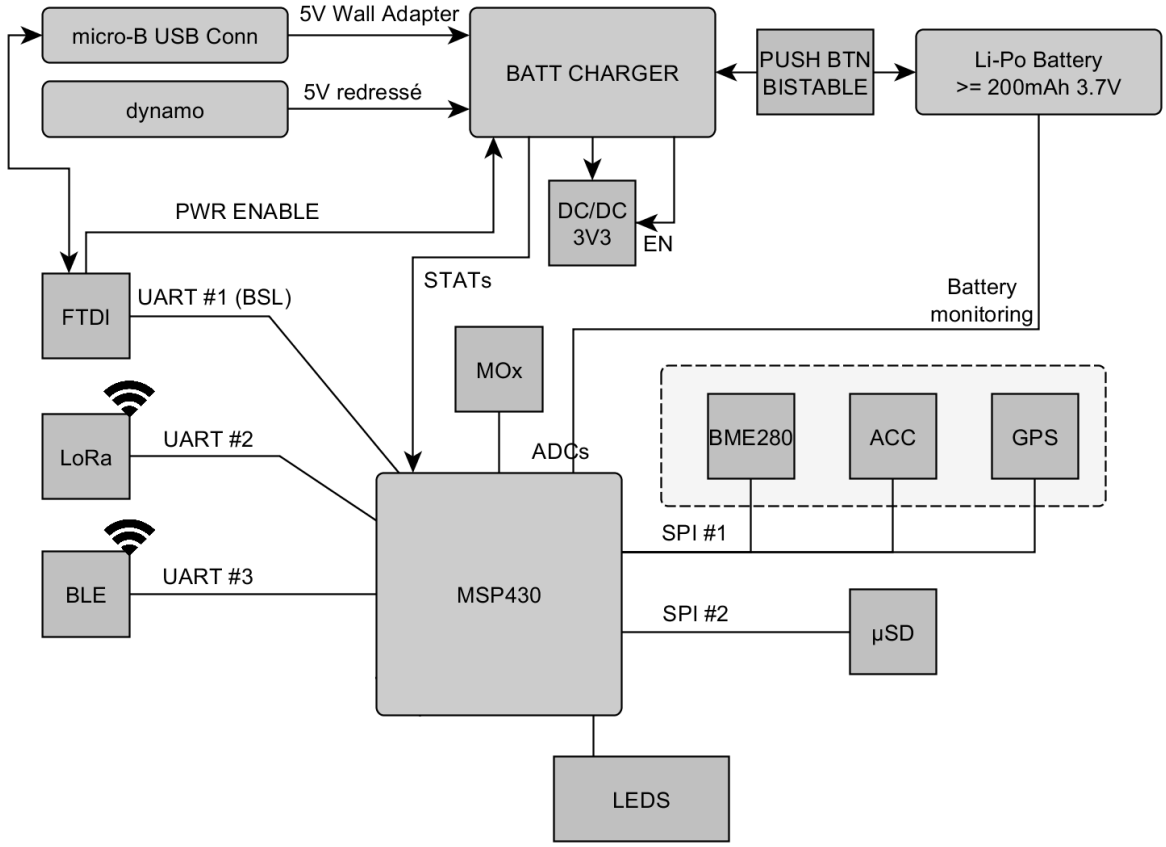


FIGURE 3.7 – Architecture matérielle.

Le port UART (Universal Asynchronous Receiver Transmitter) ou le port micro-USB peuvent servir à accueillir un module complémentaire disposant de nouveaux capteurs à tester tout en réutilisant les capacités de sauvegarde et de communication de notre système. Ces deux types de communication sont couramment utilisés par les systèmes micro-électroniques. La puce FTDI permet de passer de l'un à l'autre.

Avec Roberto Pasqua, nous avons encadré des étudiants de l'INSA Toulouse sur un projet s'articulant autour de l'utilisation du port UART. L'objectif était de connecter un module complémentaire disposant de capteurs et d'afficher les informations recueillies sur un site web en ligne. Ils ont ainsi pu réaliser une preuve de concept pour les capteurs d'ozone MiCS-2614 et Alphasense-O3 et de monoxyde de carbone Alphasense-CO.

Le port micro-USB sert également d'alimentation en courant continu. Le système peut être aussi alimenté par un bornier en alternatif. Le tout est couplé à une batterie tampon (pour éviter les variations d'intensité) et un système de mesure de charge.

Les DELs [4] servent à identifier rapidement l'état interne du système (pour le débogage ou indiquer la fin de l'initialisation). Le Bluetooth Low Energy (BLE) facilite en outre la communication avec le système en mouvement, ce qui est utile au débogage. De plus, dans le cas de nombreux vélos équipés, le BLE permet d'envisager des cas de *Rendez-Vous* (cf. section 1.3.2) et de ré-étalonnage en temps réel. La communication par LoRa, un système de communication longue portée à basse consommation d'énergie, permet d'imaginer un « Internet of Bikes », où les vélos s'échangent les données collectées, par communication points à points, et les font remonter à un serveur central en temps réel, sans avoir à s'y rendre.

Les composants sont récapitulés Table 3.3.2.2.

Table 3.3 – Composants de notre système embarqué.

Composant	Modèle
Micro-capteur de pollution de l'air	MiCS-4514
Micro-capteur des conditions météo.	BME280
Micro-capteur de géolocalisation	GPS A2235-H
Micro-capteur d'accélération	ST LIS2DH12
Moyen de communication	micro-USB, UART, BLE, LoRa, DELs
Système de stockage	carte micro-SD 2Go
Système d'alimentation	micro-USB (continu) ou bornier (alternatif) et batterie tampon
Générateur de courant	dynamo ou batterie

Le circuit imprimé (Figure 3.8) présente ces principaux composants, ce qui permet d'appréhender leur position et leur taille relatives. Si nous devions faire un nouveau circuit, nous éloignerions le capteur de température du capteur MOx, qui chauffe et perturbe les mesures.

Une version intermédiaire du système embarqué a été présenté deux mois après le début de la thèse, ce qui nous a permis de prendre en main la plateforme finale plus rapidement et de procéder aux premiers retours. Puis, quatre mois plus tard, vingt-quatre exemplaires

4. Diode ÉlectroLuminescente ou LED en anglais.

ont été livrés. Le code source associé à certains composants (BME280, BLE, MiCS-4514) a été finalisé jusqu'à trois mois après livraison des exemplaires.

Le système embarqué final coûte en moyenne 150 euros à l'unité et tient dans une main (photographie Figure 3.9). Ainsi, le coût et l'encombrement de notre système respectent les niveaux de contraintes que nous visions (cf. section 2.2).

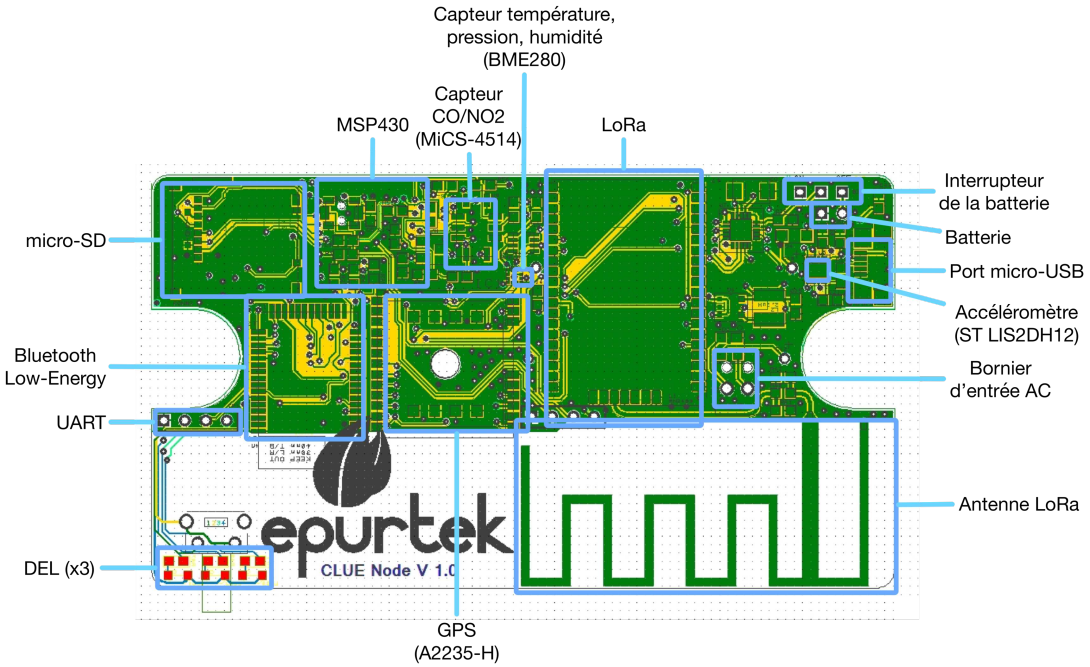


FIGURE 3.8 – Circuit imprimé.

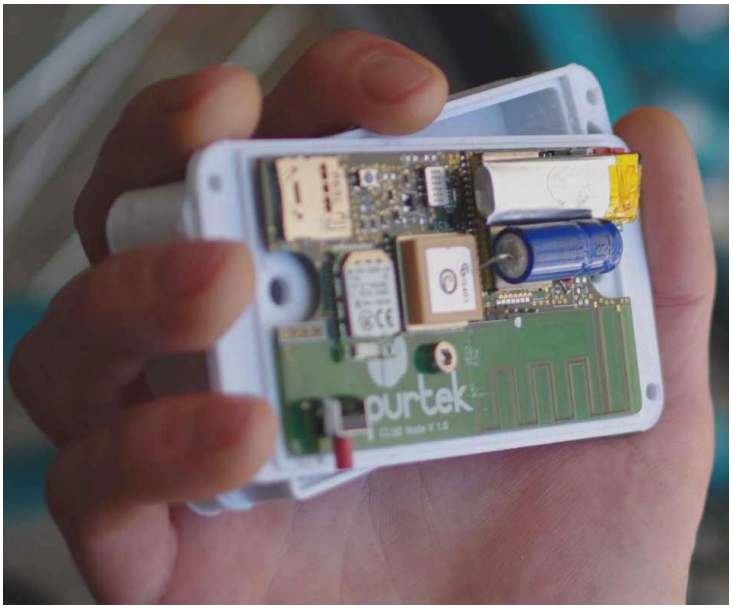


FIGURE 3.9 – Photographie du prototype final.

3.3.2.2 Architecture logicielle

Le code source se décompose en modules : un pour chacun des composants matériels (cf. Table), une console système pour unifier les entrées/sorties, et une machine à état pour la gestion des tâches des modules. Ces modules suivent le patron de conception *pont* (Gamma et *al.*, 1993) qui scinde le code en deux niveaux d'abstractions.

Le bas niveau permet la gestion des différentes tâches du module, dont notamment l'interaction avec le composant physique associé par déplacement de pins. Par exemple, dans le cas du module GPS, les tâches possibles sont l'allumage, l'arrêt ou la mise en veille du composant, la configuration des données récupérées, la mise à jour de la date ou de la position.

Le haut niveau, à la manière du patron de conception *façade* (Gamma et *al.*, 1993), permet une gestion identique des modules par des interfaces similaires : une interface dispose uniquement de deux fonctions, sans paramètre et avec code de retour, qui sont la fonction app_X_init, pour l'initialisation, et la fonction app_X_task, pour l'exécution des tâches qui lui incombent. Les tâches d'un module dépendent de booléens qui lui sont propres, modifiés soit par une interruption du composant physique associé, soit par la machine à états. Une exception est faite pour la console qui est pourvue d'une fonction supplémentaire de communication afin de pouvoir émettre un message, depuis n'importe où pour le débogage et uniquement dans la machine à état après déploiement.

Ainsi, la fonction *main* est une succession d'initialisations de chaque module, puis une boucle infinie qui appelle chacune des fonctions app_X_task, et enfin un mécanisme de watchdog afin de redémarrer le système en cas de blocage dans un état plus d'un certain temps. A la fin d'un tour de boucle, si plus aucune tâche n'est en attente, le MSP430 rentre en veille et ne se réveille qu'à la réception d'une nouvelle interruption, condition nécessaire à l'apparition de nouvelles tâches.

La machine à état fournie, très simple, ne servait que pour la preuve de concept.

3.3.3 Modifications apportées

3.3.3.1 Difficultés matérielles rencontrées

Des modifications matérielles ont également été apportées pour le bon fonctionnement du système. En effet, pour comprendre la grande variabilité de consommation énergétique, nous avons utilisé l'outil EnergyTrace de CCS qui permet de suivre la consommation de courant. Par différence de code avec et sans BLE, nous avons détecté la présence de deux résistances soudées à tort par le sous-traitant d'*Epurtek*, engendrant un court-circuit. De plus, nous avons détecté de mauvaises soudures du GPS sur certains exemplaires du système. Enfin, l'écriture systématique sur les secteurs en tête de la micro-SD provoque une usure très rapide.

En outre, la première version du système communiquait en LoRa avec une passerelle déployée par deux stagiaires de l'INSA Toulouse, Guillaume Hortes et Paul Charayron, encadrés par Pascal Berthou, maître de conférences au LAAS-CNRS, six mois avant le début de cette thèse. L'adaptation de la communication entre la seconde version et cette passerelle a été laissée à notre charge. Néanmoins, lorsque la connexion fut établie, en vue d'un déploiement d'une dizaine de passerelles pour couvrir Toulouse, nous avons

commandé la micro-puce utilisée pour la passerelle, mise à jour par le fabricant, et dont le protocole avait été modifié. De plus, seulement un exemplaire de la seconde version du système disposait d'un récepteur LoRa, l'installation des autres a été laissée à notre charge par *Epurtek*. Bien qu'une étude préliminaire pour maximiser le nombre d'informations reçues a été menée, tout en respectant le duty cycle du LoRa et la probabilité de non réception du message, nous n'avons pas poursuivi l'établissement de la communication en LoRa.

Enfin, le port micro-USB de nombreux exemplaires s'est dessoudé à force d'utilisation, lors du développement et du déploiement. Il aurait fallut prévoir une structure permettant d'assurer sa fixation au circuit électronique qui tienne compte de la pression exercée pour connecter le câble.

3.3.3.2 Difficultés logicielles rencontrées

Pour modifier le code source embarqué, en langage C, nous avons utilisé le kit de développement (cf. Figure 3.10) et l'IDE Code Composer Studio (CCS) – une extension d'Eclipse – fournis par TI. Sans rentrer dans le détail, des modifications ont été nécessaires pour assurer une certaine fiabilité au système.

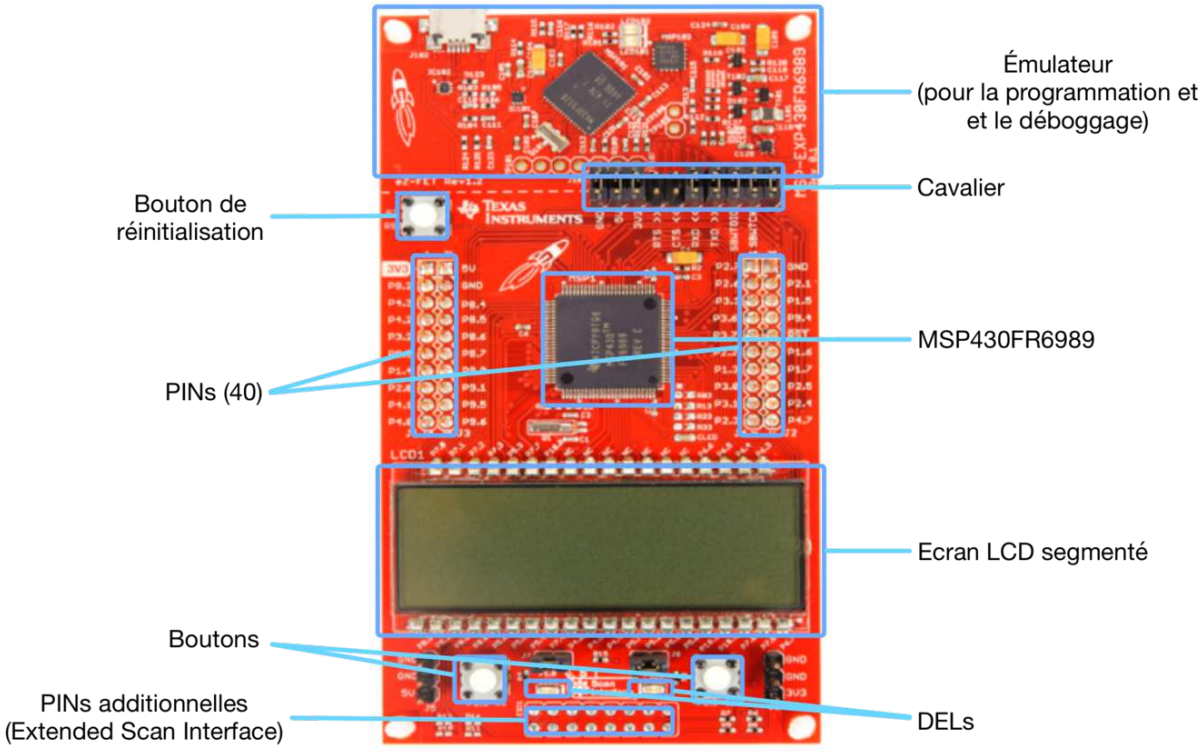


FIGURE 3.10 – Kit de développement TI du MSP430FR6989.

Entre autres, nous avons ajouté des fonctionnalités d'entrées/sorties dans la console et des fonctionnalités d'utilisation des DELs pour faciliter le débogage, développé une application mobile pour visualiser les entrées/sorties – sur la base du code open source de l'application associée au BLE – et mis en place des mécanismes pour assurer une certaine fiabilité (gestion des codes d'erreur, ajustement du watchdog, non dépassement de la pile).

Nous avons également vérifié le fonctionnement de chaque module indépendamment. Cela nous a permis de détecter une division par 0, des problèmes de cast d'entiers, la nécessité d'un paramétrage de l'accéléromètre par exemplaire du capteur, un problème de pointeurs du buffer, des soucis d'index pour l'écriture sur la carte micro-SD, des confusions dans les pins à monter ou descendre, un problème dans la gestion des interruptions, une fréquence de mise à jour trop faible pour les informations du système d'alimentation et de l'ADC mais aussi, de revoir les calculs fournissant les valeurs du MiCS-4514, et enfin d'optimiser l'échange de messages avec le GPS pour la synchronisation aux satellites et utiliser le mode veille en sauvegardant la dernière position connue (pour faciliter la réacquisition future de la position) ; ce qui a évité des redémarrages intempestifs par le watchdog.

3.3.4 Fonctionnement final

Ordonnancement des tâches et gestion de l'énergie L'ordonnancement des tâches du système est géré par une machine à état (cf. Figure 3.11). Elle est composée de trois états qui correspondent aux modes de fonctionnement de l'instrument de mesure : normal, veille, critique.

Les conditions de transition entre les états sont définies en fonction de la charge énergétique (suivant un mécanisme d'hystérésis), et éventuellement du déplacement du système (détecté à l'aide de l'accéléromètre). L'état critique assure que le système ne s'éteigne pas brusquement et endommage les composants. Le mode veille est un état intermédiaire : en cas de diminution de la charge énergétique (lors d'un arrêt du cycliste à un feu rouge par exemple), le système n'utilise que les fonctions vitales, afin de s'épargner le plus que possible la perte de temps d'une réinitialisation du système. En effet, lorsque le système s'initialise, le temps de synchronisation du GPS (pour obtenir date et position) n'est pas négligeable, et peut atteindre la dizaine de minutes lorsque le ciel est couvert. L'état normal exécute en sus une série de tâches liées à la mesure. Le traitement de cette série de tâches est présenté Figure 3.12.

Ces tâches sont des appels à des fonctionnalités d'un module (cf. section 3.3.2.2). Elles sont soit déclenchées périodiquement, soit par un autre évènement (une autre tâche ou une transition de la machine à état). Les tâches implémentées sont décrites dans la Table 3.4. Les mesures des capteurs sont effectuées toutes les secondes.

Ces appels périodiques sont paramétrés dans un tableau, dynamiquement modifiable via la console en BLE notamment. Ces attributs sont :

- les états de l'instrument de mesure dans lesquels la tâche est appelée,
- la période entre deux appels de la tâche,
- la dépendance à la date et à la position GPS, pour les informations de mesure notamment,
- la dépendance à des composants physiques,
- la journalisation [5] de la tâche via la console.

A chaque transition, si de nouveaux composants physiques sont requis par des tâches du nouvel état, ils passent en mode actif, sinon, s'ils ne sont plus utilisés, ils passent en

5. Enregistrement séquentiel et daté d'un évènement.

mode veille (ou éteint si l'initialisation est rapide).

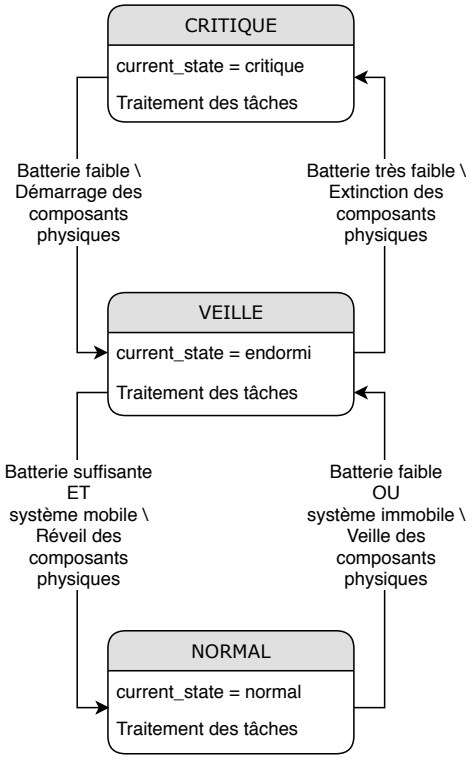


FIGURE 3.11 – Machine à état décrivant le fonctionnement de notre instrument de mesure.

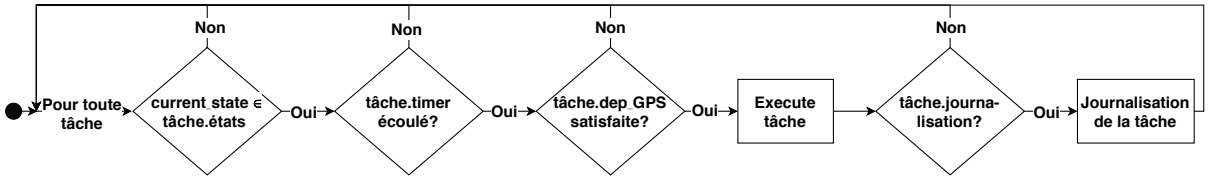


FIGURE 3.12 – Traitement des tâches.

TABLE 3.4 – Tâches implémentées.

Tâche	Description
checkMvt	Détection de la mobilité du système
logGPS	Récupération des données du GPS (date, position)
logBATT	Récupération de l'état de charge de la batterie
logACC	Récupération des données de l'accéléromètre (1 s à 32 Hz)
logBME	Récupération des données du BME280
logMOx	Récupération des données du MiCS-4514
heatCO_START	Allumage du chauffage de la couche sensible du CO
heatfCO_STOP	Arrêt du chauffage de la couche sensible du CO
heatNOx_START	Allumage du chauffage de la couche sensible du NO_x
heatNOx_STOP	Arrêt du chauffage de la couche sensible du NO_x
logDEBUG	Journalisation d'informations à définir pour déboguer

Enfin, la généricité de l'architecture permet d'autres usages. Une instance dédiée à la détection de trous sur la chaussée, par transformée de Fourrier, a été mise en œuvre par un groupe d'étudiants de M1 à l'Université Paul Sabatier, encadré par Nicolas Rivière, maître de conférences au LAAS-CNRS.

Gestion des capteurs et journalisation La machine à état ainsi définie permet une acquisition générique des données des capteurs via les tâches log(*) avec une fréquence d'échantillonnage paramétrable. La notation $x(*)y$ désigne tout objet dont le nom a pour préfixe x et pour suffixe y. Ici, log(*) désigne donc logGPS, logBATT, logACC, logBME, logMOx, logDEBUG. Chaque acquisition peut être journalisée ou transmise par le BLE. La Table 3.5 présente les paramètres des tâches implémentées.

Table 3.5 – Paramètres des tâches implémentées.

Tâche	États	Période	GPS	Dépendances	Journalisation
checkMvt	Veille, Normal	1 s	Non	ACC	Non
logGPS	Normal	1 s	Non	GPS	Oui
logBATT	Normal	1 min	Non	∅	Oui
logACC	Normal	1 s	Oui	ACC, GPS	Oui
logBME	Normal	1 s	Oui	BME280, GPS	Oui
logMOx	Normal	1 s	Oui	MiCS-4514, GPS	Oui
heatCO_START	∅	NC	NC	MiCS-4514	Oui
heatfCO_STOP	∅	NC	NC	MiCS-4514	Oui
heatNOx_START	∅	NC	NC	MiCS-4514	Oui
heatNOx_STOP	∅	NC	NC	MiCS-4514	Oui
logDEBUG	Critique, Veille, Normal	1 s	Non	∅	Oui

Les tâches heat(*) permettent de gérer les chauffages des couches sensibles du capteur MiCS-4514. Elles peuvent être déclenchées lors des transitions dans la machine à état (mode continu) ou périodiquement (mode pulsé).

La journalisation dépend de la tâche. Le format d'une inscription est le suivant [6] :

T ⟶ ID ⟶ ...

avec T la date d'exécution de la tâche, ID l'identifiant de la tâche et ... l'ensemble des valeurs de retour. De plus, d'autres évènements sont journalisés. Ils concernent le redémarrage du système, le fonctionnement de la machine à état et les échanges de messages pour l'initialisation des capteurs.

Ces inscriptions sont ensuite transmises via la console. Lorsque le système est déployé, elles sont uniquement écrites sur la carte micro-SD.

6. La notation « ⟶ » désigne une tabulation.

3.4 Retour d'expérience

3.4.1 Solution de bout en bout et simplifications

En vue d'un déploiement auprès d'une association de location de vélos, le système final doit être automatiquement géré de bout en bout, de la gestion de l'alimentation en énergie à l'affichage des données collectées. D'une part, cela assure une bon maniement du système par n'importe quel utilisateur, d'autre part, la visualisation des données favorise son adoption.

Pour cela, nous avons mis en place un système portable de communication à installer au lieu de récupération des vélos. Ce système est composé d'une Raspberry Pi 3 connectée à l'Internet par wifi. Il permet le téléversement des données, soit par BLE lorsque le système de mesure est à proximité, soit par USB lorsque le système de mesure est connecté. Ces données sont ensuite formatées et envoyées à notre serveur distant puis sauvegardées dans une base de données. Le formatage des données est présenté au chapitre suivant. Il permet également de mettre à jour à distance le programme embarqué dans le système de mesure lors du branchement en micro-USB.

De plus, la réception de nouvelles données sur le serveur distant met à jour un tableau de bord (cf. Figure 3.13) qui présente visuellement la base de données : nombre total de points, temps total d'acquisition des données, vitesse moyenne, température moyenne, trajets des vélos, cartes des données par capteur, courbes d'évolution temporelle des données des capteurs, etc. Ce tableau de bord dynamique permet de sélectionner une période, un instrument de mesure ou un trajet pour explorer plus particulièrement une partie des données. Il est fait à l'aide de Markdown en R, et publié sur l'Internet à l'aide Shiny.

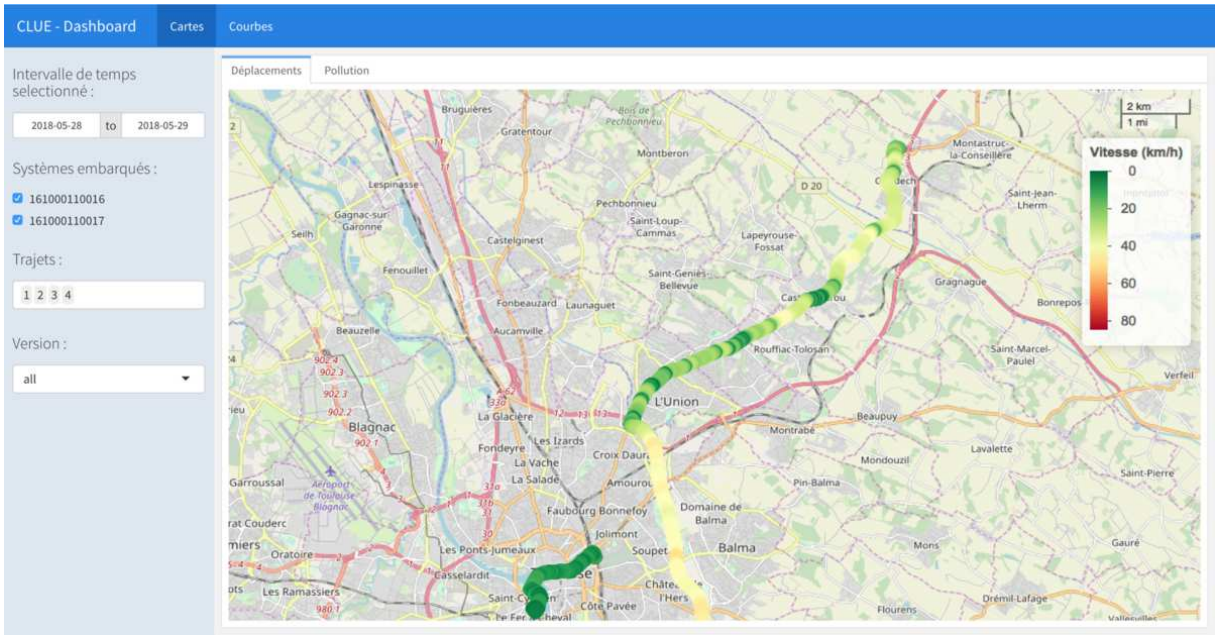


FIGURE 3.13 – Tableau de bord présentant nos données.

Enfin, pour conserver une certaine flexibilité dans le déploiement du système sur un vélo couplé à une dynamo ou à une batterie, nous avons utilisé un fichier de configuration. Ce fichier – défini à l'aide de directives de préprocesseur – indique le pré-traitement à effectuer pour configurer la machine à état en fonction des paramètres, qui varient en fonction du type d'alimentation (batterie ou dynamo) et de son utilisation (station fixe ou mobile). Nous avons également instancié un mode de débogage.

Cette automatisation a été très coûteuse en temps. L'absence totale d'intervention de la part de l'utilisateur, mis à part éventuellement connecter un câble USB, a conduit à gérer de nombreux cas. En conséquence, nous avons décidé de simplifier le prototype initial. Nous avons ainsi abandonné l'utilisation de la dynamo et du LoRa au profit d'un système plus minimal mais plus fiable.

3.4.2 Alimentation : dynamo et batteries

La solution initiale projetait d'utiliser la dynamo pour fournir l'énergie au système via le bornier dédié en alternatif. Pour assurer l'alimentation du système, même en cas de pause (à un feu rouge par exemple, mais aussi idéalement entre deux utilisations), une petite batterie tampon, un accumulateur lithium-ion polymère (Li-Po) de 200 mAh à 3,7 V, est adjointe (4 cm × 1 cm × 0,3 cm). Le choix d'une batterie Li-Po réside dans leur faible auto-décharge, leur haute densité énergétique (comme les accumulateurs lithium-ion), leur robustesse et leur grand nombre de cycles (avantages des Li-Po sur les Li-ion). La charge de la batterie tampon nécessite une dizaine de minutes de vélo – pour fournir le courant nécessaire au bon fonctionnement du MSP430 – à vitesse relativement soutenue pour des vélos de loisir (10–15 km/h) et pâtit du froid de l'hiver.

Avec l'aide de l'atelier micro-électronique du LAAS-CNRS, et plus particulièrement celle de Jérôme Manhes – ingénieur, nous avons remplacé cette batterie par un super-condensateur de 1,5 F à 5 V (2,5 cm × 1 cm × 1 cm), plus rapide à charger, mais qui ne conserve pas la charge énergétique plus de quelques minutes.

Néanmoins, la consommation du MiCS-4514, bien que faible, nécessite également une vitesse minimale du cycliste aux alentours de 10–15 km/h. Cette contrainte nous a conduit à nous réorienter vers des cyclistes entraînés ou à ajouter une autre batterie, plus volumineuse (11 cm × 6 cm × 2 cm) mais de capacité bien plus importante, à charger préalablement.

3.4.3 Réalisation du boîtier et appareillage

Afin d'accueillir le circuit imprimé fourni par *Epurtek*, nous avons confectionné un boîtier. Ce boîtier est fixé à l'arrière de la lampe arrière du vélo (cf. Figure 3.14), et relié à la dynamo en parallèle du feu arrière. Ses intérêts sont la protection des intempéries, la discrétion du système sur le vélo mais surtout la constance de l'écoulement d'air sur le capteur de pollution. En effet, afin d'éviter un flux d'air trop variable – dépendant de la vitesse du vélo – et donc une analyse a posteriori, l'air ne peut s'engouffrer dans le boîtier qu'aux deux extrémités par une ouverture qui débouche sur une chambre.

FIGURE 3.14 – Photographie d'un vélo équipé du boîtier de mesure situé sous le porte-bagage, derrière le feu arrière.

L'utilisation d'une chambre pour amortir un flux d'air est classique en mécanique des fluides, cependant nous n'avons pas poussé l'analyse de flux en laboratoire ou par simulation. En effet, dans l'étude de Unnikrishnan et Vetterli (2013), la corrélation entre la vitesse et la mesure est négligeable.

Pour produire ce boîtier, nous avons été aidé pour la conception par Xavier Dollat, ingénieur au LAAS-CNRS et responsable de l'atelier mécanique ; qui a ensuite imprimé les différentes versions à l'aide d'une imprimante 3D. L'avantage de l'impression 3D est de pouvoir produire rapidement de nombreux boîtiers à moindre coût.

La première version a été imprimée en noir, de la même couleur que la lampe arrière, pour une meilleure discrétion. Nous avons cependant noté un écart entre la température fournie par le BME280 et la température extérieure. Nous avons alors comparé l'impact de l'exposition au soleil sur la température intérieure pour un boîtier noir et un boîtier blanc. Nous avons constaté que la température à l'intérieur du boîtier blanc correspondait à la température réelle mais que celle du boîtier noir été supérieure d'environ 5 °C en un quart d'heure d'exposition en été. Nous nous sommes également questionnés sur l'impact des COV présents dans le plastique utilisé pour l'impression en 3D mais n'avons pu conduire un protocole d'expérimentation. A posteriori, une solution en bois aurait évité certaines indéterminations.

La Figure 3.15 présente le schéma final du boîtier. Sa taille après impression est de 11 cm × 5,75 cm × 2,5 cm.

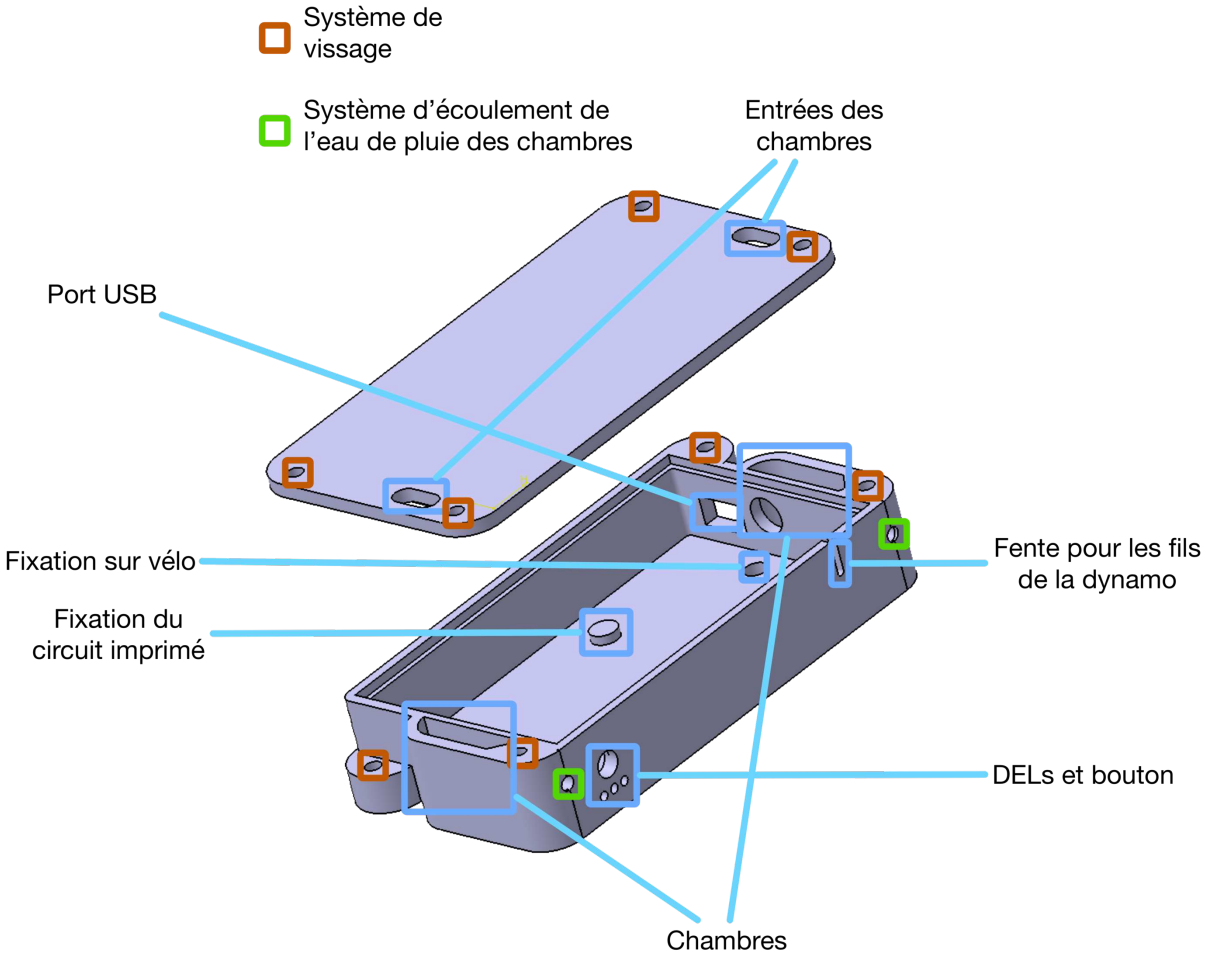


FIGURE 3.15 – Schéma du boîtier.

3.4.4 Synchronisation d'un récepteur GPS en mouvement dans un milieu urbain

Pour déterminer la position d'un récepteur GPS et la date, il faut qu'un minimum 3 satellites soient synchronisés. Lors d'un démarrage à chaud, la mémoire du récepteur contient une position et une date valides. Ces informations facilitent la synchronisation avec les satellites. Lorsque ce n'est pas le cas, le démarrage à froid peut prendre plusieurs dizaines de secondes afin de fournir un position stable (Gong et *al.*, 2012; Schuessler et Axhausen, 2009). Dans notre cas, il aurait pu être intéressant d'utiliser ce mécanisme puisque le récepteur est fixé au vélo et donc sa position est toujours à jour. Néanmoins, comme nous utilisions la dynamo, le système était souvent éteint et la date incorrecte. Pour notre récepteur, le temps d'initialisation est de l'ordre de 5 minutes avec un ciel dégagé et 10 minutes avec un ciel couvert.

De plus, la précision dépend du nombre de satellites supplémentaires, de la synchronisation temporelle entre tous les satellites – assurée par des horloges atomiques –, de la réception des signaux (qualité du récepteur, perturbations de l'environnement, variabilité de la vitesse de l'onde dans l'atmosphère, phénomènes relativistes), de la qualité de l'horloge interne du récepteur et de la précision des calculs par multilatération et des méthodes

de correction d'erreurs. La précision d'un récepteur GPS civil est de l'ordre du mètre.

Notre récepteur GPS est sensible aux perturbations de l'environnement, dont notamment aux « canyons urbains ». Ce sont des lieux où les signaux satellites n'atteignent pas directement l'appareil GPS mais sont renvoyés par des obstacles tels que des immeubles de grande hauteur ou des arbres feuillus. En conséquence, le signal est perdu ou dégradé. A Toulouse, le principal canyon urbain identifié est le long du canal du Midi sous deux allées de platanes. Il s'agit également de la piste cyclable la plus utilisée de la ville. Nous observons également cet effet lors des jours très nuageux.

Certains systèmes, tels que les smartphones, utilisent le réseau Internet pour améliorer la rapidité de la synchronisation du récepteur GPS et la précision de la position. L'inconvénient de ce couplage est le temps de mise en œuvre.

3.5 Évaluation des performances de nos capteurs en situation contrôlée

De nombreuses études rapportent des différences de performance des micro-capteurs entre les conditions de test en laboratoire et les conditions réelles (Mead et *al.*, 2013; Lewis et *al.*, 2015; Spinelle et *al.*, 2017; Castell et *al.*, 2017; Jerrett et *al.*, 2017). Afin d'évaluer les performances de nos capteurs, nous contrôlons d'une part en laboratoire la sensibilité des capteurs d'un système, puis d'autre part, nous intercomparons *in situ* la réponse de quatre capteurs au même endroit et donc sujets aux mêmes conditions environnementales.

3.5.1 En laboratoire

Nous testons en laboratoire la température du capteur de paramètres météorologiques BME280 et le CO du capteur de gaz MiCS-4514 d'un système.

Pour tester le BME280, nous mettons le boîtier dans une étuve, de commande 35 °C puis 45 °C. La Figure 3.16 présente la comparaison de la mesure délivrée par le BME280 dans le boîtier – en bleu – et un capteur de température de référence également placé dans l'enceinte – en vert. Ce capteur de référence n'est pas automatique et un ensemble de points a été relevé manuellement, d'abord de façon rapprochée pendant la première montée en température, puis après une longue période pour observer la stabilisation.

Nous notons un décalage moyen de 1 °C mais les capteurs suivent la même tendance. Leur corrélation est de 0,99. Ce capteur semble donc souffrir d'un faible biais et être très sensible aux variations.

Le test du MiCS-4514 a été réalisé sur le banc instrumenté du LAAS-CNRS, dont une photo est présentée à la Figure 3.17.

L'ouverture des bouteilles de gaz est déclenchée par le débiteur de gaz. Il permet de mélanger, à température ambiante, un ou plusieurs gaz à de l'air pur et de l'eau gazeuse et ainsi contrôler la concentration en gaz et en humidité. La précision des proportions finales de ce mélange dépend des quantités introduites pour chaque gaz, et donc de celle du débiteur de gaz et des concentrations des bouteilles de gaz. Notre débiteur a une marge d'erreur de 5 % et nous utilisons une bouteille de 100 ppm de CO, ce qui fait une marge d'erreur finale de 5 ppm (cf. Table de conversion entre ppm et $\mu g/m^3$ en annexe 4). Nous

ne pouvons ni tester le capteur de NO_2 entre 1 et 10 ppm, car la meilleure précision que nous pouvions obtenir est de 5 ppm, ni observer son comportement, car il sature au delà de 10 ppm.

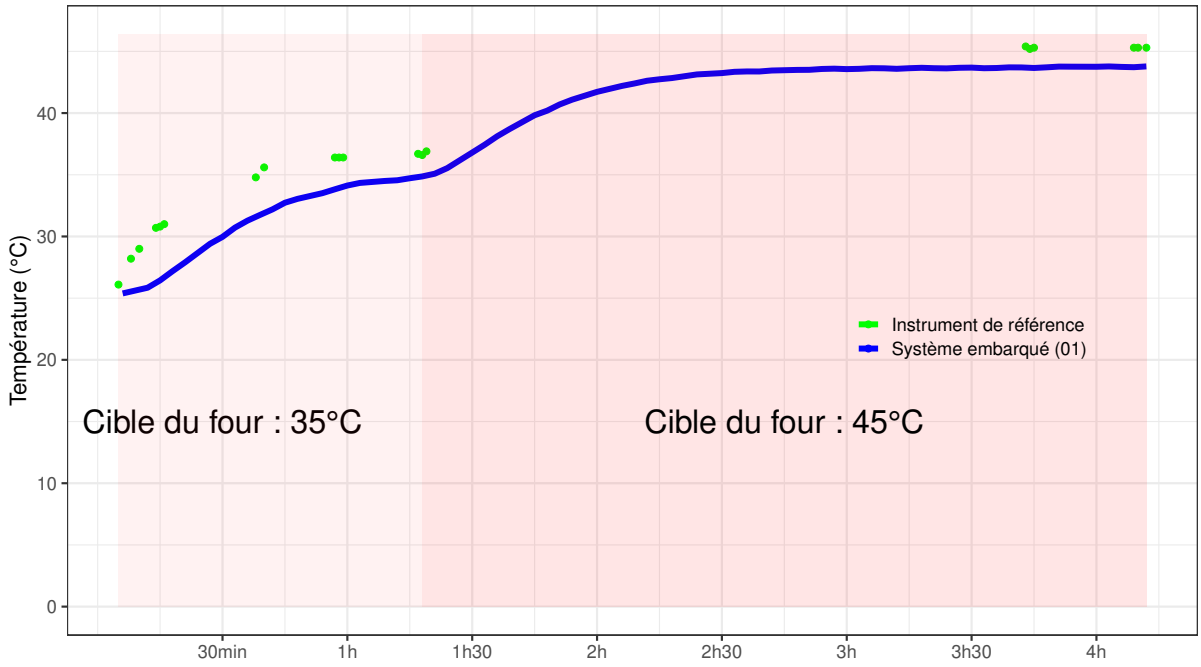


FIGURE 3.16 – Test du boîtier en température.

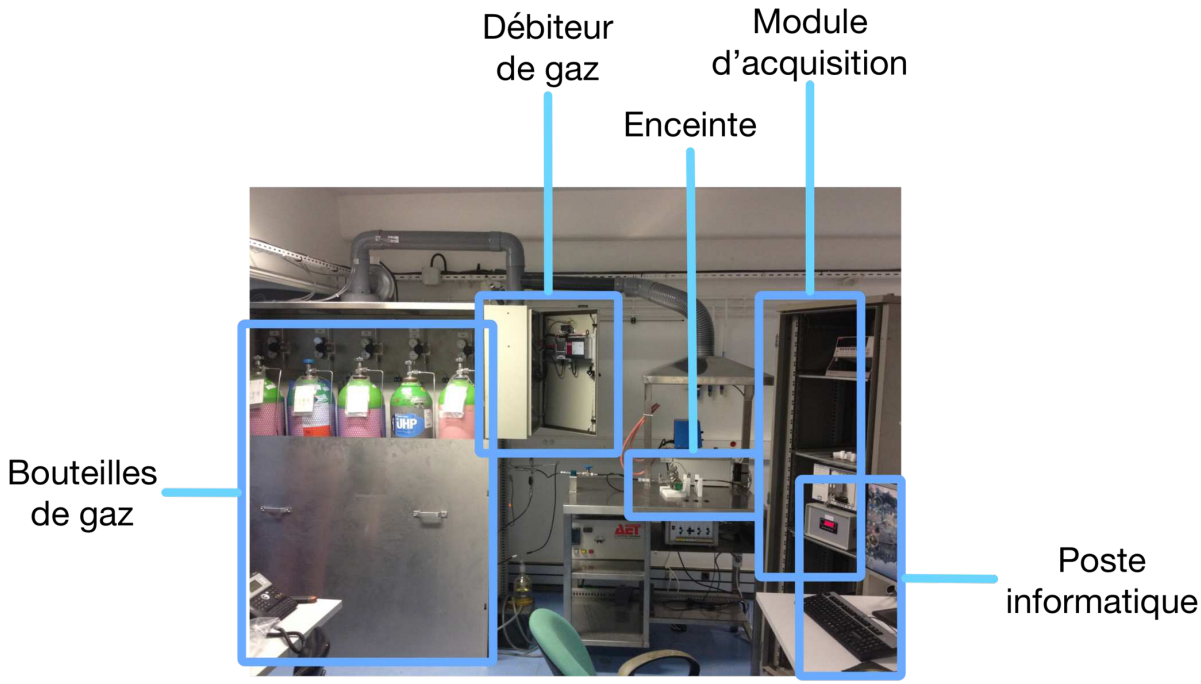


FIGURE 3.17 – Photographie du banc d'essais du LAAS-CNRS.

Le banc est conçu pour tester des capteurs sans système d'acquisition. Ils sont placés dans l'enceinte et les données collectées par le module d'acquisition sont traitées par le système informatique. Afin de tester le système et son boîtier dans son ensemble, nous avons utilisé une enceinte plus grande pour accueillir notre système, conçue à l'aide d'un récipient en verre avec un couvercle en plastique (type Tupperware) rendu hermétique à l'aide de silicone.

Nous étudions la réponse du capteur lorsque nous injectons une quantité prédéterminée de gaz dans dans l'enceinte fermée contenant notre boîtier. La Figure 3.18 montre l'évolution de la résistance du capteur en fonction du temps.

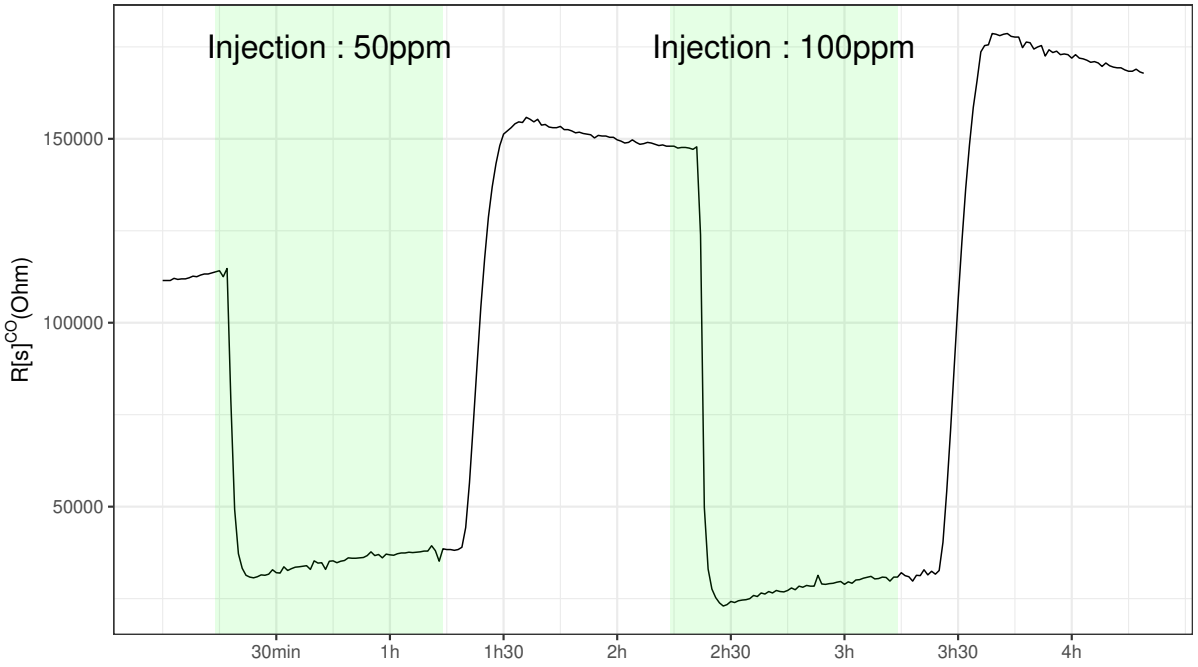


FIGURE 3.18 – Variation de résistance du capteur en fonction de l'injection de CO dans l'enceinte pour des concentrations de 50 ppm et 100 ppm.

En première approche, nous effectuons quatre injections de 1 h à humidité constante de 20 % : la première de 50 ppm de CO, la seconde d'air pur, la troisième de 100 ppm de CO et la quatrième d'air pur. Durant l'expérience, la température dans l'enceinte est constante et égale à 32 °C et la pression est de 995 mbar. Le débit est calculé de façon à ce que l'air dans l'enceinte se renouvelle en 5 minutes.

Nous retrouvons les quatre injections sous forme de paliers retardés d'environ 5 minutes, soit le temps que le gaz pénètre toute l'enceinte. Nous remarquons que le capteur est sensible aux variations de concentration en CO. Selon la fiche technique, la réponse est logarithmique. Néanmoins, les réponses pour 50 et 100 ppm sont proches, respectivement 37 kΩ et 30 kΩ, mais les phases d'air pur ont des réponses décalées. Cela traduit une dérive de la ligne de base de notre capteur d'environ 150 Ω par heure.

Lorsque nous normalisons par la ligne de base, la réponse ne correspond pas à celle indiquée par la fiche technique à la Figure 3.6, ni même à une courbe linéairement dépendante. Il nous est alors impossible d'établir une relation directe entre la réponse du

capteur et la concentration en gaz.

Parallèlement, et comme indiqué dans la Figure 3.19, nous observons que la résistance sur le capteur NO_2 n'est pas affectée. Cependant, nous remarquons un pic de variation lors de la première injection en relation avec une variation brutale de la température, indiquant une sensibilité accrue de ce capteur. Pour une variation de 0,2 °C, la réponse normalisée – sans unité – varie entre 1 et 1,7. Cette relation met en évidence une forte corrélation entre les mesures du MiCS-4514 et du BM280.

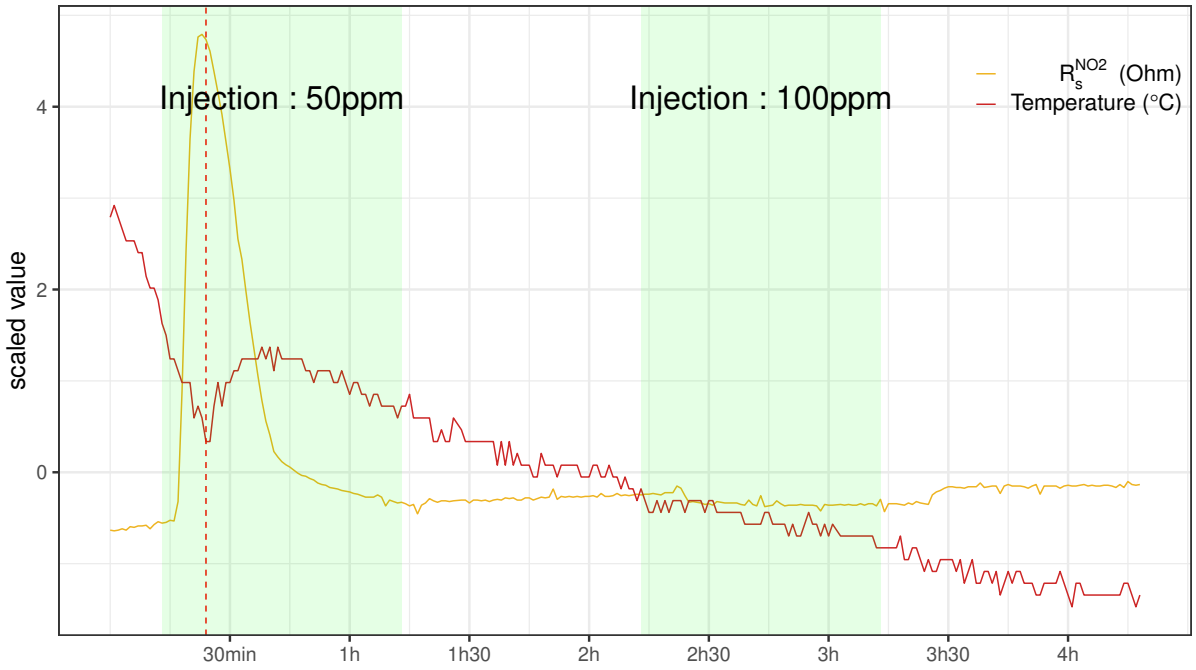


FIGURE 3.19 – Variation de la résistance sur le capteur NO_2 et de la température lors de l'expérience de changement de concentration de CO.

Nous en déduisons que nous ne pourrons pas exploiter la courbe du fournisseur sans un étalonnage complet du système en fonction des paramètres météorologiques. Nous décidons de ne pas pousser davantage l'étalonnage en laboratoire pour nous focaliser sur l'intercomparaison *in situ* de plusieurs systèmes.

3.5.2 *In situ*

Nous voyons dans cette expérience de laboratoire et d'après la littérature actuelle que les capteurs type MICS-4514 sont relativement difficiles d'emploi en raison de leur sensibilité aux paramètres météorologiques et à la fluctuation de leur ligne de base.

Dans une approche *in situ*, nous cherchons à ce que les capteurs mesurent sensiblement la même chose lorsqu'ils sont soumis aux mêmes conditions environnementales, et cela même si les valeurs absolues ne sont pas nécessairement connues.

Afin d'étalonner les capteurs entre eux, nous effectuons deux expériences de comparaisons en 2 lieux différents avec 4 systèmes.

Nous effectuons une première expérience pendant le week-end du 8 décembre 2017 au 11 décembre 2017, sur notre lieu de travail, situé dans un technopole. Les capteurs ont été exposés sur le toit du bâtiment du LAAS-CNRS à environ 7 m de hauteur sous abri. L'air circule de manière naturelle dans le boîtier sans ventilation.

La seconde expérience a eu lieu pendant la semaine du 24 décembre 2017 au 2 janvier 2018, en milieu rural. Les boîtiers ont été installés à un mètre de hauteur sous abri. Le milieu ambiant est très peu influencé par les émissions routières. Un des capteurs, le numéro 20, n'a cependant pas fonctionné durant la première moitié de la seconde expérience.

Les variations de température, pression et humidité enregistrées par les capteurs sont présentées à la Figure 3.20.

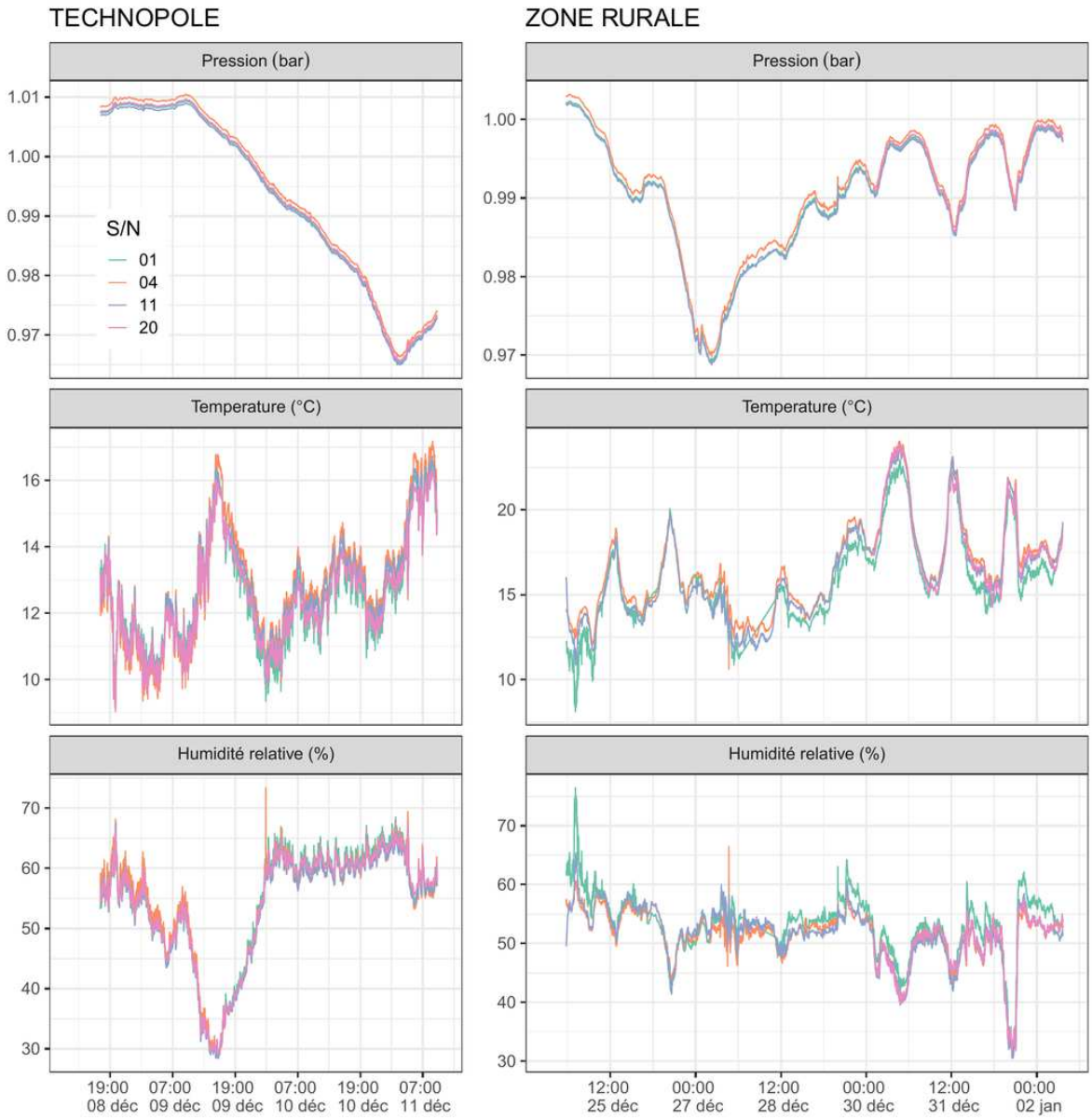


FIGURE 3.20 – Variation de pression, température, humidité enregistrée sur les 4 systèmes intercomparés durant les 2 expériences.

Nous remarquons que les valeurs fournies par le BME280 sont similaires quelque soit le système. Les gammes de température et d'humidité sont sensiblement similaires entre la première et la seconde expérience. Durant la première expérience, nous notons une variation de température et d'humidité importante en fin de journée du 9 décembre. La température est en moyenne de 12,6 °C et varie entre 9 °C et 17 °C. L'humidité relative moyenne est de 54 % et varie entre 28 % et 73 %. Durant la seconde expérience, nous notons trois pics de température, pression et humidité à partir du 30 décembre. La température est en moyenne de 16,5 °C et varie entre 8 °C et 24 °C. L'humidité relative est en moyenne de 51,7 % et varie entre 30 % et 76 %.

Les réponses du capteur MICS-4514 sont présentées à la Figure 3.21. Cette Figure présente la résistance du capteur normalisée par la valeur médiane de la série de données.

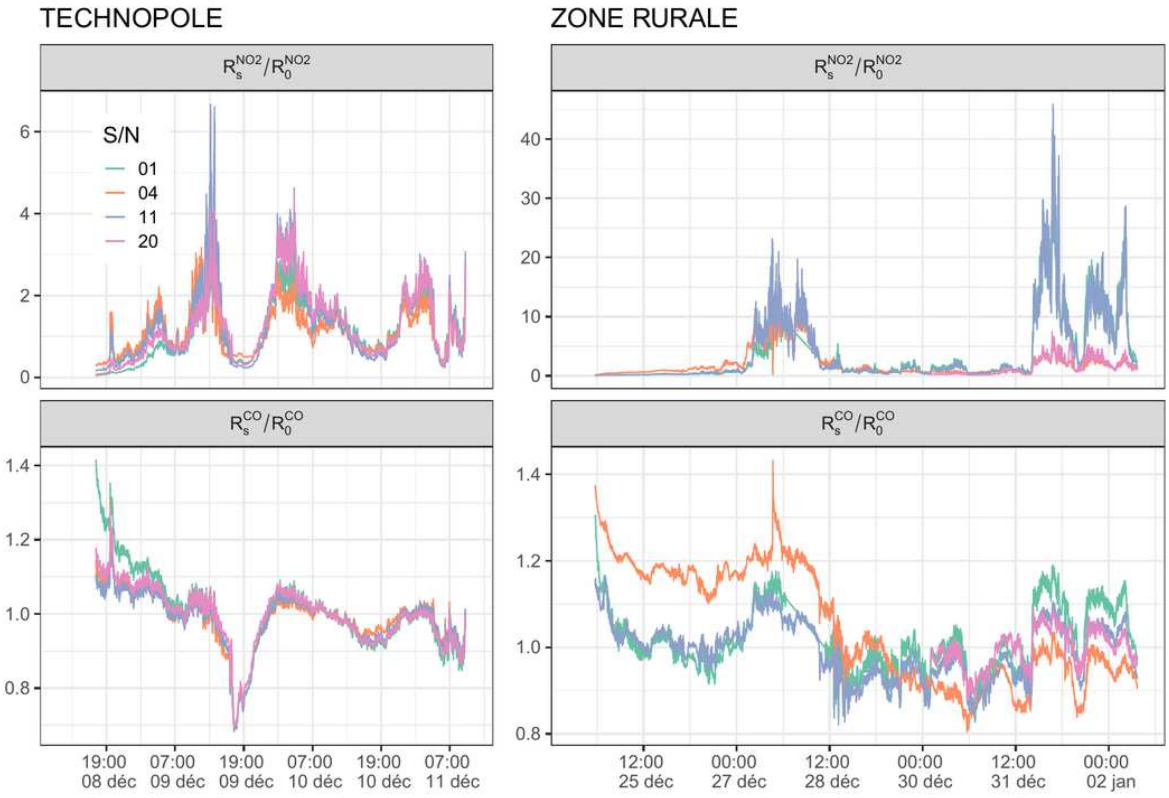


FIGURE 3.21 – Variation normalisée des réponses du capteur MICS-4514 sur les 4 systèmes intercomparés durant les 2 expériences.

La fréquence d'acquisition de 2 Hz est ramenée à une mesure par minute en prenant la valeur médiane. Nous remarquons que pour l'expérience « Technopole », les mesures sont extrêmement bien corrélées entre les capteurs. La moyenne des corrélations deux à deux est de 0,95 pour le CO et 0,89 pour le NO_2. Sans le capteur 04, la moyenne des corrélations pour le NO_2 est de 0,95. Pour l'expérience en zone rurale, nous observons immédiatement que le capteur 04 a une réponse différente. En effet, les valeurs enregistrées avant le 29 décembre sont au-dessus des valeurs des 2 autres capteurs et en dessous après cette date. Sa corrélation moyenne avec les autres capteurs est de 0,84 contre une corrélation moyenne de 0,90 pour les autres. Nous effectuons d'autres tests avec des pas d'échantillonnage plus grands, de la demie heure à la demie journée et en redressant la dérive de la courbe, mais

la corrélation aux autres capteurs n'a pas été améliorée. Nous décidons alors de retirer ce capteur de l'étude du CO, jugé non exploitable en l'état. Néanmoins, pour le NO_2, il semble fonctionner correctement. La moyenne des corrélations est de 0,91, et 0,94 en retirant le capteur 11 dont la sensibilité est plus forte que pour les autres capteurs.

Lorsque nous comparons les distributions des mesures normalisées au travers des diagrammes en boîte, présentées Figure 3.22, nous observons que lors de la première expérience, les médianes et écart-types sont semblables pour tous les capteurs, pour le CO et le NO_2. Lorsque nous comparons les deux expériences, nous observons que les médianes des distributions ont varié, et cela plus significativement pour le NO_2 que pour le CO. Pour le système 01, la moyenne a été divisée par deux entre les deux expériences pour le NO_2. Néanmoins, l'ordre de grandeur des variances reste semblable. Ceux dont la variance a le plus varié sont le système 04 pour CO, mais jugé inexploitable, et le système 20 pour le NO_2, mais qui n'a pas fonctionné tout le long de la seconde expérience.

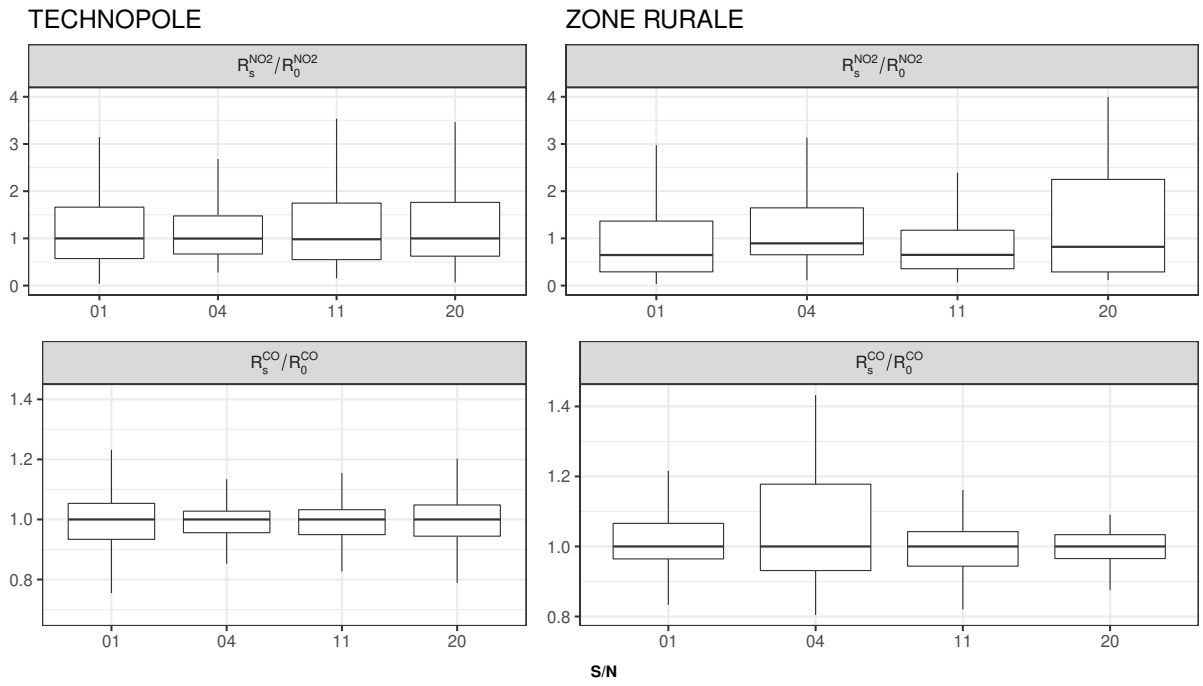


FIGURE 3.22 – Boîtes à moustaches décrivant les distributions de mesures en NO_2 et CO sur les 4 systèmes intercomparés durant les 2 expériences.

Nous montrons une cohérence entre les différents capteurs au travers de la corrélation. Un échantillonnage toutes les minutes permet de filtrer les très hautes fréquences et d'obtenir une forte corrélation entre les mesures des systèmes. Cela permet également de mettre en exergue les systèmes dont le comportement a changé relativement aux autres.

L'objectif initial était d'étalonner les capteurs par rapport à un capteur de référence, préalablement étalonné en laboratoire. Néanmoins, nous voyons après les expériences en laboratoire que cet étalonnage ne peut être fait.

Puisque cela n'est pas possible, nous nous inspirerons de deux méthodes pour étalonner nos capteurs de façon collaborative : l'étalonnage à l'aide d'une référence et l'étalonnage a posteriori basé sur les moments (Wang et *al.*, 2008). Cette seconde méthode considère

que les capteurs se déplacent dans la même région et donc que les statistiques du signal mesuré au fil du temps sont presque les mêmes.

Dans notre cas, puisque les capteurs sont au même endroit, ils observent le même signal. Par analogie avec la méthode des moments, nous considérons que les statistiques (moyenne, variance) observées par nos capteurs sont les mêmes que celles des stations de référence déployées par ATMO Occitanie dans la région toulousaine. Pour cela, nous étudions les statistiques des stations en fonction du type de lieu (zone urbaine/technopole, zone rurale). Ces résultats sont présentés au chapitre suivant, section 4.4.1.

Pour le NO_2, en zone urbaine, les concentrations sont centrées autour de $50\,\mu g/m^3$ et varient entre 20 et $80\,\mu g/m^3$. En zone rurale, les concentrations sont centrées autour de $20\,\mu g/m^3$ et varient entre 10 et $30\,\mu g/m^3$.

Pour le CO, nous ne disposons pas d'autant d'informations de la part d'ATMO. En zone de trafic, nous savons que la concentration est en moyenne de $370\,\mu g/m^3$ et varie entre 150 et $600\,\mu g/m^3$. En zones urbaine et rurale nous pensons que la concentration est moindre mais nous n'avons pas plus d'information. En effet, puisque la concentration est inférieure aux seuils recommandés, le monoxyde de carbone est moins surveillé.

Pour représenter le signal observé par nos capteurs, nous prenons la série temporelle moyenne des réponses normalisées des capteurs retenus. Puisque nous supposons que les variations sont globalement constantes lors des expériences, nous souhaitons obtenir un signal décorrélé, de variance globalement constante au cours du temps. Néanmoins, nous relevons des fluctuations importantes au cours des journées qui coïncident avec celles de la température et de l'humidité. La série temporelle moyenne en NO_2 est corrélée négativement à la température et à la pression (respectivement $-0{,}25$ et $-0{,}18$) et la série temporelle moyenne en CO est corrélée négativement à la température et positivement à l'humidité (respectivement $-0{,}54$ et $0{,}34$). Pour estimer correctement la médiane et l'écart-type en concentration du polluant, il nous faut alors décorréler les mesures des paramètres météorologiques, autrement dit, déterminer la ligne de base en fonction des paramètres météorologiques. Pour cela, nous testons plusieurs méthodes de prédiction du signal en fonction des paramètres météorologiques pour les deux expériences confondues, puis nous étudions le signal décorrélé au travers du résidu de modèle.

Nous testons la relation linéaire entre le gaz (NO_2 ou CO) et les conditions météorologiques, mais elle ne donne pas de résultats probants. De plus, nous décomposons les signaux des paramètres météorologiques en tendance de fond, en prenant la médiane horaire, et en tendance à court terme, en prenant le résidu entre la mesure et la tendance de fond. La relation linéaire n'a pas donné de meilleurs résultats. Ensuite, nous testons un GAM (cf. section 2.2.2.2) avec des splines d'ordre 10 pour les paramètres météorologiques ainsi que les décompositions en tendances. Les résultats ont été visuellement plus encourageants pour les décompositions en tendances. En augmentant l'ordre des splines à 60, les résultats s'améliorent grandement (modèle intitulé « GAM1 »). Enfin, nous ajoutons deux nouveaux paramètres au modèle GAM1, la tendance horaire et la tendance à court terme de l'autre gaz (modèle intitulé « GAM2 »). Nous notons que le signal se recentre encore davantage.

La Figure 3.23 présente les résidus des deux derniers modèles (GAM1 et GAM2) pour le CO et le NO_2 durant les deux expériences.

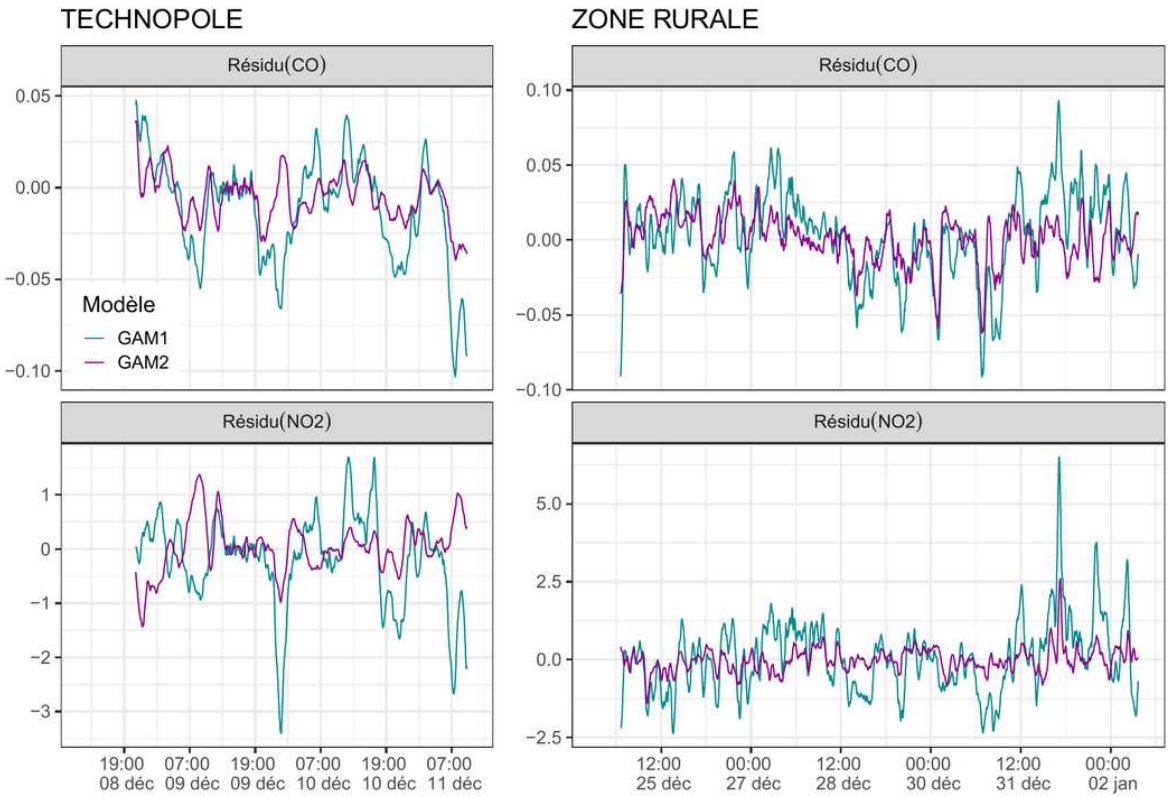


FIGURE 3.23 – Variation du résidu entre la série temporelle normalisée moyenne des capteurs MiCS-4514 et la prédiction des modèles GAM1 et GAM2 durant les deux expériences.

Ainsi nous retenons ce dernier modèle (GAM2). Pour celui-ci, pendant la première expérience, la médiane pour le CO est de $-0{,}0039$ et le premier décile du signal commence à $-0{,}0290$ et le dernier à $0{,}0153$; la médiane pour le NO_2 est de $0{,}0175$ et le premier décile du signal commence à $-0{,}7440$ et le dernier à $0{,}9632$.

Pendant la seconde expérience, la médiane pour le CO est de $0{,}0022$ et le premier décile du signal commence à $-0{,}0267$ et le dernier à $0{,}0253$; la médiane pour le NO_2 est de $-0{,}0178$ et le premier décile du signal commence à $-0{,}6098$ et le dernier à $0{,}5022$.

Pour le NO_2, les valeurs sont plus élevées lors de la première expérience, ce qui est attendu car le technopole est a priori plus pollué que la zone rurale. Pour le CO, nous remarquons l'inverse, ce qui est également attendu car le capteur de CO réagit négativement au contact du gaz. De plus, les ordres de grandeur des variations du signal entre la fiche technique et le signal observé par ATMO Occitanie correspondent approximativement, mais d'autres expériences sont nécessaires pour valider cette approche.

Enfin, nous faisons correspondre les concentrations journalières moyennes en NO_2 relevées par ATMO Occitanie (min, médiane, max) à nos statistiques (premier décile, médiane, dernier décile) (cf. Figure 3.24).

Ces correspondances sont liées de façon linéaire avec un coefficient de détermination de $0{,}99$ pour les deux expériences, mais seulement avec trois points. Il semblerait intéressant de reproduire cette méthode pour de nouvelles expériences et de nouvelles statistiques afin d'établir une relation fiable entre le signal décorrélé et la concentration réelle au regard

de la gamme de valeurs observées.

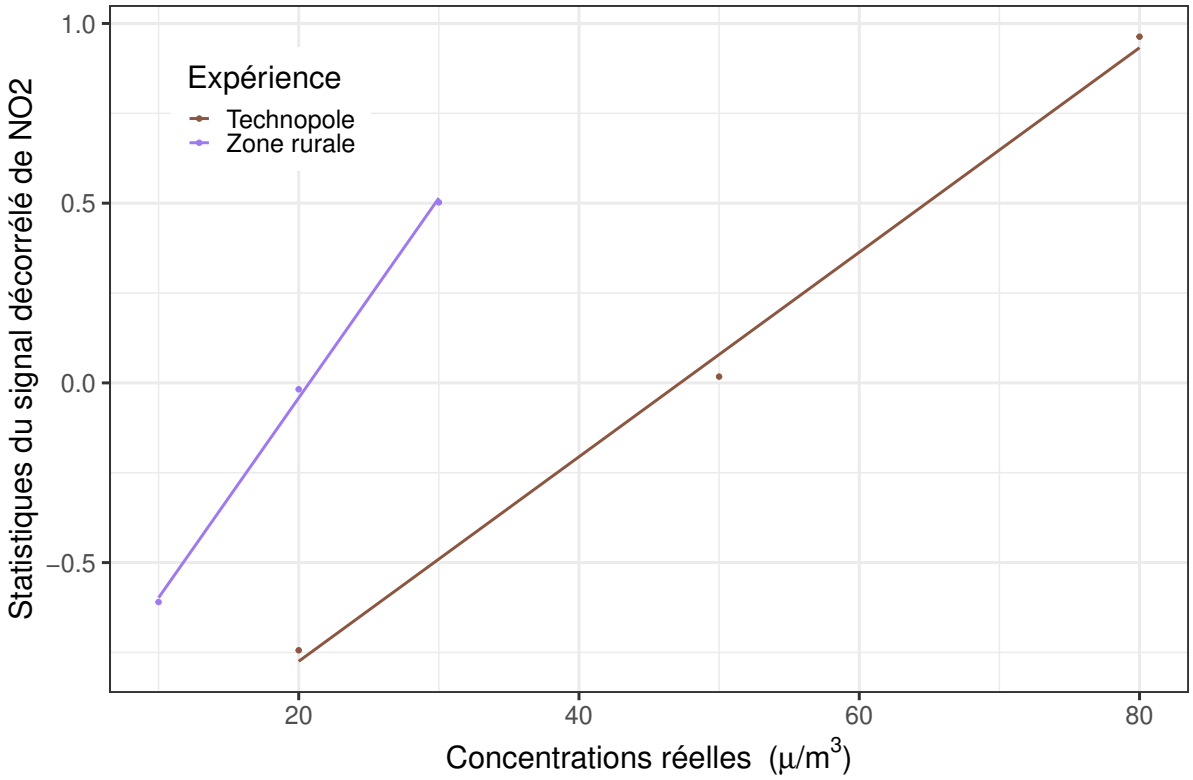


FIGURE 3.24 – Concentrations journalières moyennes en NO_2 relevées par ATMO (min, médiane, max) en fonction de nos statistiques (premier décile, médiane, dernier décile).

3.6 Conclusion

Ce chapitre a été consacré au développement d'un système embarqué de mesure de composants gazeux dédié au vélo. Il a été fait le choix de développer un système ayant un fonctionnement quasi autonome afin de le rendre transparent vis à vis de l'utilisateur. Ce choix a imposé de fortes contraintes de développement afin d'obtenir un système fonctionnel de bout en bout.

Le circuit imprimé final est contrôlé par un microcontrôleur MSP430. Il permet de sauvegarder et communiquer automatiquement les mesures. Le système est également constitué d'un capteur de géolocalisation GPS, d'un capteur de paramètres météorologiques et d'un accéléromètre. Il est protégé des chocs et des intempéries par un boîtier et peu visible sur un vélo grâce à sa petite taille. Les contraintes énergétiques montrent que l'alimentation par la dynamo est peu exploitable car, le temps que l'énergie soit stockée, le trajet est déjà bien entamé. Cela pose un sérieux problème pour l'initialisation du récepteur GPS (presque 20 minutes pour les démarrages à froid par ciel nuageux) et rend inutilisable le système pour des trajets courts. Nous avons donc dû utiliser une batterie supplémentaire chargée par l'utilisateur.

La conception du système embarqué pour la pollution de l'air en zone urbaine s'est articulée autour d'un micro-capteur de gaz. Nous avons choisi le micro-capteur MiCS-4514 à Métal-Oxyde Semi-conducteur (MOx). Le MiCS-4514 cible la pollution automobile en mesurant les NO_2 principalement émis par les véhicules diesel et le CO principalement émis par les véhicules essence. Ce type de capteur a une bonne sensibilité aux gaz ciblés. Il consomme suffisamment peu d'énergie pour qu'un cycliste la produise et est étudié dans notre laboratoire par une autre équipe. Les MOx exploitent les échanges d'électrons d'un oxyde métallique au contact d'un gaz cible et la variation de la résistivité de l'oxyde métallique pour mesurer la variation de concentration du gaz cible. Nous avons étudié le comportement d'un système en laboratoire en ce qui concerne la variation de température et la variation du CO. Le NO_2 n'a pas été testé en raison des faibles valeurs atmosphériques que nous n'avons pas pu reproduire en laboratoire. Le BME280 a un comportement fiable et renvoie une mesure de la température très corrélée à la référence. La réponse du MiCS-4514 dérive dans le temps et l'étalonnage de la fiche technique ne semble pas exploitable, même grossièrement. Ce capteur est sensible aux variations des conditions environnementales (température, pression et humidité relative). Nous avons fait correspondre les valeurs observées par nos capteurs à celles observées par les stations de référence d'ATMO Occitanie. Cette technique suggère que la réponse du capteur est localement linéaire, mais que le coefficient linéaire dépend de la plage des valeurs mesurées. Elle semble potentiellement applicable à l'analyse des capteurs sur vélo en catégorisant les régions de l'espace en fonction du *niveau a priori* de pollution.

La plateforme est également générique et peut accueillir d'autres capteurs (bruit, lumière...). Il était initialement prévu d'inclure un capteur de particules (PM_{10} ou $PM_{2.5}$), ce qui n'a pu être réalisé. Nous avons néanmoins pu tester l'ajout de nouveaux capteurs dédiés à la mesure de la pollution (O3 et CO) en connectant un module complémentaire via le port UART au travers d'un projet étudiant.

Le chapitre suivant est dédié au déploiement de cette plateforme à l'échelle de la métropole de Toulouse.

Bibliographie

Angelis, L. D. et Minnaja, N. . Sensitivity and selectivity of a thin-film tin oxide gas sensor. Sensors and Actuators B : Chemical, 1991. DOI : 10.1016/0925-4005(91)80006-6.

Castell, N. , Dauge, F. R. , Schneider, P. , Vogt, M. , Lerner, U. , Fishbain, B. , Broday, D. et Bartonova, A. . Can commercial low-cost sensor platforms contribute to air quality monitoring and exposure estimates ? Environment International, 2017. DOI : 10.1016/j.envint.2016.12.007.

Dufour, N. . Conception et réalisation d'un multicapteur de gaz intégré à base de plateformes chauffantes sur silicium et de couches sensibles à oxydes métalliques pour le côntrole de la qualité de l'air habitacle. Doctorat, Université Paul Sabatier - Toulouse III, 2013. https://www.semanticscholar.

org/paper/Conception-et-réalisation-d'un-multicapteur-de-gaz-Dufour/f6a0e861ca55c752ebc854474de96156ad736783.

Gamma, E. , Helm, R. , Johnson, R. et Vlissides, J. . Design Patterns : Abstraction and Reuse of Object-Oriented Design. ECOOP'93 - Object-Oriented Programming, 1993. DOI : 10.1007/978-3-642-48354-7_15.

Gong, H. , Chen, C. , Bialostozky, E. et Lawson, C. T. . A GPS/GIS method for travel mode detection in New York City. Computers, Environment and Urban Systems, 2012. DOI : 10.1016/j.compenvurbsys.2011.05.003.

Jerrett, M. , Donaire-Gonzalez, D. , Popoola, O. , Jones, R. , Cohen, R. C. , Almanza, E. , Nazelle, A. d. , Mead, I. , Carrasco-Turigas, G. , Cole-Hunter, T. , Triguero-Mas, M. , Seto, E. et Nieuwenhuijsen, M. . Validating novel air pollution sensors to improve exposure estimates for epidemiological analyses and citizen science. Environmental Research, 2017. DOI : 10.1016/j.envres.2017.04.023.

Joo, S. , Oh, C. , Jeong, E. et Lee, G. . Categorizing bicycling environments using GPS-based public bicycle speed data. Transportation Research Part C : Emerging Technologies, 2015. DOI : 10.1016/j.trc.2015.04.012.

Korotcenkov, G. . Metal oxides for solid-state gas sensors : What determines our choice ? Materials Science and Engineering : B, 2007. DOI : 10.1016/j.mseb.2007.01.044.

Korotcenkov, G. et Cho, B. K. . Metal oxide composites in conductometric gas sensors : Achievements and challenges. Sensors and Actuators B : Chemical, 2017. DOI : 10.1016/j.snb.2016.12.117.

Lewis, A. , Lee, J. , Edwards, P. , Shaw, M. , Evans, M. , Moller, S. , Smith, K. , Ellis, M. , Gillott, S. , White, A. et Buckley, J. . Evaluating the performance of low cost chemical sensors for air pollution research. Faraday Discuss, 2015. DOI : 10.1039/C5FD00201J.

Liu, H. , Zhang, L. , Li, K. H. H. et Tan, O. K. . Microhotplates for Metal Oxide Semiconductor Gas Sensor Applications—Towards the CMOS-MEMS Monolithic Approach. Micromachines, 2018. DOI : 10.3390/mi9110557.

Mead, M. I. , Popoola, O. A. M. , Stewart, G. B. , Landshoff, P. , Calleja, M. , Hayes, M. , Baldovi, J. , McLeod, M. , Hodgson, T. , Dicks, J. , Lewis, A. , Cohen, J. , Baron, R. , Saffell, J. et Jones, R. . The use of electrochemical sensors for monitoring urban air quality in low-cost, high-density networks. Atmospheric Environment, 2013. DOI : 10.1016/j.atmosenv.2012.11.060.

Menini, P. . Du capteur de gaz à oxydes métalliques vers les nez électroniques sans fil. Habilitation à diriger des recherches, Université Paul Sabatier - Toulouse III, 2011. https://hal.archives-ouvertes.fr/tel-00697471.

Schuessler, N. et Axhausen, K. W. . Processing Raw Data from Global Positioning Systems without Additional Information. Transportation Research Record, 2009. DOI : 10.3141/2105-04.

Spinelle, L. , Gerboles, M. , Villani, M. G. , Aleixandre, M. et Bonavitacola, F. . Field calibration of a cluster of low-cost commercially available sensors for air quality monitoring. Part B : NO, CO and CO2. Sensors and Actuators B : Chemical, 2017. DOI : 10.1016/j.snb.2016.07.036.

Unnikrishnan, J. et Vetterli, M. . Sampling and Reconstruction of Spatial Fields Using Mobile Sensors. IEEE Transactions on Signal Processing, 2013. DOI : 10.1109/TSP.2013.2247599.

Wang, C. , Ramanathan, P. et Saluja, K. K. . Moments based blind calibration in mobile sensor networks. Communications, 2008. ICC'08. IEEE International Conference on, 2008. DOI : 10.1109/ICC.2008.176.

Yi, W. Y. , Lo, K. M. , Mak, T. , Leung, K. S. , Leung, Y. et Meng, M. L. . A Survey of Wireless Sensor Network Based Air Pollution Monitoring Systems. Sensors, 2015. DOI : 10.3390/s151229859.

Chapitre 4

Application à la métropole de Toulouse

Ce n'est qu'avec les yeux des autres que l'on peut bien voir ses défauts.
– Proverbe Chinois

Sommaire

4.1 Introduction

Dans le chapitre 2, nous avons comparé plusieurs projets de réseaux de capteurs mobiles pour la pollution de l'air en zone urbaine au regard de six indicateurs : la couverture spatiale, la résolution temporelle, la qualité des données, les besoins en maintenance, le temps d'autonomie du système et le coût. Dans l'optique de maximiser la couverture spatiale à un coût réduit, nous avons choisi d'étudier un réseau de micro-capteurs low-cost sur vélo. Par la suite, nous avons caractérisé théoriquement la taille du réseau nécessaire pour couvrir une ville comme Marseille, située aux alentours d'une quarantaine de vélos. Dans le chapitre 3, nous avons présenté la conception du système à embarquer sur les vélos, du prototypage au produit final, puis nous avons évalué les performances des capteurs embarqués. Il a été pensé pour être transporté par des cyclistes volontaires : il est robuste et facile à utiliser.

Ce chapitre expose le déploiement de notre réseau de capteurs sur vélos dans la ville de Toulouse. D'abord, nous identifions six stratégies générales pour rassembler des participants et collecter des mesures et en choisissons deux d'entre elles. La première fut de faire appel à des cyclistes volontaires, touristes ou amateurs, par le biais d'une association de location de vélos. La seconde fut de solliciter les vélo-taffeurs [1] de notre laboratoire. Cette dernière nous a permis d'obtenir un jeu de données exploitable. Ce jeu de données est présenté par la suite et analysé en suivant la démarche résumée par la Figure 4.1.

Dans un premier temps, nous traitons les données enregistrées par nos systèmes embarqués pour reformer les trajets à vélo des utilisateurs. En effet, ces données géolocalisées sont ponctuelles et ne correspondent pas à un trajet à vélo à proprement parler ; le système a pu être laissé allumé entre deux trajets, être redémarré au cours d'un trajet par le watchdog, être transporté en voiture...

Dans un deuxième temps, nous étudions le profil des cyclistes au travers de leur utilisation du vélo et de la dynamique de leurs déplacements (durée moyenne, distance moyenne, vitesse moyenne). Ensuite, sur la base de ces profils réels, nous discutons du réalisme du modèle de cycliste construit au chapitre 2.

Dans un troisième temps, nous analysons quelques cas particuliers (vélos côte à côte, quartier, *Rendez-Vous*) en revenant aux réponses des capteurs MiCS-4514. Par la suite, nous examinons la présence de *Rendez-Vous* avec les stations fixes de référence d'ATMO Occitanie et entre vélos. Les *Rendez-Vous* nous permettent d'esquisser un étalonnage entre ces réponses et les concentrations réelles à l'aide des stations de référence et une méthode de détection des capteurs défaillants. Néanmoins, nous n'avons pas suffisamment abouti cette approche pour l'exploiter. Enfin, nous étudions les trajets des vélos dans leur ensemble, nous spatialisons les réponses des capteurs à l'aide des méthodes de spatialisation utilisées au chapitre 2 afin d'estimer les niveaux de pollution dans la ville de Toulouse et nous discutons ces résultats. Ces niveaux de pollution ne correspondent pas à des concentrations réelles mais permettent de déterminer les points chauds en NO_2 et CO.

1. Personne qui se rend au travail à vélo.

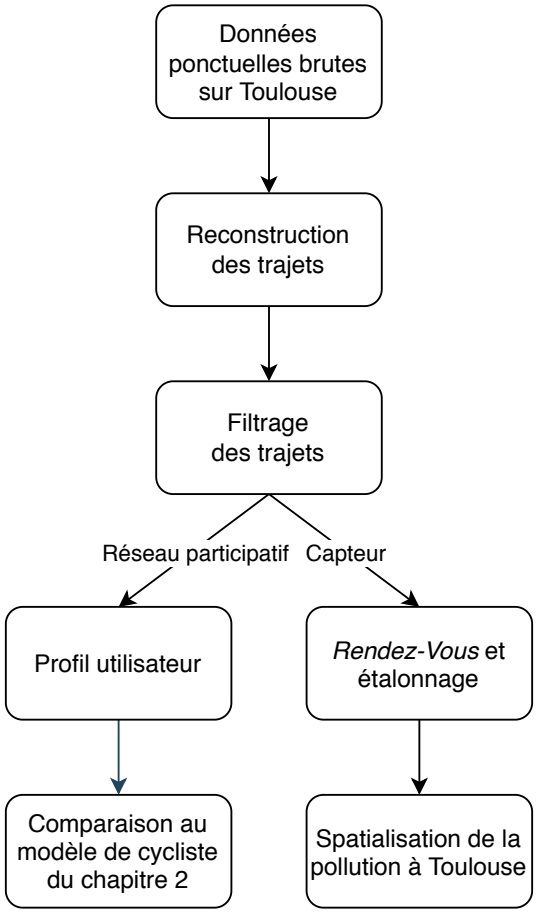


FIGURE 4.1 – Démarche du chapitre 4.

4.2 Stratégies de mesure

Nous distinguons six stratégies pour collecter des mesures mobiles. D'une part, il est possible de rassembler des participants conscients de l'expérience :

— le chercheur à l'origine de l'étude,
— des expérimentateurs avertis et suivant un protocole,
— ou des volontaires plus ou moins engagés (du citoyen bénévole à l'utilisateur d'un service).

D'autre part, il est possible d'orchestrer la capture d'informations :

— par des robots (LaMarca et *al.*, 2002) – définis ici comme des machines mobiles de fonctionnement connu – qui peuvent être détournés pour embarquer des capteurs (par exemple, le réseau de transport public) ou programmés spécifiquement (par exemple, une flotte de drones ou de vélos sans pilote (Cook, 2004; He et *al.*, 2015)),
— grâce à la ludification (ou gamification en anglais) – qui est l'utilisation de mécanismes de jeu pour faire adopter un produit, un service ou un comportement – (par exemple, Pokemon Go qui incite les joueurs à se déplacer et donc à effectuer une activité physique (Althoff et *al.*, 2016) pour capter les déplacements des joueurs et pour estimer la réaction d'un joueur à une récompense),
— ou par la surveillance, notamment via les smartphones (de plus en plus pratiquée

par les constructeurs et envisagée comme solution sécuritaire).

Ensuite, nous différencions les stratégies en fonction de leur biais de sélection dans la zone couverte. Plus le comportement est dicté au regard d'un a priori sur les données, plus celles-ci souffriront de ce biais.

Dans le cas du chercheur ou des robots, le protocole à suivre est défini explicitement et respecté. Par exemple, une flotte de drones peut être dédiée à la surveillance de la qualité de l'air et déterminer son déplacement en fonction de son apriori de la concentration en un polluant donné (Belkhiri et *al.*, 2018).

Dans le cas des expérimentateurs ou des joueurs (ludification), des instructions sont implicites et guident les participants. Ce cas semle particulièrement intéressant mais encore peu étudié.

Dans le cas des volontaires ou de la surveillance, les individus sont libres de leur conduite.

En outre, nous identifions deux axes d'analyse. D'abord, le fait que la mobilité soit imposée (robots) ou non (volontaires) engendre des contraintes différentes. Dans le cas d'une mobilité fortement contrainte, des études complémentaires sont nécessaires pour assurer la représentativité des mesures au regard des motifs de déplacement. Dans le cas d'une mobilité libre, il faut adopter une démarche pédagogique envers les volontaires et leur présenter le système et les résultats pour les stimuler. De plus, il faut étudier la représentativité de l'échantillon des utilisateurs. Néanmoins, les participants motivés peuvent contribuer activement au projet et réduire les coûts (du simple fait de posséder un smartphone par exemple) (Kosmidis et *al.*, 2018).

Le second axe d'analyse concerne les conditions de mesure et l'impact sur les données. Dans le cas du chercheur, les conditions sont contrôlées et l'incertitude évaluée, ce qui assure la qualité des données. Toutefois, le temps d'expérimentation est coûteux pour le chercheur. Dans le cas d'un système de surveillance, les conditions de mesure ne sont pas précisées aux utilisateurs donc ne sont pas contrôlées mais la quantité de données collectées est bien plus importante. Cependant, le système requiert un haut niveau d'automatisation (autonomie, maintenance, collecte des données) et de respecter la vie privée des individus.

La Figure 4.2 reprend ces six stratégies et schématise ce propos. Les deux stratégies colorées sont celles que nous avons adoptées. La première, en rouge, fut d'équiper une partie des vélos de location d'une association et la seconde, en vert, de solliciter les vélotaffeurs de notre laboratoire de recherche. Seule la seconde stratégie nous a permis de collecter un jeu de données exploitable.

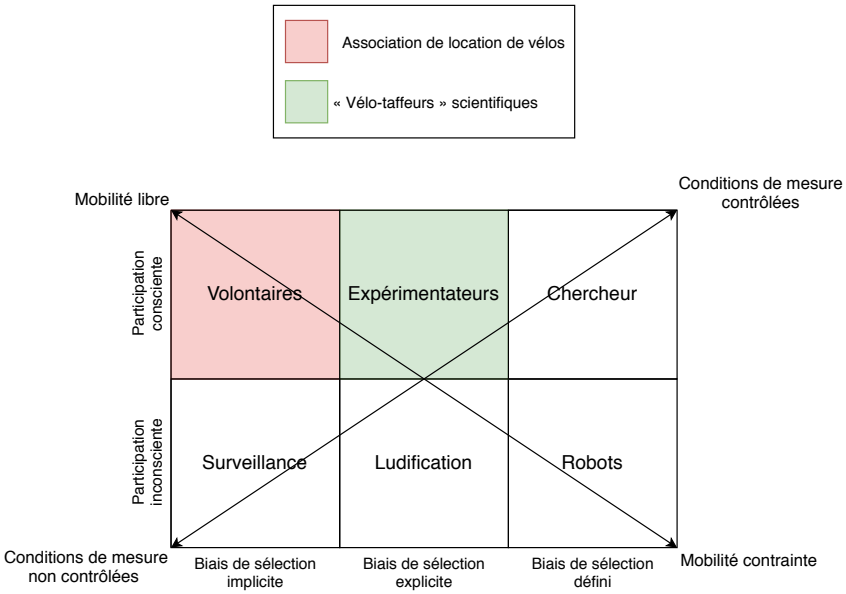


FIGURE 4.2 – Stratégie de déploiement des systèmes en fonction des contraintes de mobilité et des conditions d'observation.

4.2.1 Association de location de vélos

La première stratégie mise en œuvre fut de faire appel à des cyclistes volontaires via une association toulousaine de location de vélos, « la Maison du Vélo ». Elle dispose d'un large panel d'abonnés composé d'amateurs de vélos et de touristes. En effet, d'une part, elle propose des abonnements à la semaine, au mois ou à l'année et un service gratuit d'aide à la réparation. D'autre part, elle est située en plein cœur de Toulouse, en face de la gare, associée à un restaurant. Elle propose également des abonnements à la journée et conseille sur les lieux à visiter. De plus, sa position centrale permet d'espérer que des vélos croisent les trois stations de référence de la ville. Aucune statistique d'utilisation n'a pu être fournie par la Maison du Vélo.

Au-delà du panel d'utilisateurs large et au profil diversifié, l'avantage de ce partenariat est de déléguer la gestion des utilisateurs à un tiers : les questions diverses sur le système embarqué, le recueil du consentement et le prêt du système embarqué prennent un temps non négligeable. De plus, le retour des vélos au lieu de location permet de n'utiliser qu'une seule passerelle pour collecter les données des systèmes embarqués. Enfin, l'association nous a procuré une flotte de vélos identiques – ce qui réduit les incertitudes quant à l'interprétation des données fournies par les accéléromètres – et équipés de dynamos.

Ce partenariat nécessite de fournir un système robuste et fonctionnel de bout en bout en minimisant l'intervention d'un tiers. La mesure par le système doit être entièrement indépendante, sans initialisation préalable par l'utilisateur. Au mieux l'énergie doit être entièrement pourvue par la dynamo et les données téléversées sans fil, au pire un membre de l'association peut brancher un câble. Pour favoriser l'adoption du système, nous avons développé une page web qui présente en temps réel les données collectées sous la forme d'un « tableau de bord » (dashboard). Les solutions logicielles développées ont été présentées section 3.4.1. En outre, la fixation des systèmes embarqués aux vélos et la maintenance

des systèmes embarqués endommagés, souvent au niveau du port micro-USB, engendrent des allers-retours entre l'association et le laboratoire.

Cette forte contrainte a nécessité un long temps de développement avant de fournir un système utilisable. Nous avons équipé 16 vélos mais les trajets collectés ont essentiellement permis de déboguer des cas jamais rencontrés lors de nos tests. De plus, l'ignorance du moment où les vélos équipés sont utilisés, couplée au phénomène d'« effet de canyon urbain » (identifié le long du canal du Midi – piste cyclable toulousaine emblématique entourée de deux allées de platanes –, cf. section 3.4.4) a retardé cette phase de débogage. Le principal problème était de garantir un temps d'initialisation plus court que le temps du trajet. En effet, les locations à la journée – afin d'avoir un retour rapide – sont principalement faites par des touristes, qui effectuent de petits trajets (de l'ordre de la dizaine de minutes) à faible vitesse (environ 5–10 km/h). Or le temps de charge de la batterie du système embarqué est sujet à la vitesse du cycliste et le temps de synchronisation du récepteur GPS est incertain et dépendant de la visibilité du ciel (météorologie, présence d'obstacles...). En conséquence, nous avons changé la batterie tampon Li-Po par un supercondensateur (évoqué section 3.4.2) ce qui nous a permis de grandement améliorer le nombre de trajets collectés. Cependant, nous avons diagnostiqué tardivement ce cas de dysfonctionnement.

Toutefois, le peu de résultats a démotivé les membres de l'association. Au cours des locations, quatre vélos équipés ont été volés au rythme d'environ un par mois, mais aucun système n'a été abîmé sciemment ou volé. Aucune donnée n'a été exploitable au-delà de la visualisation de nuages de points (positions GPS de mauvaise qualité) ou de portions de trajets. De nombreuses incertitudes déclenchaient le système de watchdog (cf. section 3.3.3.2) et faisaient redémarrer le système. Pour pallier les problèmes d'énergie et de GPS, nous avons préféré par la suite nous concentrer sur une base d'utilisateurs qui ne nécessite pas une solution pleinement autonome.

4.2.2 « vélo-taffeurs » scientifiques

Après cette tentative avec l'association de location de vélos, nous avons décidé de nous tourner vers les membres de notre laboratoire qui viennent travailler quotidiennement à vélo, les « vélo-taffeurs ». En effet, nous avons déjà choisi d'étudier un modèle de mobilité similaire pour sa simplicité au chapitre 2 (trajet quotidien unique, polarisé sur quelques zones attractives). Cela nous a permis de caractériser le nombre de participants nécessaires, aux alentours de quelques dizaines. Bien que seulement douze systèmes embarqués aient été fonctionnels au lieu d'une quarantaine (cf. chapitre 2), l'objectif de cette étude est de mettre en regard les données synthétiques du chapitre 2 et celles collectées à Toulouse.

De plus, la proximité de la localisation de ces participants – dans notre laboratoire – a facilité la collecte. Nous nous sommes affranchis de l'utilisation de la passerelle et nous avons pu directement déverser les données dans une base de fichiers.

Selon l'Institut national de la statistique et des études économiques (INSEE), en Occitanie les vélo-taffeurs représentent 2,2 % des travailleurs et sont plutôt des hommes de tout âge, urbains et diplômés du supérieur, parcourant des distances inférieures à 5 km. Dans notre cas, ne disposant que d'un faible nombre de systèmes embarqués, les cyclistes

les plus éloignés du laboratoire et les plus motivés ont été privilégiés. Cela a permis de fédérer facilement un groupe de dix-huit « vélo-taffeurs » (3 femmes, 15 hommes ; 10 entre 25 et 35 ans, 8 entre 35 et 50 ans) urbains et ruraux et de collecter des données efficacement. Le temps de la collecte de données a duré sept mois, de septembre 2018 à mars 2019 avec douze systèmes embarqués ; et nous avons rassemblé des participants durant toute la période de l'expérience. Les données collectées sont présentées section 4.3.

En outre, le vélo-taffeur scientifique peut être considéré comme un « expérimentateur » selon les différentes stratégies exposées précédemment (participation consciente, biais de sélection explicite) pour deux raisons.

La première est que les scientifiques sont sensibilisés au respect d'un protocole simple pour assurer des données exploitables, ce qui permet de passer moins de temps à développer une solution de bout en bout au prix de quelques interactions de la part de l'utilisateur. En l'occurrence, pour pallier les problèmes d'initialisation évoqués à la section précédente, nous avons adjoint une batterie au système embarqué pour ne plus avoir besoin de la dynamo. Cela assure une alimentation constante au système. Le protocole qui en découle est simplement de charger la batterie du système et attendre l'acquisition de la position GPS indiquée par le clignotement de DELs puis fixer solidement le système au vélo. Nous avons opté pour un ruban auto-agrippant pour facilement attacher et détacher le système du vélo.

La seconde est le fait que tous les utilisateurs proviennent de la même entreprise (profil vélo-taffeurs) ce qui introduit un biais plus fort qu'avec l'association de location de vélos. Les utilisateurs ont plus de chances d'avoir les mêmes centres d'intérêt, de fréquenter les mêmes lieux et d'habiter dans les mêmes quartiers. Cela peut toutefois être un avantage pour l'étalonnage par *Rendez-Vous* (cf. section 1.3.2). De plus, cela améliore la résolution temporelle en ces lieux tout en préservant une bonne résolution spatiale.

La Table 4.1 récapitule les nouvelles contraintes et les biais attendus pour les deux bases d'utilisateurs étudiées, les locataires volontaires de l'association « la Maison du Vélo » (cf. section précédente) et les vélo-taffeurs scientifiques.

Table 4.1 – Contraintes et biais.

		Association de location de vélos	vélo-taffeurs scientifiques
Contraintes	ajoutées	robustesse fonctionnement de bout en bout, autonome pédagogie envers l'asso collecte de données distante, non surveillée maintenance, temps de dpct asso–labo	pédagogie envers l'utilisateur facilité d'installation du système attente de retour scientifique
	relâchées	pas de pédagogie envers l'utilisateur 1 seul lieu de collecte, 1 seule passerelle coût des vélos vélos identiques	temps de développement réduit proximité du lieu de collecte, pas de passerelle coût des vélos
Biais	motifs de déplacement	touristes, cyclistes amateurs	mêmes profil-utilisateur probables trajets réguliers : parents probables (trajet via école), etc
	couverture spatiale	centrée sur l'asso, au centre de Toulouse probabilité de croiser une station de réf. favorable aux pistes cyclables	centrée sur le labo, excentrée de Toulouse (périphérique) ville et campagne probabilité de *Rendez-Vous*
	résolution temporelle	toutes heures	matins et soirs (heures des trajets)

4.3 Jeu de données collecté

4.3.1 Formatage des données

Les informations collectées par les vélo-taffeurs scientifiques du LAAS-CNRS sont un ensemble de fichiers de journalisation des systèmes embarqués dont la taille totale fait environ 1,5 Go. Pour rappel (cf. section 3.3.4), un fichier de journalisation contient une série d'inscriptions correspondant à des évènements. Un extrait de fichier est présenté en annexe 7. Le format d'une inscription est le suivant :

T _____ ID _____ ...

avec T la date d'exécution de la tâche, ID l'identifiant de la tâche et ... l'ensemble des valeurs de retour. La procédure de traitement des fichiers de journalisation est présentée à la Figure 4.3.

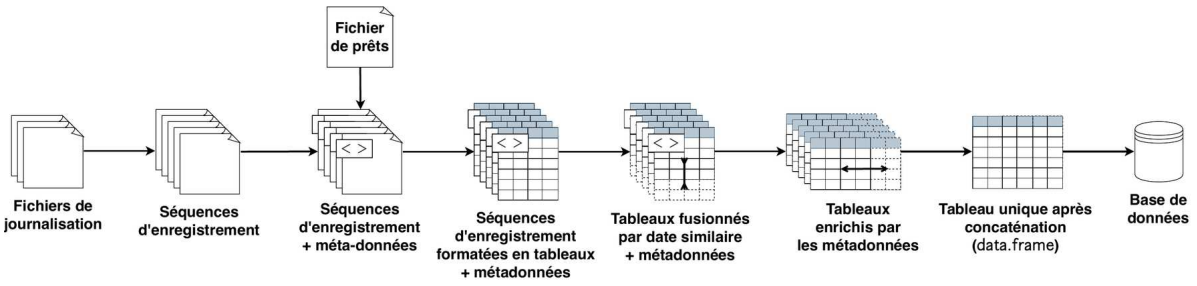


FIGURE 4.3 – Chaîne de traitement des fichiers de journalisation des systèmes embarqués.

Pour former notre jeu de données, nous traitons chaque fichier individuellement et le découpons à chaque redémarrage – indiqué par une inscription. Cette inscription possède comme première valeur de retour l'identifiant du système embarqué et comme seconde valeur de retour la version du logiciel. Cela permet de changer une carte micro-SD de système embarqué. En effet, dans la configuration adoptée, les cartes micro-SD ont une faible durée de vie et doivent être renouvelées fréquemment du fait de leur écriture systématique sur les secteurs en tête.

A chaque nouveau fichier – correspondant à un redémarrage, nous associons les métadonnées suivantes :

— le nom du fichier découpé en tant qu'un identifiant unique,
— l'identifiant du système embarqué,
— la version du logiciel embarqué,
— le nom de l'utilisateur du système embarqué.

Pour retrouver l'utilisateur d'un système embarqué, nous utilisons un fichier type tableur rempli manuellement où nous avons noté les prêts des systèmes embarqués.

Puis, nous formatons chaque fichier en tableau où chaque ligne correspond à une mesure et chaque colonne aux informations journalisées. Puisqu'une mesure correspond à un type de capteur (MOx, météo., GPS), les lignes ont des valeurs manquantes. Les colonnes du tableau sont les suivantes :

— la date (AAAA/MM/JJ HH:MM:SS),
— les coordonnées géographiques (longitude, latitude, altitude),

— le nombre de satellites synchronisés lors du calcul des coordonnées géographiques, ce qui indique la précision des coordonnées géographiques,
— la température,
— la pression,
— l'humidité relative,
— la résistance de l'élément sensible du capteur de NO_x,
— la résistance de l'élément sensible du capteur de CO.

Ensuite, nous fusionnons les lignes correspondant aux mêmes dates. Si les mesures proviennent de deux types de capteurs différents, la fusion consiste à garder les valeurs présentes dans chacune des lignes ; sinon, nous prenons la moyenne des valeurs.

Par la suite, nous ajoutons les métadonnées associées à chaque tableau sous la forme de nouvelles colonnes (aux valeurs toutes identiques). Enfin, nous fusionnons tous ces tableaux pour n'en former qu'un et le stocker en base de données et sous forme de data.frame en R.

La Figure 4.4 est une carte affichant tous les points GPS de la base de données. La couleur dépend du système embarqué qui les a collectés. Par la suite, les visualisations seront centrées sur la ville de Toulouse et ses environs pour une meilleure lisibilité.

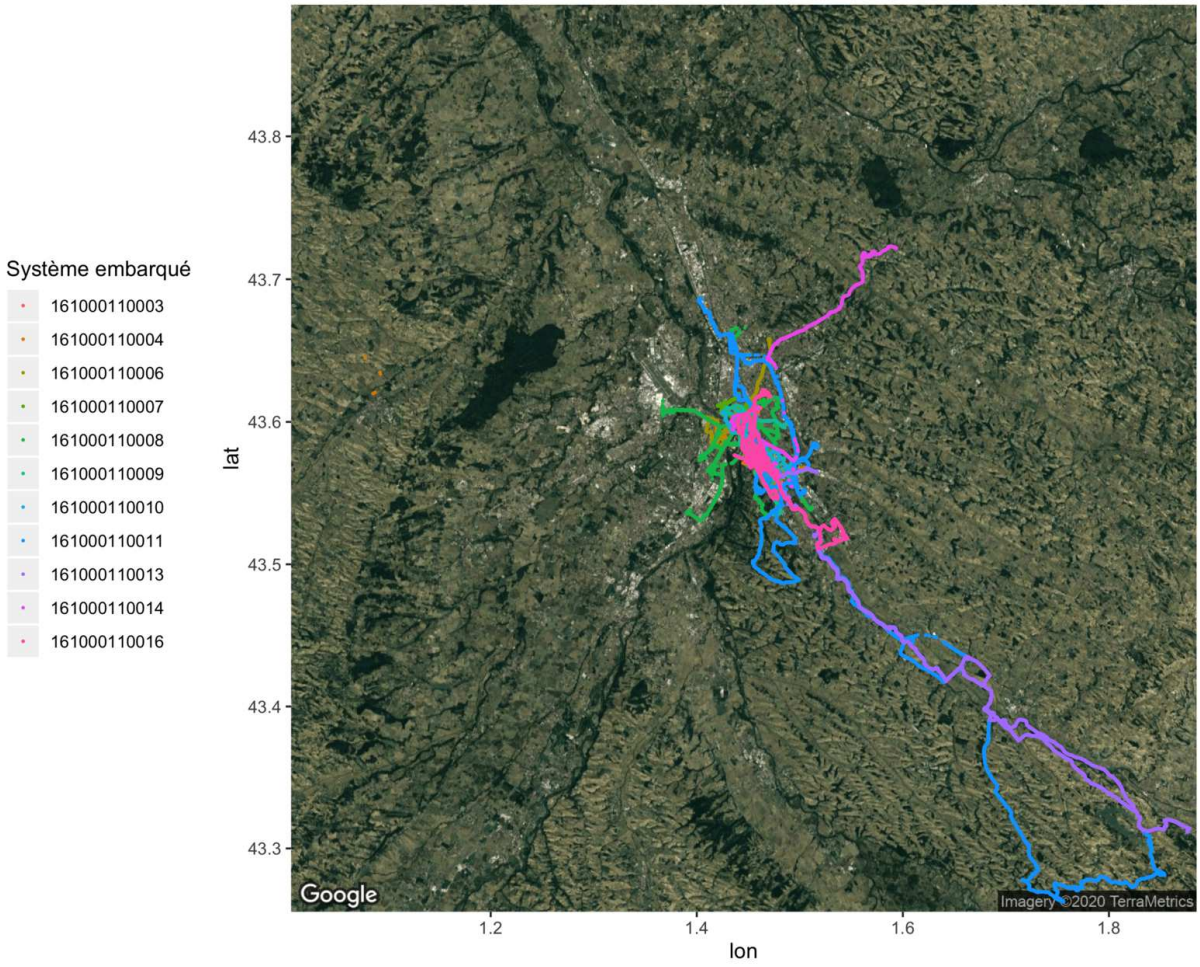


FIGURE 4.4 – Positions GPS collectées par les 11 systèmes embarqués.

4.3.2 Filtrage et reconstruction des trajets

Nous observons sur la Figure 4.4 que la zone couverte est relativement dense sur la région toulousaine mais que deux systèmes embarqués ont longé l'autoroute A61 jusqu'à Castelnaudary. Pour étudier ce jeu de données, nous allons d'une part nous intéresser à quelques trajets particuliers, et d'autre part nous recentrer sur l'agglomération toulousaine. Plus précisément, nous nous restreindrons à la zone d'étude définie par les coordonnées GPS (43°30′ N,1°20′24″ E) et (43°38′60″ N,1°33′ E), puis l'enveloppe convexe définie par les trajets à vélo. La zone visée est un carré de 16,5 km de côté.

Les mesures collectées sont ponctuelles et ne correspondent pas à un trajet à vélo déterminé. Certains utilisateurs ont laissé allumé le système entre les trajets, en intérieur, en extérieur ou dans leur voiture. Il nous a donc fallu filtrer les données pour ne conserver que les moments de collecte en extérieur. En effet, outre les trajets à vélos, les moments où le vélo est garé en extérieur peuvent être intéressants pour ré-étalonner le capteur de MOx. Pour cela, nous décomposons d'abord le jeu de données en fonction du fait que la mesure est prise en mouvement ou non. Puis, nous distinguons d'une part les trajets en voiture des trajets à vélo et d'autre part les moments immobiles en intérieur et en extérieur.

Visuellement, nous observons une nette différence de qualité entre les positions calculées par le récepteur GPS avec 3 satellites et avec plus de 3 satellites. La Figure 4.5 – centrée sur la ville de Toulouse et ses environs – présente l'ensemble des positions calculées à partir de 3 et 4 satellites. La couleur correspond à la vitesse du système calculée à partir des positions successives. Nous observons que les points ont une vitesse de l'ordre de 100 km/h avec 3 satellites et 25 km/h avec 4 satellites. De plus, à partir de 4 satellites, les positions semblent suivre les routes. Néanmoins, il reste des positons imprécises (entourées en rouge dans la Figure 4.5). Ces positions correspondent soit à des phases d'initialisation du GPS soit à des moments où le système est en intérieur. Cependant, les positions dont le nombre de satellites synchronisés est supérieur ne corrige pas ce problème et supprime des trajets intéressants (fléchés en vert).

De plus, la répartition des mesures en fonction du nombre de satellites est équilibrée entre 3 et 7 satellites, avec environ un million de mesures, puis décroît significativement. Ainsi, nous filtrons les données en ne gardant que les positions dont le nombre de satellites synchronisés par le récepteur GPS est supérieur ou égal à 4. Cela supprime environ 15 % des points mais évite de prendre trop de temps à pré-traiter les données. En outre, nous ne conservons que les positions dont la vitesse est inférieure à 150 km/h. Cela nous permet de conserver toutes les positions des trajets en voiture (pour mieux les reconstruire et les filtrer) mais d'exclure les points aberrants.

Concrètement, afin de déterminer si une position est en mouvement, nous enrichissons notre jeu de données en calculant de nouvelles colonnes :

— la vitesse et le cap du vélo à l'aide des positions GPS,
— le nombre de fois où le cap du vélo change de sens sur une fenêtre glissante de 10 secondes,
— le nombre de fois où la valeur absolue de la vitesse projetée selon la direction du vélo dépasse 5 m/s sur une fenêtre glissante de 10 secondes,
— la distance entre la position courante et la position moyenne sur les 10 prochaines

secondes.
Elles sont mises à jour après chaque filtrage de données.

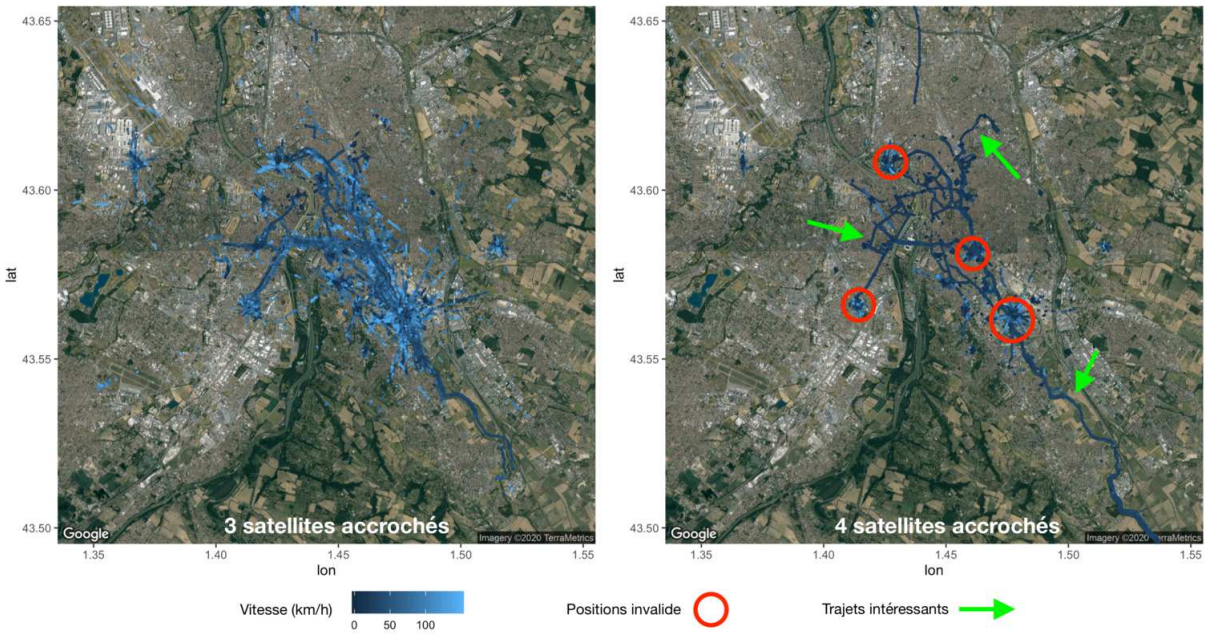


FIGURE 4.5 – Positions GPS collectées avec respectivement 3 et 4 satellites synchronisés. Les cercles rouges correspondent à des zones de positionnement imprécises et les flèches vertes indiquent des trajets visuellement identifiés comme corrects. La visualisation est recentrée sur Toulouse.

Puis, nous considérons que les données mobiles sont celles dont le cap du vélo ne change pas de sens au moins 80 % du temps dans une fenêtre de 10 secondes, dont la vitesse projetée selon la direction du vélo dépasse 5 m/s au moins 80 % du temps dans une fenêtre de 10 secondes et dont la distance entre la position courante et la position moyenne sur les 10 prochaines secondes est supérieure à 9 mètres. Nous ne conservons que les données mobiles dont 50 % du voisinage à 10 secondes sont aussi mobiles.

Puis, pour chaque système embarqué, nous déterminons les dates de début et fin des trajets en considérant qu'après toute période d'immobilisation de plus de 5 minutes, il s'agit d'un nouveau trajet. Nous pouvons alors reconstruire les trajets en récupérant toutes les positions entre les dates initiales et finales, et calculer de nouvelles informations sur les trajets : la durée, la longueur, la vitesse moyenne, la température médiane, le pourcentage de données initialement filtrées.

Ensuite, nous filtrons les trajets reconstruits qui sont liés à une imprécision des valeurs fournies par le récepteur GPS et non à un déplacement réel. Nous distinguons deux cas : ceux dus à l'initialisation du GPS durant laquelle la position converge vers la position réelle et ceux dus à l'imprécision du GPS, de l'ordre du mètre. Nous filtrons les trajets formés à cause de l'imprécision du GPS en supposant qu'il s'agit de ceux dont la vitesse moyenne du trajet est inférieure à 12 km/h, ou dont la distance entre le départ ou l'arrivée et la position moyenne du trajet (position correspondant à la moyenne des longitudes et à la moyenne des latitudes) est inférieure à 200 m. Puis nous filtrons les moments d'initialisation du GPS en considérant que ce sont ceux de vitesse moyenne de plus de 35 km/h ou de moins

de 2 minutes ou dont 70 % des valeurs avaient été filtrées initialement. Enfin, nous filtrons les trajets en voiture. Après exploration de nos données, nous considérons que ce sont ceux :

— dont la température médiane est supérieure à 35 °C,
— ou dont la température médiane est supérieure à 30 °C et la vitesse moyenne supérieure à 25 km/h,
— ou dont la température médiane est supérieure à 25 °C et la vitesse moyenne supérieure à 30 km/h.

Les trajets restants forment la base de trajets à vélo.

Nous formons également une base de moments immobiles en extérieur en fusionnant les positions qui n'appartiennent à aucun trajet et les trajets associés à une imprécision du GPS. Nous identifions les moments de mesure consécutifs en considérant qu'après toute période de plus de 5 minutes, il s'agit d'un nouveau moment. Enfin, nous ne conservons que les trajets de température médiane inférieure à 25 °C.

La chaîne de traitement pour reconstruire et filtrer les trajets est présentée Figure 4.6.

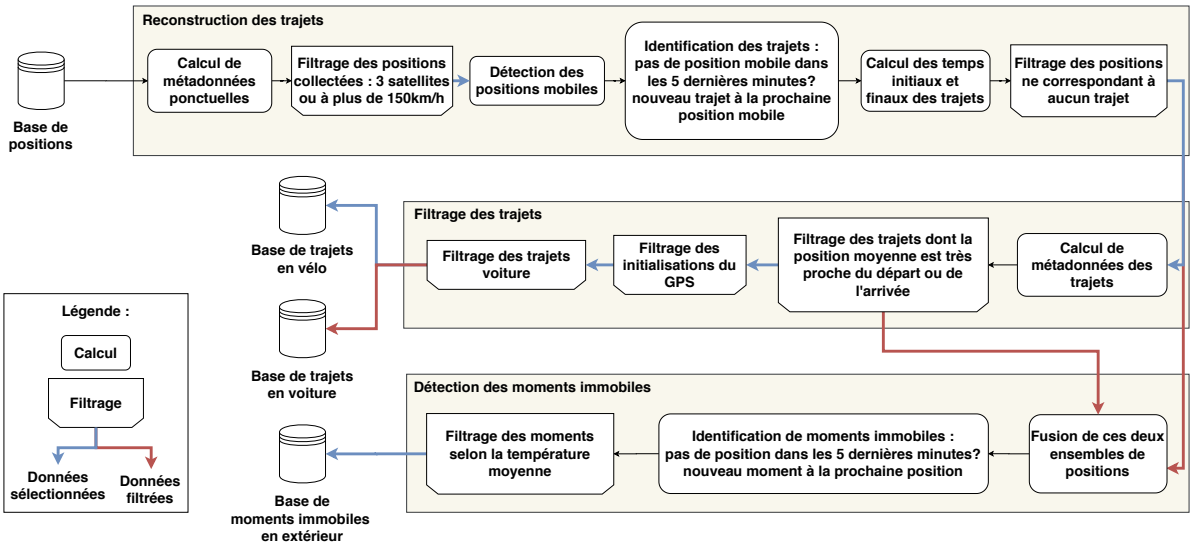
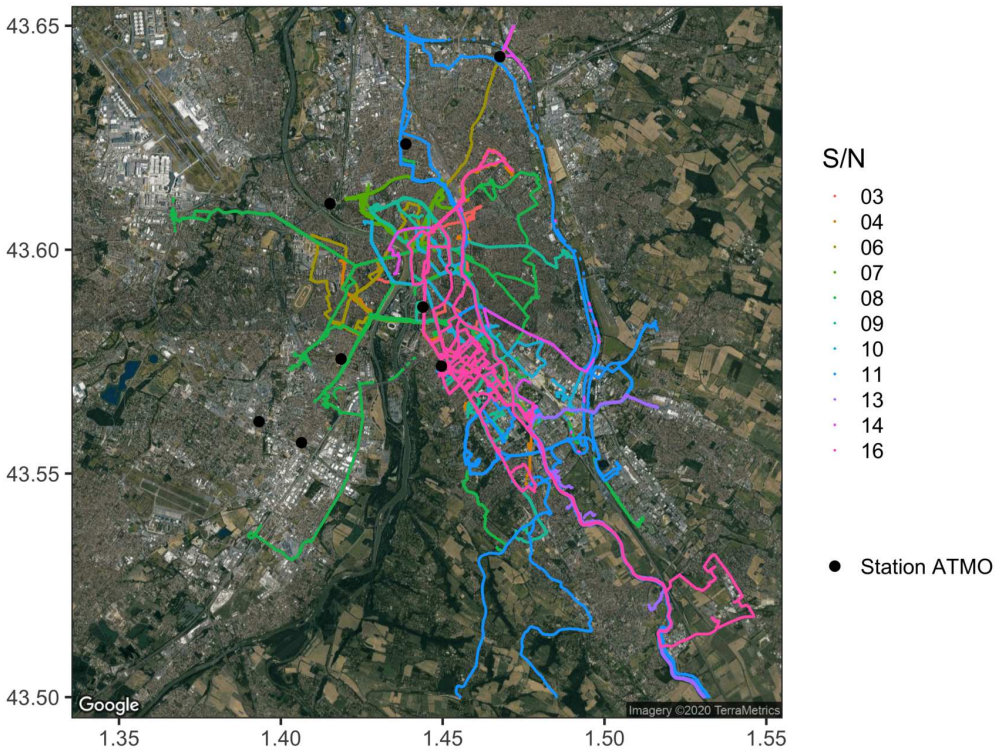


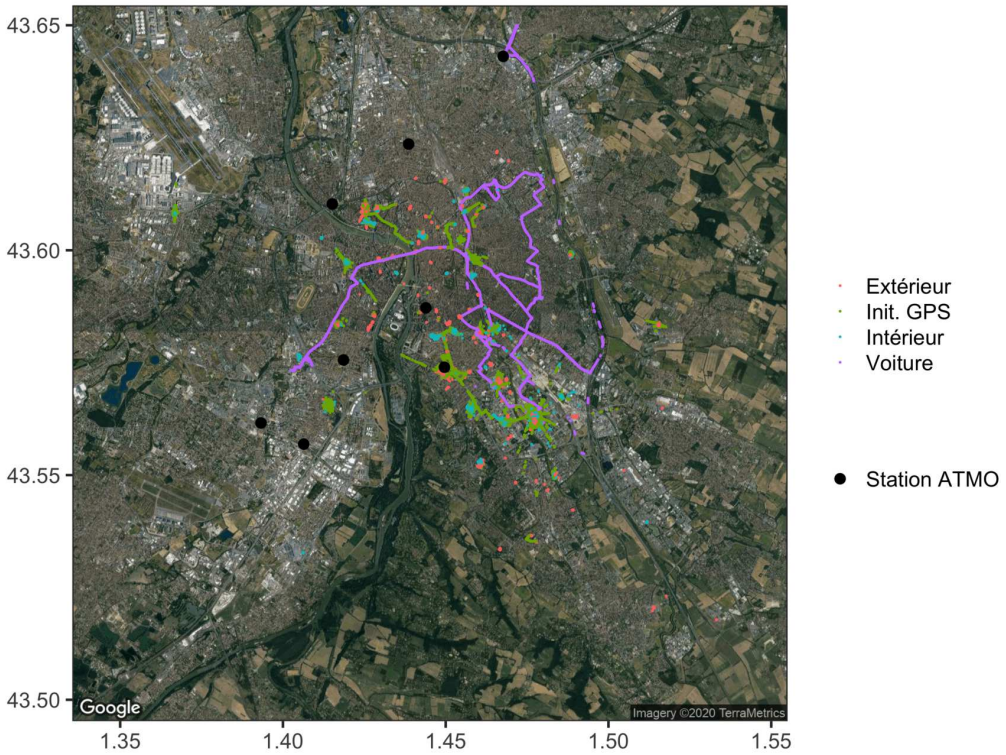
FIGURE 4.6 – Procédure de reconstruction des trajets à partir de la base de données des observations.

La Table 4.2 présente les résultats du filtrage de notre base de positions. Au total, les 599 trajets à vélo reconstruits représentent 6 % du jeu de données initial. Nous reconstruisons également 1084 moments immobiles en extérieur. Cependant, nous observons un biais introduit par l'utilisateur 9 auquel correspond plus de la moitié de ces moments. Sans lui, nous disposons de 423 moments immobiles en extérieur pour 528 trajets à vélo. La distance moyenne de ces moments immobiles est de 0,39 km à une vitesse moyenne de 2,8 km/h pour une durée moyenne de 30 minutes à une température moyenne de 19,6 °C. Cette vitesse non nulle est due au bruit de la position GPS. Ces statistiques paraissent cohérentes. La Figure 4.7 présente la base de trajets à vélo finale et les autres bases de données formées (moments en extérieur, en intérieur, trajets en voiture, initialisation du GPS). Nous observons une forte concentration de trajets à vélo entre le centre de Toulouse et notre laboratoire. De plus, nous notons qu'un trajet est sur la voie rapide ce qui semble

indiquer que certains trajets en voiture n'ont pas été filtrés.



(a) Base de trajets à vélo



(b) Trajets en voiture, initialisations du GPS, moments immobiles en extérieur et en intérieur

FIGURE 4.7 – Résultats du filtrage de notre base de points.

Table 4.2 – Résultats du filtrage de notre base de positions.

Utilisateur	Données brutes	% données à vélo	Trajets	Moments en extérieur	Trajets en voiture	Moments en intérieur	Init. GPS
1	29161	33	11	13	0	2	2
2	602579	3	31	0	13	74	85
3	201422	0	0	0	0	0	149
4	982856	3	32	0	4	122	453
5	2310417	5	57	10	0	430	508
6	13677	31	3	62	0	410	4
7	148	0	0	0	0	0	0
8	2287654	3	112	1	0	8	737
9	746075	21	71	661	1	89	230
10	80688	26	14	86	0	199	27
11	292871	11	36	12	0	28	42
12	522885	1	8	43	0	20	323
13	922447	6	61	2	1	290	336
14	273978	1	5	70	0	241	80
15	190084	3	10	0	0	68	29
16	23205	18	5	24	0	7	0
17	1116659	7	71	0	1	4	288
18	917917	8	72	100	0	162	322
Total	11514723	6	599	1084	20	2154	3615

4.3.3 Profils utilisateurs

La Figure 4.8 présente le nombre de positions acquises durant l'expérience par les vélos. Nous avons plus de 50 000 mesures – soit environ 14 h à raison d'une mesure par seconde – par mois à l'exception du mois d'Octobre. En effet, nous avons d'abord validé le protocole avec les vélo-taffeurs de notre équipe de recherche pendant le mois de Septembre avant d'étendre l'expérience à tout le laboratoire au mois d'Octobre. Les échanges des systèmes embarqués justifient cet écart dans le nombre de mesures. Le maximum de données est acquis au mois de février 2019 après une période de renouvellement de la base des utilisateurs (car nous ne disposions que de 12 systèmes pour 18 utilisateurs) et les systèmes ont été restitués durant le mois de Mars.

A échelle journalière, nous notons que les données sont assez bien réparties sur toute la durée de l'expérience à l'exception des périodes congés. La répartition horaire des observations est donnée par la Figure 4.9. Nous remarquons nettement le mouvement pendulaire de la circulation des vélos le matin et le soir. L'effet du mouvement pendulaire n'a pas été étudié au chapitre 2 car nous n'avions pas pris en compte la variation temporelle de la fréquence d'observation et l'évolution journalière des émissions. Néanmoins, nous nous rendons compte avec cette Figure que le biais d'observation induit par l'utilisation des vélo-taffeurs correspond aux pics journaliers de pollution (cf. section suivante). La Table 4.3 présente des statistiques concernant le profil des utilisateurs.

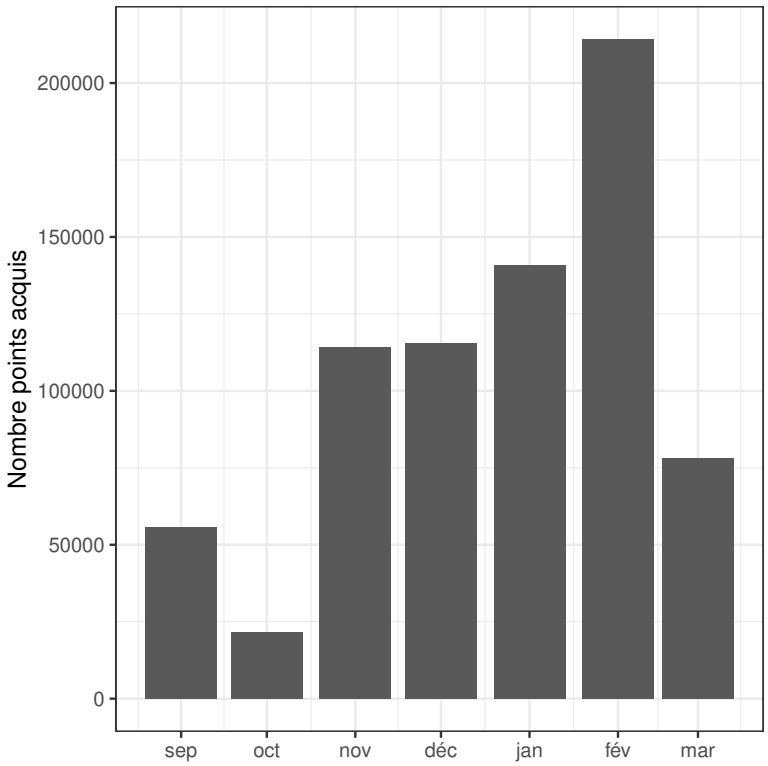


FIGURE 4.8 – Nombre points GPS acquis par mois durant l'expérience en 2018 et 2019.

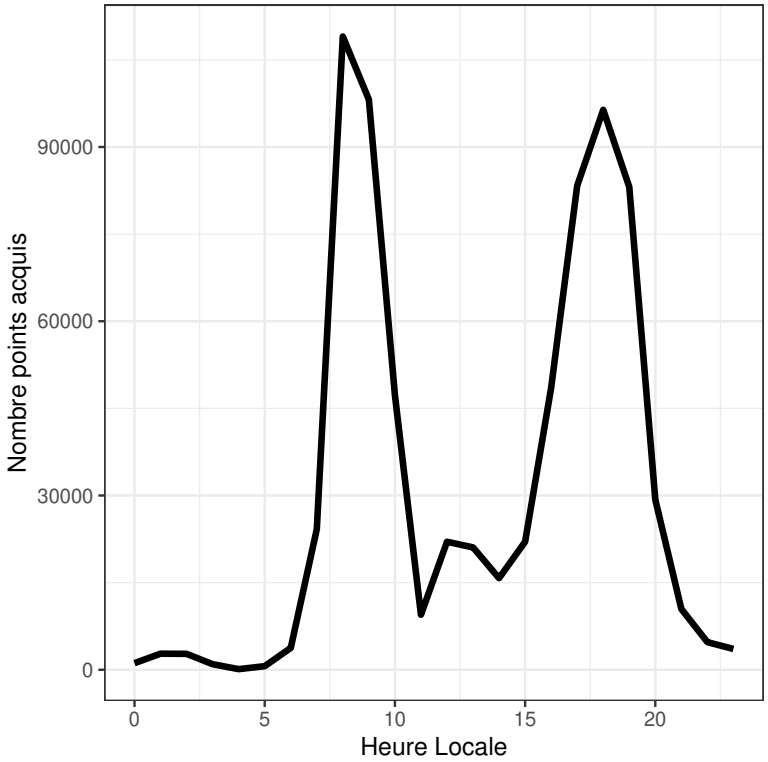


FIGURE 4.9 – Nombre points GPS acquis par mois durant l'expérience en 2018 et 2019.

Nous observons qu'un trajet dure en moyenne 20 minutes, à une vitesse de 16,66 km/h sur une distance de 5,7 km. L'utilisation, décrite par le nombre de jours pendant lesquels le vélo a été utilisé rapporté au nombre de jours de prêt, est très variable entre les utilisateurs, allant de 1 % à 42 %. Néanmoins, lorsque nous regardons tous les trajets collectés, au moins un de nos systèmes a été actif plus d'un jour sur deux pendant l'expérience. De plus, lorsqu'un trajet est effectué, il est généralement accompagné d'un second. En effet, en se focalisant sur les jours pendant lesquels le vélo a été utilisé, le nombre de trajets est en moyenne de 2,67 par utilisateur et de 5,21 pour l'ensemble des trajets collectés. Autrement dit, pour 11 systèmes en fonctionnement (l'utilisateur 7 avait un système durant tout la période et aucun trajet n'a été collecté), il y en a environ un quart qui fonctionne un jour sur deux.

Table 4.3 – Statistiques décrivant les utilisateurs.

Utilisateur	Jours de prêt	Trajets	% de jours d'utilisation	Trajets/jour utilisé	Durée moyenne (min)	Distance moyenne (km)	Vitesse moyenne (km/h)
1	44	11	18	1,38	14,83	4,28	17,21
2	81	31	11	3,44	14,22	3,64	17,48
3	19	-	-	-	-	-	-
4	420	32	3	2,29	21,55	7,10	15,51
5	124	57	18	2,48	36,87	11,18	17,47
6	57	3	3	1,50	24,81	6,00	16,42
7	67	-	-	-	-	-	-
8	118	112	31	3,03	11,58	2,85	15,30
9	157	71	21	2,09	38,07	15,13	23,16
10	103	14	7	1,75	26,76	6,40	15,57
11	71	36	23	2,12	16,79	4,90	18,64
12	151	8	2	2,00	22,28	5,32	16,16
13	69	61	33	2,65	17,77	4,87	17,16
14	59	5	5	1,67	14,63	3,79	16,21
15	83	10	4	2,50	12,33	2,13	14,37
16	59	5	1	5,00	14,50	3,08	14,13
17	71	71	42	2,37	21,57	5,33	15,82
18	121	72	14	4,00	19,15	5,03	16,01
Tous trajets	210	599	54	5,21	20,48	5,69	16,66

Dans la mesure où certains utilisateurs ont un nombre de trajets moyen inférieur à 2, nous supposons que notre système n'a pas été pleinement actif à tout moment. Puisqu'il ne semble pas que nous ayons filtré de trajets à vélos, le système devait être éteint (par choix, oubli ou dysfonctionnement) ou transporté en voiture. Cependant, cela ne semble pas affecter les statistiques d'utilisation en terme de durée, vitesse, et distance moyennes.

En comparaison avec la simulation effectuée sur Marseille, la base de vélo-taffeurs étudiée est différente. Concernant la distribution des longueurs des trajets, nous savons (et le confirmons avec les statistiques d'utilisation) que les trajets sont plus longs que la normale car il s'agit d'un critère de choix des utilisateurs.

De plus, dans le cas réel, tous les vélo-taffeurs proviennent de la même entreprise. Nous observons que ces trajets ne concernent pas uniquement les trajets maison–travail mais sont concentrés sur le centre-ville et le lieu de travail. Dans notre simulation sur la ville de Marseille, nous avions défini deux zones attractives (le centre-ville et les plages). Cet aspect semble donc cohérent.

Ensuite, concernant le nombre de trajets quotidiens, nous observons qu'environ un quart des systèmes embarqués sont utilisés environ un jour sur deux. Il nous faudrait

donc multiplier le nombre d'utilisateurs par 20 pour satisfaire le nombre minimal de trajets quotidiens afin de fournir une simulation dont la variance de l'erreur du modèle est stable (cf. section 2.4.1).

La Figure 4.10 présente la distribution des types de voies pour l'agglomération toulousaine (zone d'étude) et les données collectées par les vélos. Nous observons que les routes de type *cycleway* (piste cyclable), *primary* (axes principaux) et *tertiary* (routes reliant à des villages ou hameaux) sont sur-représentées dans les données collectées par les vélos comparativement à la zone d'étude. A l'inverse, les routes de type *residential* (zone résidentielles), *service* (voies privées, accès aux bâtiments) et les routes impraticables (*steps* pour des escaliers, *pedestrian* pour une voie piétonne, *motorway_link* pour les jonctions d'autoroutes, *subway* pour les métros et *rails* pour les voies ferrées) sont sous-représentées. De plus, nous observons une proportion non négligeable de trajets sur des routes de type *motorway* par nos cyclistes. Il s'agit de trajets en voiture non filtrés.

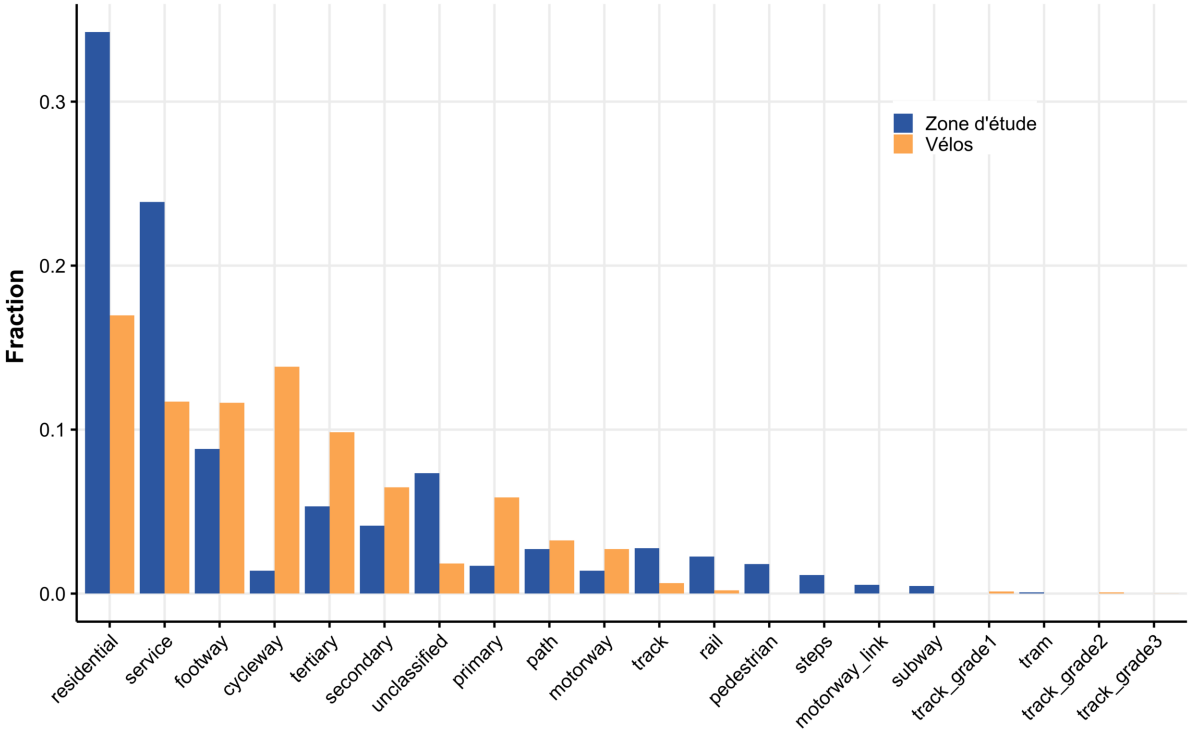


FIGURE 4.10 – Distribution des types de route pour toute la zone d'étude, nos données collectées par les vélos et les trajets simulés au chapitre 2.

Comparativement à l'étude sur Marseille, nous constatons que les écarts de densité de la distribution du type de voies entre la zone d'étude et les trajets des vélos (simulés sur Marseille, réels sur Toulouse) suivent la même tendance, à l'exception des zones piétonnes qui n'ont quasiment pas été empruntées par nos vélos. Néanmoins, les écarts de densité pour les pistes cyclables et les voies principales (*primary*) sont davantage accentués dans la réalité que dans notre simulation. Cela peut être dû au fait qu'il y a plus de pistes cyclables à Toulouse qu'à Marseille et que les utilisateurs empruntent des voies qu'ils connaissent (*primary*) et ne cherchent pas le trajet optimal à chaque déplacement.

En conclusion, notre simulation semble relativement bien décrire la distribution du type de voies et l'attractivité de deux points de la zone d'étude. En outre, la longueur des trajets simulés est inférieure à la réalité. Elle semble donc utilisable au regard des ces critères. Toutefois, le nombre de trajets collectés ne satisfait pas les exigences définies pour assurer une erreur de modélisation minimale par nos méthodes de spatialisation à un pas de temps journalier. Nous spatialisons les données collectées sur toute la période. Pour obtenir plus de trajets, nous aurions dû davantage sélectionner nos utilisateurs car certains se sont très peu servis des systèmes. En outre, sans surprise, il aurait été profitable de bénéficier de plus de systèmes de mesure et de plus d'utilisateurs.

4.4 Évaluation de l'état de la pollution dans Toulouse

Dans cette analyse, nous nous servons des résultats de l'évaluation *in situ* des performances du capteur MiCs-4514 (cf. chapitre 3) pour traiter les données collectées dans la ville de Toulouse.

Nous en avions conclu que normaliser les réponses des capteurs à l'aide de la valeur médiane observée au cours de l'éxperience permet d'obtenir des réponses équivalentes entre capteurs.

Dans cette logique, il nous semble judicieux d'utiliser la base de moments immobiles en extérieur pour effectuer cette normalisation. Cela permet de corriger la dérive temporelle des capteurs. Pour cela, les moments immobiles les plus intéressants sont ceux en milieu rural. En effet, en milieu rural, les variations de concentrations observées sont plus faibles qu'en milieu urbain. Les valeurs médianes exploitées par les capteurs pour la renormalisation auront statistiquement plus de chance de représenter une même concentration réelle de polluant. Dans notre cas, un autre lieu intéressant est notre laboratoire car nous sommes assurés que tous les vélos s'y rendent. Cela n'a pas été mis en œuvre par manque de temps.

Nous aurions aussi pu étudier la normalisation par la valeur médiane en fonction de l'utilisateur (et non plus uniquement en fonction du système embarqué). Cela aurait traduit les différences de pratiques (impact de la vitesse de l'utilisateur, position du système sur le vélo...). Nous avons juste normalisé les réponses de nos capteurs par leurs valeurs médianes au cours de la totalité de l'expérience.

En outre, au chapitre 3, nous n'avions ni pu valider la décorrélation des réponses en fonction des paramètres météorologiques ni la correspondance des réponses aux concentrations réelles. Nous ne reprendrons donc pas cette démarche ici.

Avant de nous intéresser aux mesures effectuées par nos vélos, nous présentons l'état de la pollution de l'air dans Toulouse à l'aide des stations de référence déployées par ATMO Occitanie.

4.4.1 Analyse temporelle des mesures ATMO Occitanie sur Toulouse

Au chapitre 2, seule la variabilité spatiale des concentrations a été prise en compte. Chaque simulation était considérée comme indépendante et l'ensemble des observations

acquises de manière instantanée. Dans le cas de mesures réelles, les concentrations de polluants subissent une variation temporelle durant le déplacement des cyclistes. Afin d'appréhender cette variation, ce paragraphe fait une première analyse de la variabilité temporelle des concentrations de polluants à l'échelle de l'agglomération de Toulouse à partir des mesures enregistrées sur les stations fixes de surveillance de la qualité de l'air. Les données proviennent de l'Open Data de ATMO Occitanie [2].

Durant la période d'expérimentation, entre septembre 2018 et mars 2019, les stations de mesure du réseau ATMO Occitanie ont enregistré les concentrations de différents polluants réglementaires dans la ville de Toulouse (cf. distribution spatiale des stations au chapitre 1). Nous présentons ici quelques statistiques des variations de NO_2, CO et PM_{10}. Les données sont extraites de l'Open Data d'ATMO Occitanie en moyenne horaire. Toutes les stations sont de type « station urbaine » et sont catégorisées par influence : trafic, fond et industrielle [3]. La Table 4.4 présente les valeurs horaires minimales, moyennes et maximales, en NO_2 et PM_{10}, observées sur les différentes stations. Les 3 stations « trafic » sont situées en bordure de périphérique. Elles présentent les moyennes horaires les plus élevées, entre 43 $\mu g/m^3$ et 69 $\mu g/m^3$ pour le NO_2 et 24 $\mu g/m^3$ et 29 $\mu g/m^3$ pour les PM_{10}. Les stations « trafic » ont une concentration moyenne 2 fois supérieure à celle des stations de fond. La gamme de mesures s'étend de concentrations quasi nulles à environ 200 $\mu g/m^3$ de NO_2 et 124 $\mu g/m^3$ de PM_{10} sur une des stations sous influence industrielle. Cette gamme de mesures est uniquement indicative car les valeurs instantanées dans le contexte d'une mesure sur vélo, donc très proche des sources, peut fluctuer davantage.

Table 4.4 – Concentrations de NO_2 et de PM_{10} en $\mu g/m^3$ observées aux stations de ATMO Occitanie à Toulouse durant la période d'étude.

Station	influence	NO_2			PM_{10}		
Toulouse		moy	min	max	moy	min	max
Rte Albi Trafic	Trafic	43	2	188	24	-1	122
Port de l'Embouchure	Trafic	46	1	167	24	2	109
Périphérique Trafic	Trafic	69	4	217	29	1	113
Berthelot Urbain	Fond	21	2	119	16	-2	68
Jacquier Urbain	Fond	22	0	122	18	-1	122
Mazades Urbain	Fond	21	0	113	17	-2	85
SETMI Chapitre	Industrielle	-	-	-	20	-1	124
SETMI Eisenhower	Industrielle	-	-	-	19	-0	122

La représentation graphique de la série temporelle des maxima journaliers en NO_2 et PM_{10} est donnée Figure 4.11 pour chacune des stations. Les maxima journaliers ont la même tendance que les valeurs moyennes. Nous constatons que la variabilité journalière est fortement corrélée entre les différentes stations pour un polluant donné. En moyenne journalière de NO_2, le coefficient de corrélation entre la valeur moyenne des stations de fond et de trafic est R=0,89. En moyenne horaire, ce coefficient tombe à R=0,71. Pour les PM_{10}, ce coefficient journalier est de R=0,93 (respectivement R=0,81 en moyenne horaire).

2. `http://data-atmo-occitanie.opendata.arcgis.com`

3. voir le rapport du Laboratoire Central de Surveillance de la Qualité de l'Air sur la conception, l'implantation et suivi des stations françaises de surveillance de la qualité de l'air de février 2017.

Nous en déduisons que les stations fixes ont tendance à observer les mêmes phénomènes mais avec des amplitudes différentes en fonction de leur distance aux sources. Les concentrations entre les 2 polluants sont également fortement corrélées. En moyenne horaire, le coefficient de corrélation entre les PM_{10} et le NO_2 est R=0,69 pour les stations trafic et R=0,55 pour les stations de fond. Cette différence est retrouvée lorsque nous considérons les stations de manière individuelle (R=0,74 pour la station Toulouse-Périphérique). La corrélation entre les polluants pour les stations de fond est moins importante. Cela peut s'expliquer par l'influence de sources plus hétérogènes pour celles-ci comparativement aux stations trafic qui mesurent essentiellement les émissions du trafic routier sur le périphérique.

A l'échelle de la journée, il existe également une variabilité diurne. Pour des polluants primaires tels que le NO_2 et les PM_{10}, celle-ci est principalement due à la variabilité des sources et donc du trafic urbain. La Figure 4.12 représente le cycle journalier des polluants par classe de stations. Le cycle journalier de la concentration en CO est également représenté. Le CO n'étant plus mesuré depuis fin 2018, les données utilisées pour ce polluant couvrent la période du 18 février au 6 juillet 2018. Le cycle journalier est dominé par la fluctuation du trafic routier le matin et le soir. Le pic du matin est situé entre 7:00 LT et 8:00 LT celui du soir entre 18:00 LT et 19:00 LT. Ces pics sont plus tardifs sur les PM_{10} pour les stations de fond, indiquant probablement un phénomène de dilution des émissions en provenance du périphérique, ou un effet de la dynamique locale lié au cycle diurne de la couche limite atmosphérique. L'amplitude journalière du NO_2 est de $55\,\mu g/m^3$ pour les stations trafic et de $20\,\mu g/m^3$ pour les stations de fond. L'amplitude est moindre pour les PM_{10}, mais nous retrouvons un facteur 2 entre les stations de fond et les stations trafic. Le contraste journalier est 2 fois plus important pour le cycle du NO_2 que pour les PM_{10}. Nous retrouvons donc bien une hétérogénéité temporelle plus importante pour le NO_2 que les PM_{10} en relation à l'hétérogénéité spatiale plus importante également (cf. chapitre 2). Le cycle journalier du CO sur la Figure 4.12 met également en évidence la variation journalière du trafic. Néanmoins le pic matinal est plus important que le pic en soirée. Cette différence est encore plus visible lorsque nous sélectionnons les observations en fonction de la saison. Il est probable que cette différence soit liée à l'efficacité de combustion des moteurs thermiques, plus froid le matin et en hiver, avec plus d'émission de CO et moins d'émission de NO_x. Nous distinguons également un pic en milieu de journée en relation avec la circulation durant la pause méridienne. Même si les concentrations de CO sont inférieures à la réglementation, il apparaît important de mesurer ce polluant comme traceur des émissions par le trafic.

Durant notre période d'observation il existe également un cycle hebdomadaire, représenté sur la Figure 4.13. Nous remarquons une nette diminution des concentrations le week-end. Le minimum des concentrations est atteint le dimanche. Le cycle des PM_{10} est plus marqué que celui du NO_2 avec une augmentation graduelle durant la semaine. Cette courbe est cohérente avec l'effet d'accumulation des émissions et le temps de résidence des polluants (cf. chapitre 1). Nous en déduisons que l'émission de polluants est quasi-constante durant la semaine et diminue drastiquement durant les jours de week-end, essentiellement en zone trafic (le périphérique).

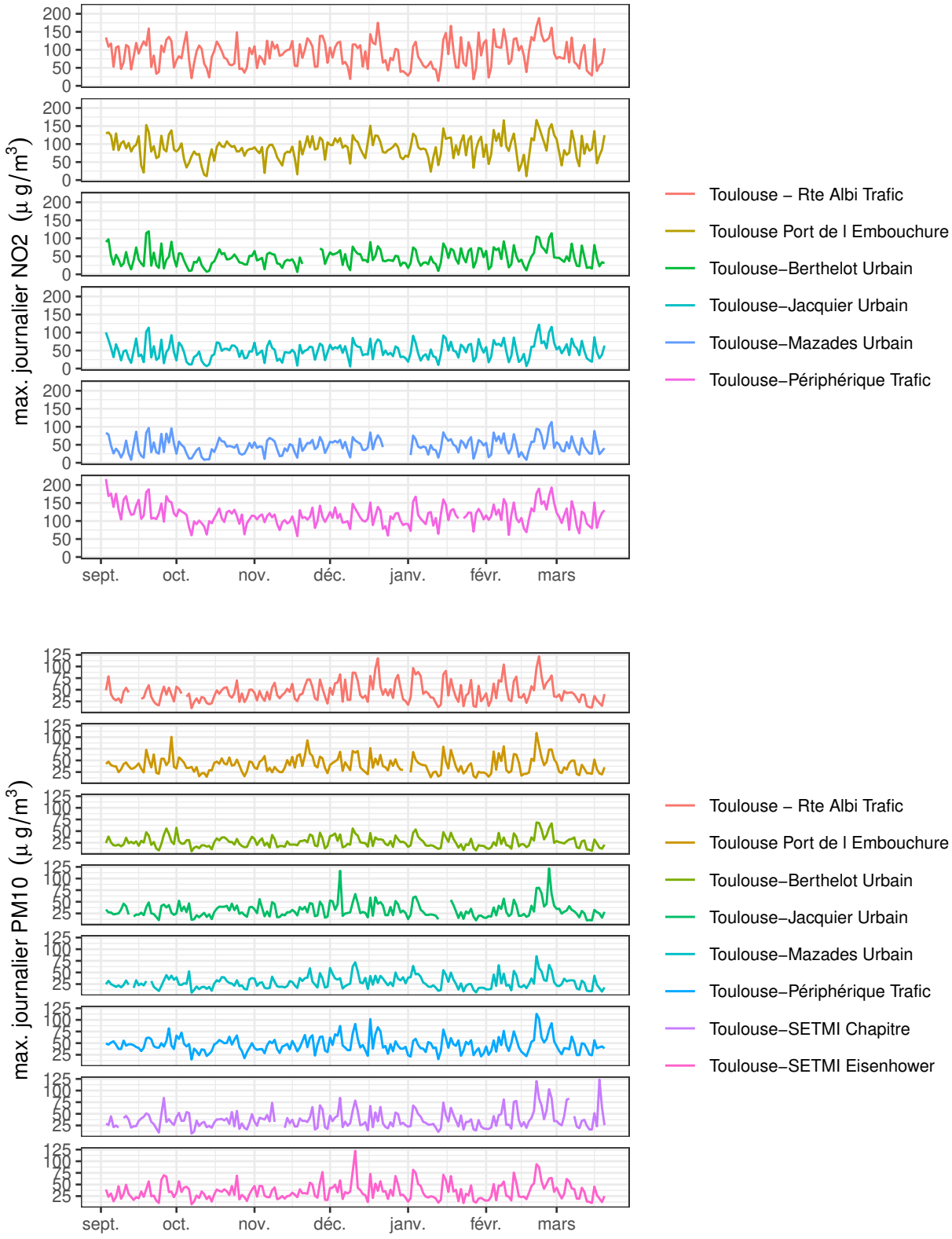


FIGURE 4.11 – Maximum journalier en concentration de NO_2 et de PM_{10} à Toulouse pour les différentes stations entre Sept. 2018 et Mars 2019 (données ATMO Occitanie).

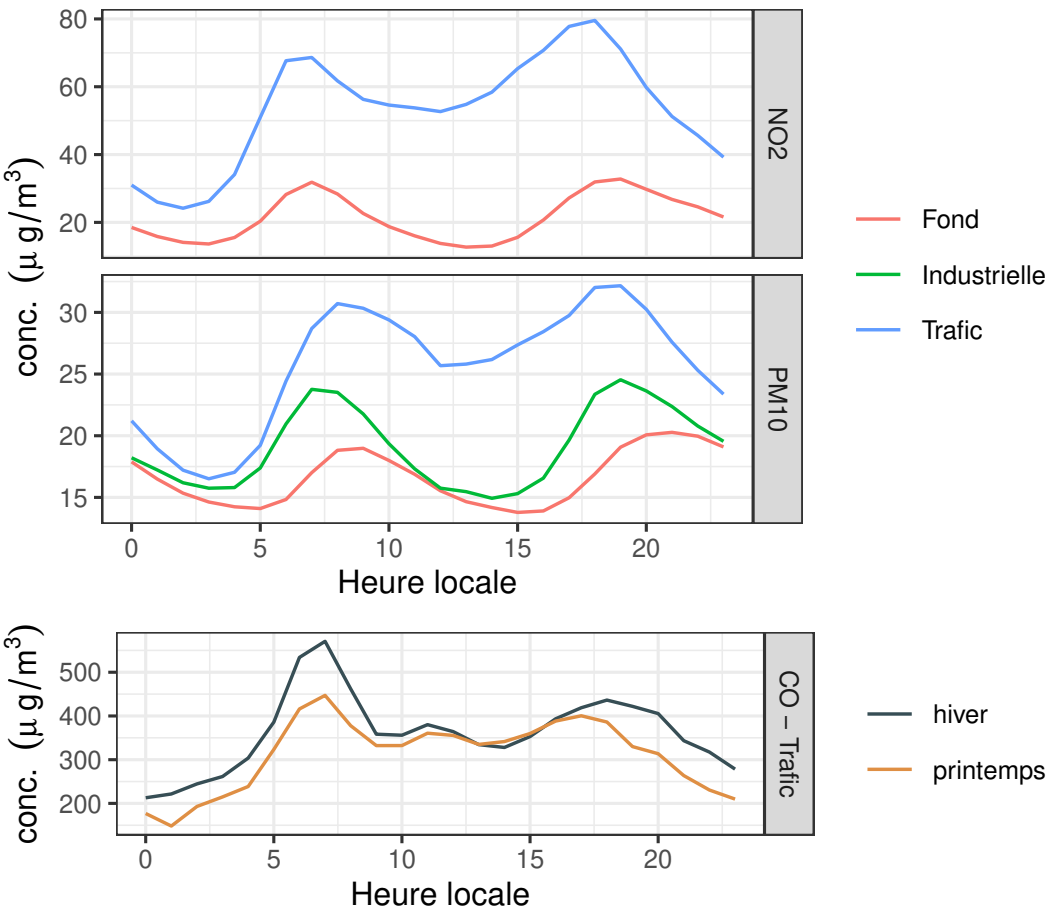


FIGURE 4.12 – Cycle journalier moyen pour le NO_2 et les PM_{10} en fonction de l'influence. Cycle journalier moyen du CO pour la station trafic en hiver et au printemps.

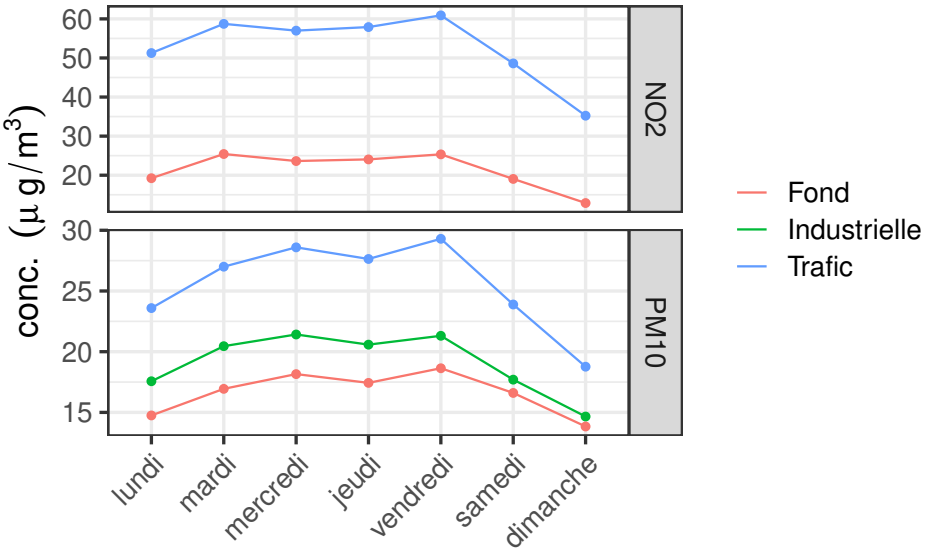


FIGURE 4.13 – Cycle hebdomadaire pour le NO_2 et les PM_{10}.

4.4.2 Analyse de trajets particuliers

Dans cette thèse, nous nous intéressons fortement au *Rendez-Vous*.

Pour que deux capteurs se rencontrent, il faut qu'ils soient dans une même zone de l'espace au même moment. Lorsque c'est le cas, la notion de *Rendez-Vous* suppose que les phénomènes qu'ils observent sont les mêmes.

Nous nous intéressons ici à trois cas particuliers afin d'examiner le réalisme de cette hypothèse.

Le premier est un *Rendez-Vous* idéal durant toute la durée du trajet : deux vélos roulent côte à côte pendant 10 minutes. Le second est un *Rendez-Vous* idéal spatialement sans considération de la temporalité : les variations locales observées par les vélos sont étudiées à l'échelle d'un quartier. Le troisième est un *Rendez-Vous* réaliste : deux vélos se déplacent dans une même zone de l'espace, en empruntant la même route pendant 2 minutes à 5 minutes d'écart.

Dans le premier cas, les réponses des capteurs sont extrêmement bien corrélées. A une fréquence de 1 Hz, nous obtenons un coéfficient de corrélation de 0,94 pour le NO_2, 0,92 pour le CO et 0,98 pour les conditions météorologiques.

Nous constatons un décalage constant entre les deux capteurs, et ce pour chaque grandeur physique. Nous nous attendions à observer la même valeur pour les BME280, réputés de réponse constante, mais ce n'est pas le cas. Cela peut être dû aux conditions d'expérimentation sensiblement différentes entre les deux instruments de mesure (position sur le vélo, flux d'air sur le capteur...) ou à une dérive d'un de ces capteurs.

En normalisant non plus avec la valeur mediane au cours de l'expérience mais au cours du trajet, les réponses sont strictement identiques. Cette méthode n'est pas envisageable en l'état pour traiter tout le jeu de données, car elle aurait pour conséquence de fournir des valeurs moyennes identiques pour un trajet en ville et pour un trajet en milieu rural. Toutefois, nous en déduisons que les capteurs observent bien le même phénomène.

Cette méthode ouvre une piste intéressante pour construire un algorithme définissant une relation d'ordre partiel sur l'ensemble des régions de la zone d'étude. En effet, cette méthode permet de déterminer les variations parmi les lieux rencontrés au cours d'un trajet, et donc par transitivité, de tous les lieux rencontrés par les vélos.

Dans le deuxième cas, nous observons les réponses normalisées de tous les capteurs dans un quartier connu : l'île du Ramier. Il s'agit du futur « poumon vert » de la ville de Toulouse (aménagements en cours). Le pont Pierre de Coubertin est un grand axe routier de la ville possédant deux fois deux voies, très fréquenté aux heures de pointes, et enjambant l'île. Il fait la jonction entre deux quartiers emblématiques de la ville de Toulouse : les Arènes et Saint-Michel. Le reste du quartier ne souffre pas spécialement des embouteillages.

Pour observer les réponses normalisées de nos capteurs dans ce quartier, nous rasterisons les trajets (cf. Figure 4.14). Pour une cellule du raster, nous prenons la moyenne des valeurs observées au cours de l'expérience. Les axes routiers principaux se dénotent fortement des axes secondaires. De plus, nous observons également une différence de réponses aux extrémités latérales des routes (vers les trottoirs), notamment le long du pont Pierre de Coubertin. Au regard des modélisations du type Inverse Distance Weighting (IDW) (cf. chapitre 2), nos données semblent donc spatialement cohérentes : la concentration

diminue significativement au regard de la distance à la route. Ces données pourraient éventuellement servir à paramétriser ce genre de modélisation.

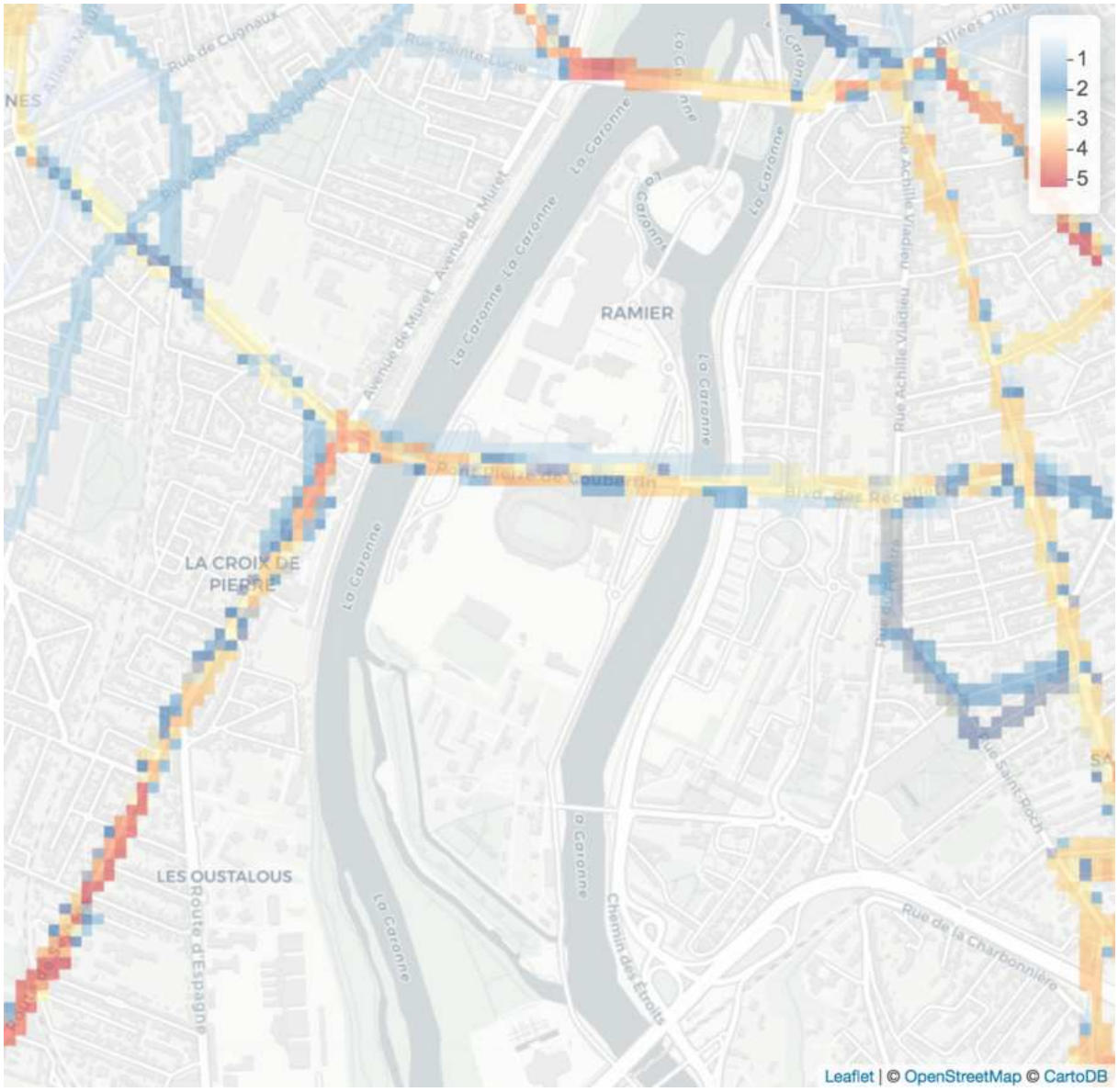


FIGURE 4.14 – Raster des réponses normalisées en NO_2 observées au cours de la totalité de l'expérience, centré sur l'île du Ramier. Résolution de 25 m × 25 m.

Dans le troisième cas, les vélos empruntent le rond-point du Grand Rond à 5 minutes d'écart pendant 2 minutes. Pour étudier le *Rendez-Vous*, nous pourrions simplement décaler la réponse de l'un par rapport à l'autre de 2 minutes. Cependant, cette approche n'est pas consistante avec tous les cas de *Rendez-Vous* imaginables. Par exemple, si un vélo est à l'arrêt et qu'un autre passe devant, cette technique n'a plus de sens.

Pour comparer les deux réponses, nous nous basons sur les coordonnées géographiques. Nous regardons les distances entre tous les points des deux trajets. En oubliant la notion de temporalité, pour chaque position d'un capteur, nous faisons correspondre la réponse du second observée en la position minimisant cette distance. Puis, nous retrouvons la

temporalité (quoique fictive) en se référant à l'un des deux capteurs. La Figure 4.15 présente ces réponses pour le NO_2. Nous représentons également la distance entre les capteurs en bleu (centrées-réduites sur les valeurs du premier capteur pour la lisibilité de la courbe). Le moment où les vélos empruntent le rond-point s'observe très nettemment : il s'agit du moment minimisant la distance. Le délai entre les vélos (en rose) permet d'appréhender la vitesse relative des deux vélos.

Les deux capteurs voient leurs réponses significativement augmenter lorsque leur distance est minimisée (rectangle bleu foncé sur la Figure 4.15). Ceci s'explique au regard du lieu (rond-point). Avant leur rencontre, ils étaient sur une piste cyclable (respectivement le long des allées Jules Guesde et le longs des allées Forain-François Verdier) et après ils rejoignent une route, puis une autre piste cyclable.

Nous observons des saturations de la réponse du deuxième capteur au regard du premier avant 19:07:40 et après 19:09:30. Ceci traduit l'éloignement des capteurs entre eux (hors du rectangle bleu clair sur la Figure).

Entre 19:07:40 et 19:09:30, la corrélation entre les réponses est de 0,73.

Nous pouvons donc les considérer en *Rendez-Vous* durant cette période. Dans le pire des cas, la distance les séparant au cours du *Rendez-Vous* est de 140 m.

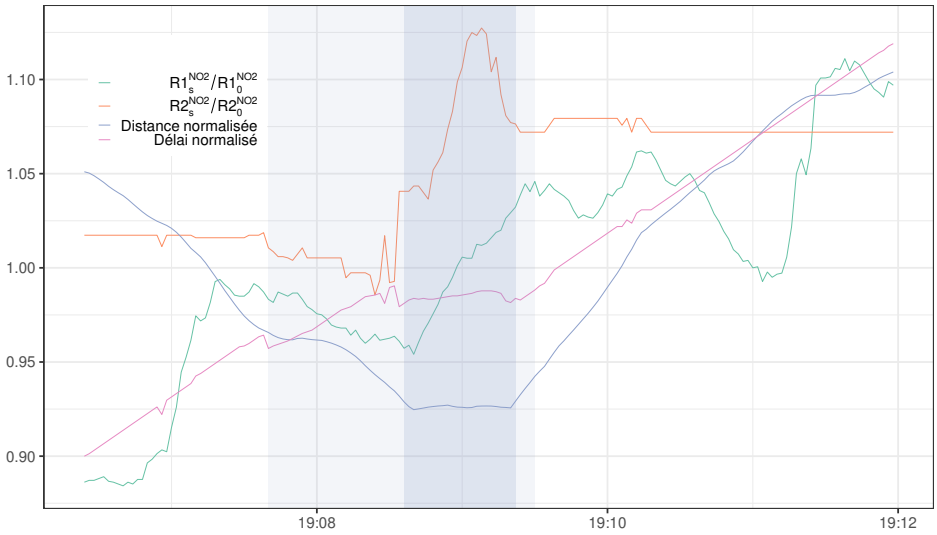


FIGURE 4.15 – Réponse normalisée d'un capteur et réponse normalisée du second capteur aux instants minimisant la distance entre les vélos. La zone en bleu clair correspond au moment où les réponses des deux capteurs sont corrélés. La zone en bleu foncé correspond au moment où la distance les séparant est minimale.

Ainsi, nos capteurs observent bien des phénomènes identiques lorsqu'ils se rencontrent, même avec un délai spatio-temporel. La notion de *Rendez-Vous* semble exploitable dans des conditions réelles pour l'étalonnage de notre flotte de MiCs-4514.

Nous aurions également voulu analyser la notion que nous nommerons « *Rendez-Vous* par évènement réccurent » (dont nous n'avons pas trouvé mention dans la littérature). En effet, la notion classique de *Rendez-Vous* exploite le fait que deux capteurs observent le même phénomène au même moment. Puisqu'il y a une forte corrélation entre les concentrations observées chaque jour, notamment en distinguant les jours ouvrés et les jours du

week-end, il serait intéressant d'étudier ce qu'observent nos capteurs en des lieux identiques, à des heures identiques, pour des jours différents. Dans la même logique, il serait intéressant d'étudier ce qu'observent nos capteurs en des lieux fortement similaires, tel que l'ensemble du périphérique, au même moment, i.e. la fenêtre spatiale ne serait donc plus définie uniquement à l'aide d'une distance euclidienne.

C'est sur la base de cette idée que nous poursuivront l'analyse par spatialisation de la pollution dans la ville de Toulouse à la dernière section.

4.4.3 Étalonnage collaboratif par *Rendez-Vous*

Lors d'un *Rendez-Vous*, l'un des deux systèmes peut servir de référence (reference-based calibration) ou pas (blind calibration). Dans notre cas, nous étudions ces deux types d'étalonnage. Ce sont les stations fixes déployées par ATMO qui jouent le rôle de stations de référence.

Il faut alors définir ce qu'est la distance acceptable pour que deux systèmes soient dans la même zone de l'espace et la durée d'un moment. La section précédente suggère que la taille des fenêtres spatiales intéressantes pour définir un *Rendez-Vous* peuvent au moins aller jusqu'à 140 m.

Nous faisons varier la taille de cette fenêtre spatio-temporelle et observons le nombre de rencontres des systèmes entre eux et avec les stations fixes d'ATMO.

Concrètement, nous déterminons un *Rendez-Vous* en calculant les distances entre toutes les mesures d'une paire de systèmes, puis en ne conservant que les positions dont la distance est inférieure à la fenêtre spatiale, et enfin en découpant cette succession de positions en ensembles disjoints espacés dans le temps par au moins la fenêtre temporelle.

Les résultats sont présentés dans les Tables 4.5 et 4.6. Plus la fenêtre spatiale du *Rendez-Vous* est grande, plus il y a de *Rendez-Vous*. Plus la fenêtre temporelle du *Rendez-Vous* est grande, moins il y a de *Rendez-Vous*. Ceci s'explique par le fait qu'en abaissant la résolution temporelle, plusieurs *Rendez-Vous* peuvent par la suite n'en constituer qu'un. C'est par exemple le cas lorsque deux systèmes se rencontrent le matin et le soir ; à un pas horaire il y aura deux *Rendez-Vous*, à un pas journalier, il n'y aura qu'un *Rendez-Vous*.

En outre, nous constatons que pour un *Rendez-Vous* entre deux capteurs mobiles, la fenêtre temporelle est plus importante que la fenêtre spatiale. En effet, nos utilisateurs proviennent du même laboratoire et fréquentent des routes similaires (notamment la piste cyclable le long du canal du Midi et les principales artères de la ville). Leur rencontre spatiale a lieu dans une fenêtre très réduite.

Pour un *Rendez-Vous* avec une station de référence, puisqu'elle fonctionne en continu, la fenêtre spatiale est plus importante que la fenêtre temporelle.

Dans le cas où nous considérons la fenêtre spatio-temporelle définie par la zone d'étude et la durée totale de l'expérience, tout système est en *Rendez-Vous* en permanence. Les 11 systèmes sont donc chacun en *Rendez-Vous* avec les 10 autres et avec les 8 stations de référence, il y a $\frac{11*(11-1)}{2} = 55$ *Rendez-Vous* entre capteurs et $11 * 8 = 88$ *Rendez-Vous* avec les stations.

Pour l'étude de la pollution de l'air, nous disposons de données horaires fournies par les stations d'ATMO Occitanie. La fenêtre spatio-temporelle (250 m, 1 h) – à laquelle correspondent les cellules en vert dans les Tables 4.5 et 4.6 – nous semble être un bon

compromis. Au regard de la section précédente, nous supposons qu'elle conserve la propriété essentielle du *Rendez-Vous*, i.e. observer le même phénomène, tout en assurant un nombre de *Rendez-Vous* suffisant pour effectuer un étalonnage.

TABLE 4.5 – Nombre de *Rendez-Vous* entre nos 11 systèmes embarqués, en fonction de la fenêtre spatio-temporelle du *Rendez-Vous*.

distance (m) \ durée	1 minute	1 heure	1 jour	toute la période
25	3488	1473	501	54
250	3383	1992	576	55
2500	3200	2257	572	55
250000	3385	2289	570	55

TABLE 4.6 – Nombre de *Rendez-Vous* entre nos 11 systèmes embarqués et les 8 stations ATMO, en fonction de la fenêtre spatio-temporelle du *Rendez-Vous*.

distance (m) \ durée	1 minute	1 heure	1 jour	toute la période
25	3	3	3	2
250	229	192	87	17
2500	1605	1224	400	59
250000	5397	3652	912	88

Concernant les *Rendez-Vous* avec les stations de référence, six des stations d'ATMO ont été rencontrées par 5 des 11 systèmes au cours de la totalité de l'expérience, 4 seulement si sont considérées celles qui ont été rencontrées plus de 5 fois.

Nous observons deux types d'ensembles de *Rendez-Vous* : ceux répartis de façon homogène autour des stations et ceux répartis uniquement le long d'une route particulière (cf. annexe 8.1).

La Figure 14 de l'annexe 8.1 présente les diagrammes de dispersion pour ces 5 systèmes embarqués pour le NO_2 (le CO n'est pas mesuré par ATMO). Un point correspond à la moyenne de la réponse normalisée du capteur au cours du *Rendez-Vous* (abscisse) et à la mesure horaire en concentration réelle de la station de référence (ordonnée). La forme du point dépend de la station dont provient la mesure horaire. La couleur du point sur le diagramme de dispersion correspond au temps passé ; un point en bleu correspond à un point au début de l'expérience (Septembre) et un point en rouge à un point vers la fin de l'expérience (Mars).

Nous filtrons les points abérrants (9 points soit 5 %) à l'aide de la distance de Cook classiquement utilisée pour former des diagrammes de dispersion. Deux des points filtrés correspondent notamment à des trajets en voiture (au regard du trajet, de la vitesse et de la température moyenne). Les valeurs peuvent être aberrantes pour d'autres raisons, comme par exemple le fait que le vélo suive une voiture.

Les coefficients de détermination R^2 par capteurs sont inférieurs à 0,5, en moyenne, si toutes les stations sont considérées. Toutefois, si chaque station est considérée de façon indépendante, la moyenne des coefficients de détermination est de 0,67.

Le capteur 11 est celui qui a le coefficient de détermination le plus élevé : 0,86. Ses *Rendez-Vous* n'ont lieu qu'avec la station Toulouse-Mazades ; et l'ensemble de ses *Rendez-Vous* sont situés le long d'une route (et non de façon homogène autour de la station).

Le capteur 16 a été en *Rendez-Vous* avec la station Toulouse-Périphérique Traffic. Son nuage de points est relativement dispersé. Toutefois, le *Rendez-Vous* avec cette station s'est effectué dans un lieu particulier : sur une passerelle à vélos à plusieurs mètres au dessus du périphérique.

Ainsi, en reprenant les termes de sensibilité et homogénéité définis au chapitre 2, nos capteurs ont une forte sensibilité avec les stations de référence mais une faible homogénéité. Pourtant, la notion d'homogénéité est plus importante pour l'étalonnage par *Rendez-Vous* (au moins sur la base de données horaires). A défaut d'homogénéité, pour utiliser l'étalonnage par *Rendez-Vous*, notamment pour corriger la dérive, il faudrait étudier la relation qui existe entre le lieu de *Rendez-Vous* visé (déterminé au cours du déploiement) et le lieu de la station de référence à l'aide d'au moins deux instruments. Ceci ne remet nullement en cause l'intérêt de l'étalonnage par *Rendez-Vous*, qui a pour vocation d'être effectué automatiquement tout au long du déploiement du capteur, alors que l'étude de représentativité des lieux entre eux ne se ferait qu'une fois. L'approche *in situ* développée au chapitre 3 esquisse cette étude de représentativité d'un lieu (technopole ou zone rurale) au regard d'un autre (station ATMO) au travers des statistiques observées.

La considération des paramètres météorologiques ne paraît pas primordiale dans la mesure où, avec suffisamment de *Rendez-Vous*, leurs effets semblent se compenser au regard du nuage de points. Nous pensons même que l'influence de ces paramètres pourrait être déterminée grâce à ces *Rendez-Vous*, à l'aide d'un algorithme d'apprentissage automatique supervisé notamment.

Concernant les *Rendez-Vous* entre vélos, ils sont très plus fréquents. La Figure 4.16 en présente deux. Le premier met en relation le capteur 08 et 14. Nous notons que vers la fin de l'expérience (points colorés en rouge) le capteur 14 a une réponse constante quelque soit celle du capteur 08. Nous supposons qu'il ne s'agit pas là d'une dérive mais d'une défaillance du capteur. En effet, lorsque ce capteur nous a été restitué, il semblait avoir été manipulé avec peu de précautions (boue, batterie rayée). Néanmoins, avec le capteur 16, le capteur 08 a un coefficient de détermination de 0,72. Il semble fonctionner correctement.

Au regard du coefficient de determination moyen d'un capteur avec les autres capteurs, il est envisageable de détecter deux cas de défaillance : soit il est très faible (inférieur à 0,2), soit il est extrêmement bon (plus de 0,95). Le premier cas traduit un problème de réversibilité ; le deuxième cas traduit un problème de sensibilité (cf. caractéristiques d'un capteur au chapitre 1).

En conclusion, avec cette fenêtre spatio-temporelle, nous obtenons environ 17,5 *Rendez-Vous* entre vélos et 1,7 *Rendez-Vous* avec les stations ATMO par jour où au moins un vélo est utilisé. Cette méthode semble donc prometteuse pour étalonner les capteurs, même avec un petite flotte de vélos (11 vélos). Toutefois, nous n'avons pas poussé l'analyse au point de pouvoir retrouver les concentrations de pollution à partir des réponses normalisées des capteurs.

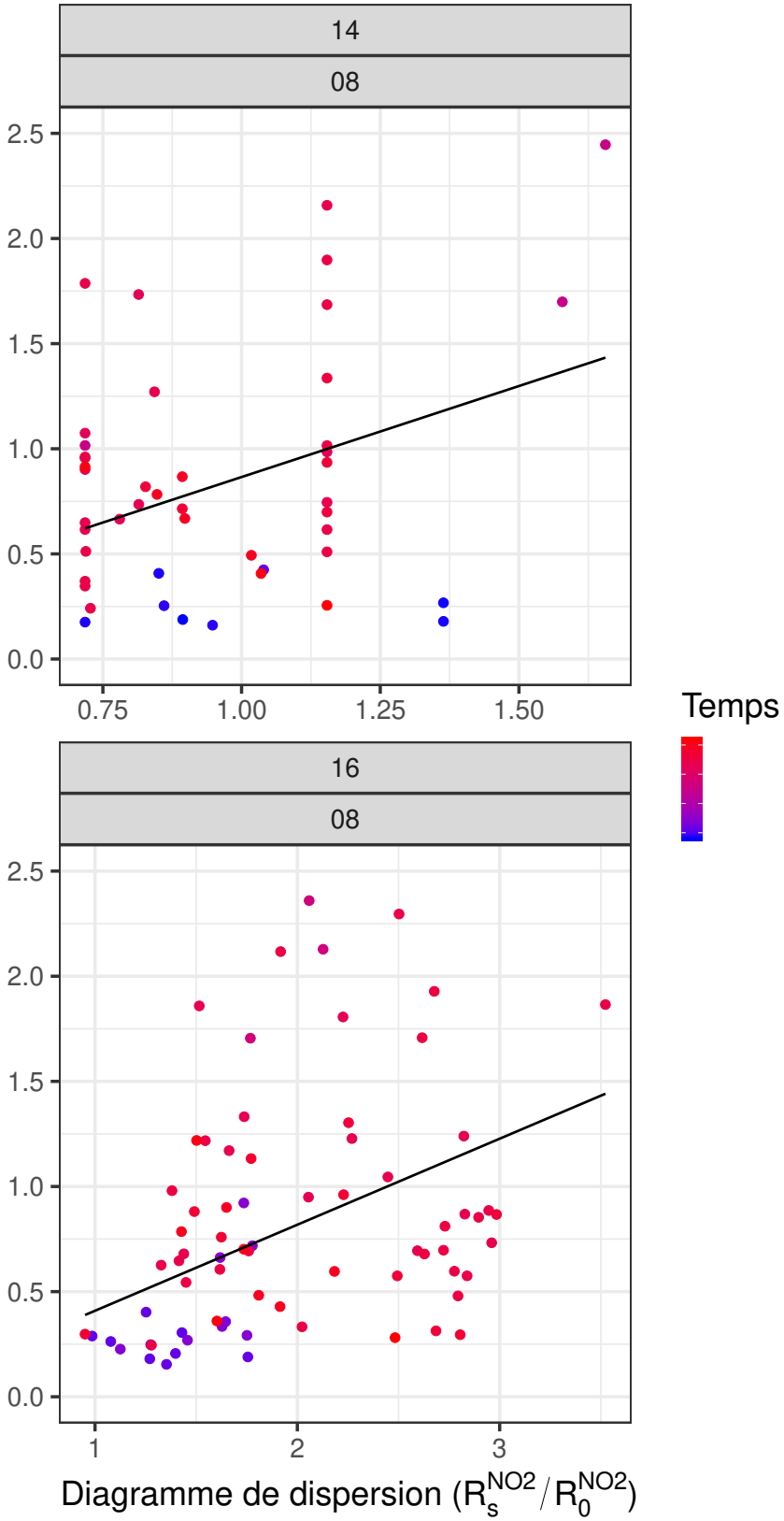


FIGURE 4.16 – Diagramme de dispersion des réponses des capteurs MiCS-4514 (entre capteurs).

4.4.4 Analyse des mesures de polluants sur vélo

Cette section analyse l'ensemble que constitue les trajets à vélos, de façon macroscopique. La Figure 4.19 montre la distribution des réponses normalisées des capteurs.

Pour le NO_2, nous observons une distribution à queue lourde. Lorsque nous ne conservons que la queue de la distribution, nous retrouvons le centre-ville. La distribution présentée est obtenue après avoir filtré les valeurs aberrantes à l'extrémité de la queue. Elles correspondent à des saturations de nos capteurs. Cela suggère que la gamme de valeurs du capteur (entre 0,05 et 10 ppm) n'est pas suffisante pour représenter la gamme de valeurs des concentrations réelles en NO_2 en ville.

Pour le CO, nous observons nettement deux modes. Ils correspondent approximativement à la supperposition de deux gaussiennes ; la première pour les mesures en périphérie de la ville, et la seconde pour les mesures en ville. Ces dernières ont une valeur quatre fois plus importante que celles en périphérie. Ce résultat est supprenant. En effet, selon la fiche technique du capteur, sa réponse est négative en présence de CO. Ceci conduit donc à penser que la concentration en CO en ville est plus faible qu'en périphérie. Toutefois, à la suite de notre étude en laboratoire, nous avions conclu que sa réponse avait dérivé au cours du temps. Ce résultat sur la ville de Toulouse questionne cette conclusion. Au regard du nombre de corrections effectuées sur le système embarqué fourni par Epurtek pour qu'il soit fonctionnel, il est possible qu'il s'agisse à nouveau d'un problème de réalisation. Nous pensons à une inversion du sens du courant dans la mesure de la résistance de l'élément sensible du capteur.

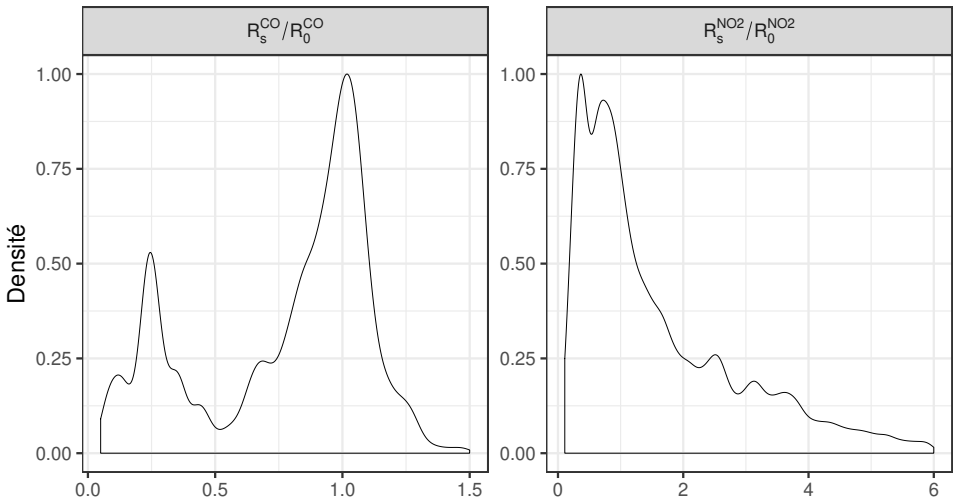


FIGURE 4.17 – Densité des mesures en CO et en NO_2.

Ainsi, en prenant les valeurs moyennes en CO et maximales en NO_2, il est possible de déterminer des niveaux de pollution dans la zone d'étude. La Figure 4.18 présente une rasterisation de toutes nos mesures sur la base de ces métriques. Les axes principaux ainsi que le centre-ville se démarquent.

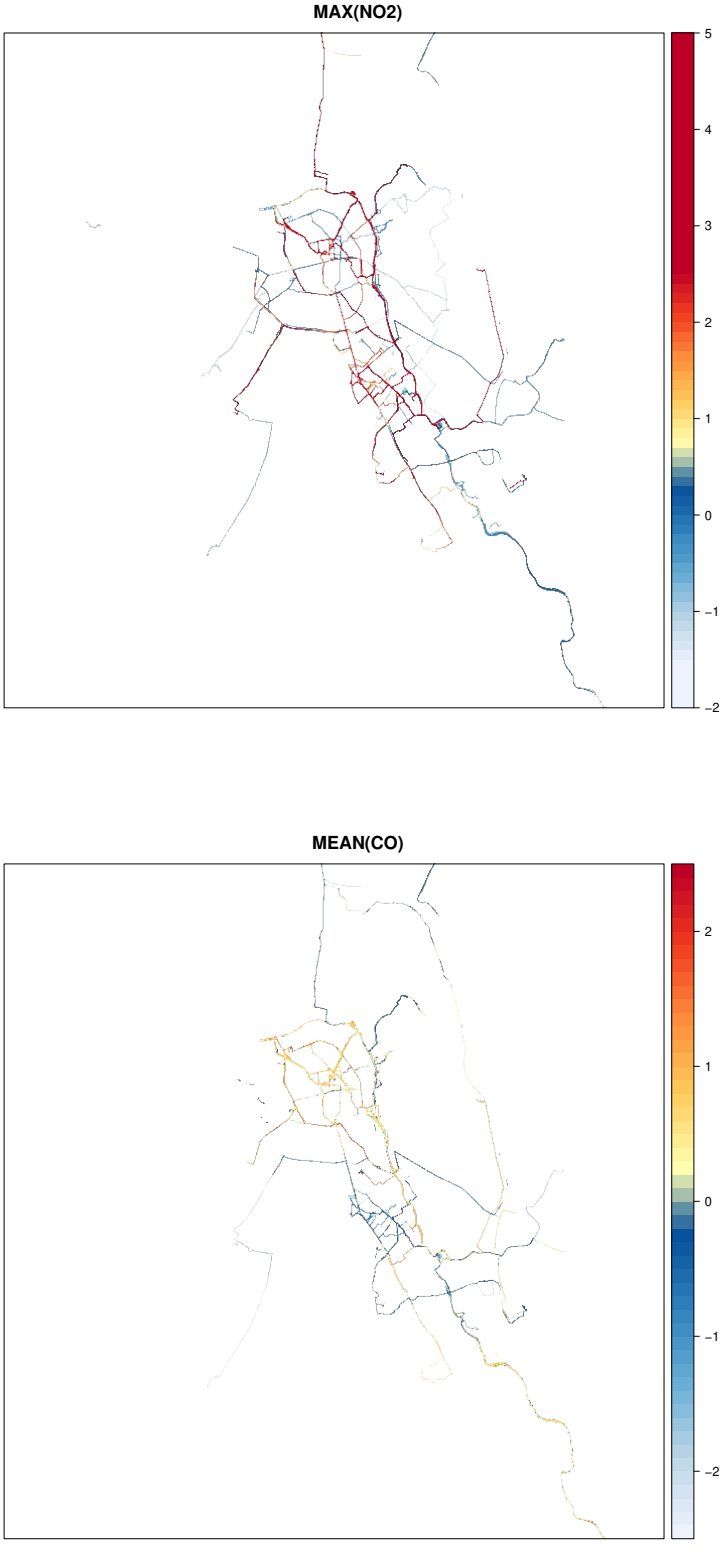


FIGURE 4.18 – Rasters représentant les valeurs maximales en NO_2 et moyennes en CO de nos mesures. Résolution de 25 m × 25 m.

Pour analyser plus finement cette observation, nous générons les mêmes variables explicatives sur la ville de Toulouse que sur la ville de Marseille, avec la méthodologie présentée au chapitre 2. Elle sont tracées en annexe 8.2. Nous constatons que la ville de Toulouse est mieux décrite, notamment en terme d'occupation des sols (landuse).

En ne conservant que les catégories suffisamment échantillonnées par les vélos, nous traçons la distribution des mesures en NO_2 et en CO en fonction de la variable catégorielle intitulée network (cf. Figure 4.19).

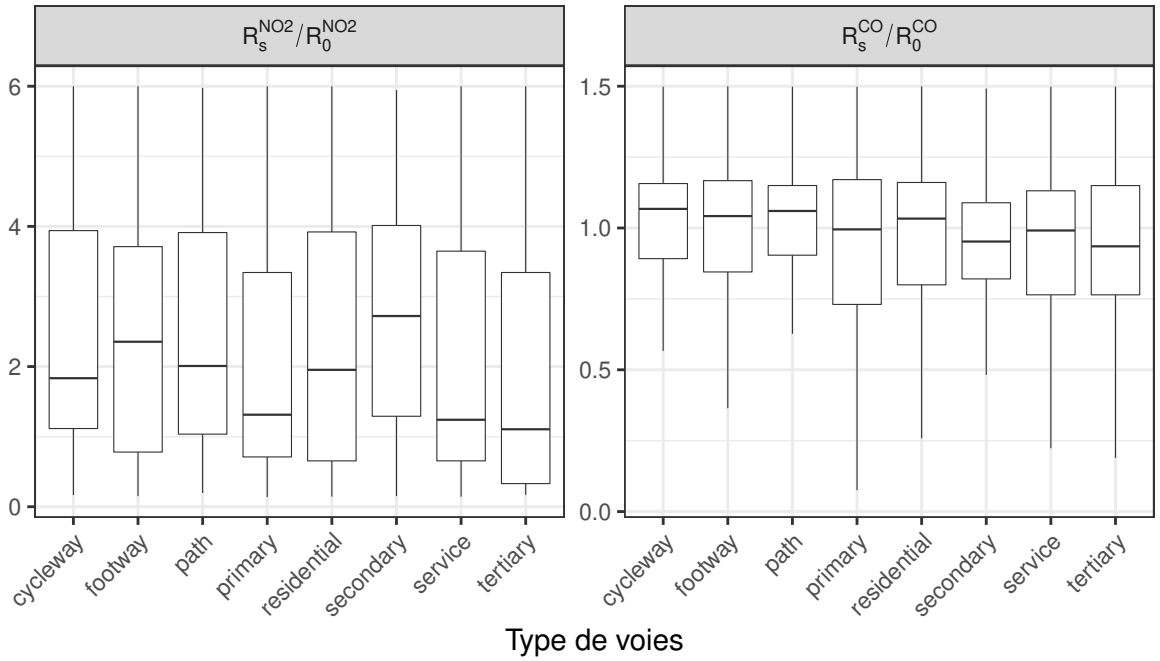


FIGURE 4.19 – Densité des mesures en CO et en NO_2 en fonction des catégories de la variable network.

Pour le NO_2, nous observons que les voies piétonnes et les voies secondaires sont les plus polluées. Les voies piétonnes sont majoritairement dans le centre de la ville et les voies secondaires sont les artères principales de la ville. Ceci est donc cohérent. Ensuite apparaissent les pistes cyclables et les voies de type résidentiel, qui elles sont autant représentées en ville qu'en périphérie. Enfin, les mesures sont les plus faibles pour les voies de type *service* et *tertiary*, davantage représentées en milieu rural. Une catégorie se dénote : les voies primaires. Effectivement le niveau de pollution est similaire à celui en milieu rural. Après analyse de ce type de voies, il s'avère que la ville de Toulouse en comporte très peu : les autouroutes, non échantillonnées après filtrage des trajets en voiture, et le boulevard Carnot. Ce dernier traverse la ville. Il a la particularité d'être plus large que n'importe quelle autre voie de la ville, et de disposer de voies de bus spécifiques que se doivent d'emprunter les vélos. Nous supposons que le niveau de pollution observé n'est pas représentatif car les vélos sont plus éloignés des voitures qu'à la normale. Or nous avons vu une nette décroissance de la concentration du NO_2 au regard de la distance à la route (cf. section précédente). Il est important de préciser que dans la ville de Toulouse, très peu de « pistes cyclables » en sont réellement. Les voies définies comme telles par la mairie sont généralement des voies à sens unique que les vélos peuvent emprunter à

contre sens. Cela augmente fictivement le nombre de km de pistes cyclables réels et biaise la représentativité de nos données pour ces voies.

Pour le CO, le type de voies semble moins impacter la mesure. Ceci est cohérent au regard de son échelle de variabilité (cf. chapitre 1) ; il s'agit d'un polluant de fond, avec des variations locales principalement dues à la pollution automobile. Bien que le vélo soit sur une route, s'il ne suit pas une voiture, la concentration est globalement la même. La rasterisation précédente des mesures en CO (Figure 4.18) suggère en effet que l'influence est davantage marquée par la proximité au cœur de la ville qu'au type de route.

4.4.5 Spatialisation des mesures du réseau de capteurs

En reprenant la méthodologie définie au chapitre 2, nous estimons des cartes modélisant l'état de la pollution de l'air en NO_2 et en CO dans la ville de Toulouse à l'aide du Krigeage, d'un LUR (GAM) et d'un réseau de neurones (cf. Figure 4.21 et 4.20).

Ces cartes ne comportent volontairement pas d'échelle car elles sont produites à partir des réponses normalisées et ne reflètent pas les concentrations réelles en polluants. Effectivement, la fiche technique des capteurs annonce une relation logarithmique entre la réponse normalisée et la concentration. En reprenant la courbe d'étalonnage annoncée par le constructeur, la gamme de valeurs de nos cartes n'est pas réaliste. Toutefois, ces cartes permettent de déterminer les lieux les plus pollués de Toulouse. Cela peut être utile pour le déploiement ultérieur de stations fixes ou pour retrouver les sources éventuelles de pollution.

Afin de réduire l'impact du cycle journalier des polluants, nous nous focalisons sur les mesures effectuées entre 16h et 18h (et dans l'idée de « *Rendez-Vous* par évènement réccurent »). Cela permet d'avoir des données qui représentent la même information partout dans la ville, et d'observer l'état de la pollution au moments où les différences sont les plus flagrantes (heures des pics de pollution), tout en préservant un nombre important de mesures (219 745). Cela nous rapproche de notre étude sur la ville de Marseille où les simulations ont été effectuées sur la base de maxima journaliers.

En définitive, ces cartes permettent d'observer la différence de variabilité des deux polluants. Pour le CO, elles sont relativement homogènes. Des zones se distinguent, d'une façon particulièrment prononcée avec le Krigeage. Le cœur de Toulouse, à l'intérieur du périphérique, ainsi que la ville de Ramonville au Sud et les voies rapides ont un niveau de pollution semblable. Pour la Ramonville, il peut soit s'agir d'un effet de bord, soit d'une réelle démarquation du niveau de pollution en CO en fonction du milieu urbain ou rural.

Le LUR et le réseau de neurones prédisent des taux de pollution comparables entre la ville et la Garonne et un niveau très élevé au niveau des ponts qui la traversent. A l'inverse, la gare a un taux de pollution faible relativement à sa localité.

Ne pouvant pas pleinement interpréter ces niveaux de pollution en CO, nous ne pousseront pas l'analyse.

Pour le NO_2 cependant, à partir de l'estimation du Krigeage, des zones plus localement réduites se distinguent, aux particularités bien connues des toulousains. D'abord, une zone industrielle et un technopole apparaissent nettement : Montaudran, dont au centre du point chaud, les parkings des entreprises, et Labège, à la sortie de la voie rapide et à proximité du péage de l'autoroute.

De plus, les grands boulevards menant à Toulouse (notamment boulevard Silvio Trentin et la route de Narbonne) sont également marqués.

Pech David, point culminant de la ville, est un autre point chaud. Cependant, cette méthode ne se base pas sur des données décrivant la ville pour établir sa prédiction. La précision avec laquelle Pech David se distingue est donc très troublante. De plus, seul le Krigeage retrouve ce point chaud.

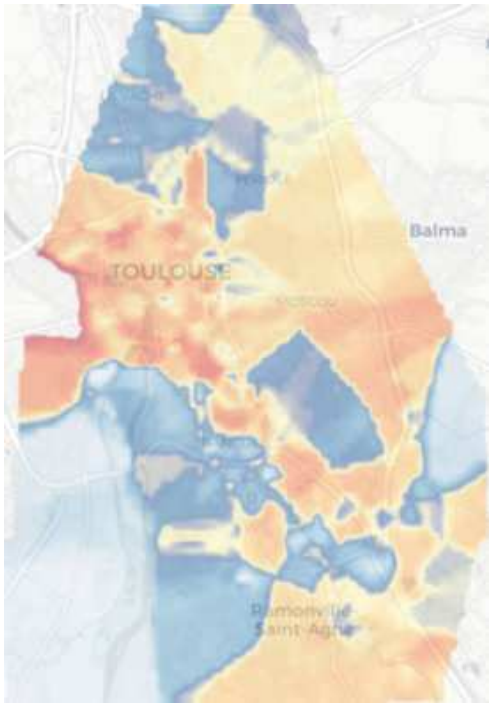
Enfin, certains points d'intérêt (au sens de la variable explicative susnommée), dont particulièrement la gare, sont également des points chauds.

Cependant, nous observons à l'Ouest une zone où les valeurs prédites sont quasi nulles. Suite à notre simulation sur la ville de Marseille et au regard des trajets effectués ne couvrant pas cette zone, nous pouvons affirmer que cela ne représente pas la concentration réelle mais une absence de prédiction.

A partir de l'estimation du LUR, nous retrouvons une forte relation de la concentration en polluant avec la distance à la route. En conséquence, le cœur de la ville semble moins pollué car il possède moins d'axes routiers principaux. Là encore, nous retrouvons des points chauds au niveau des artères principales menant à la ville, dont la route de Narbonne, et au niveau de la gare. Des quartiers connus ont également une concentration de pollution plus élevée : Jean-Jaurès (en plein centre), Fondeyre (proche des péages sur l'autoroute en direction de Bordeaux) et les Izards. L'interprétation de cette carte est à nuancer. Grâce à notre simulation sur la ville de Marseille, nous pensons reconnaître une carte qui souffre d'un manque de représentativité des zones à l'Est et à l'Ouest. Ceci n'est pas surprenant car les trajets traversent majoritairement la ville du Nord vers le Sud. Toutefois, les artères principales traversent la ville du Nord au Sud, il n'est donc pas surprenant que ces zones soient plus polluées. Les prédictions pour les zones à l'Est et à l'Ouest sont donc à prendre avec précaution.

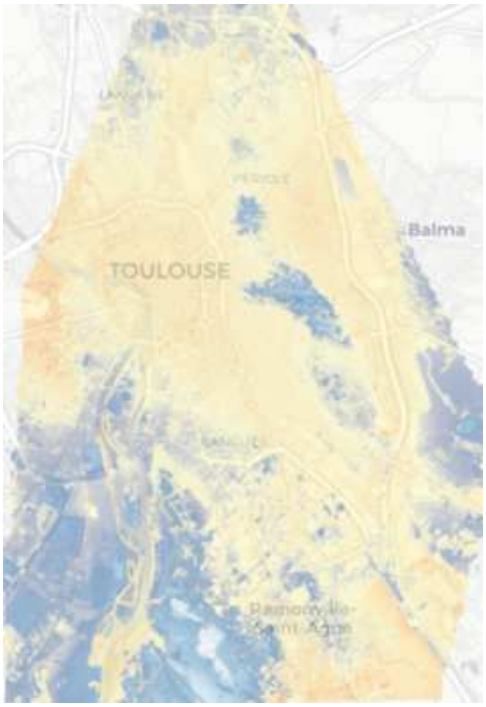
A partir de l'estimation du réseau de neurones, les voies rapides se dénotent, ainsi que le cœur de la ville, et les quartiers populaires Arènes, Bagatelle, Sept Deniers et les Izards. Elle semble fournir un compromis entre tendance de fond et variabilité locale (dont l'impact local des routes), et permet de retrouver une bonne partie des points chauds déterminés par le Krigeage, mais avec une intensité plus faible, et le LUR. Néanmoins, Pech David ne ressort pas.

En conclusion de ces prédictions pour le NO_2, toutes les méthodes s'accordent à dire que les Izards est un quartier pollué. Les points chauds se retrouvent de carte en carte, mais avec des intensités variables. De plus, ces points chauds sont des lieux caractéristiques de la ville.



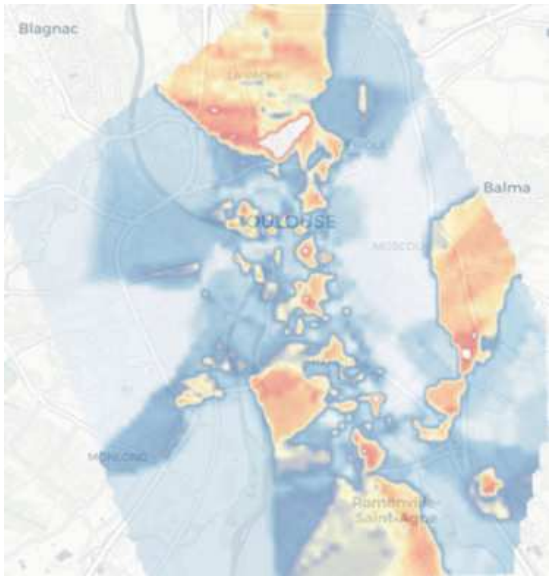
(a) Krigeage

(b) LUR

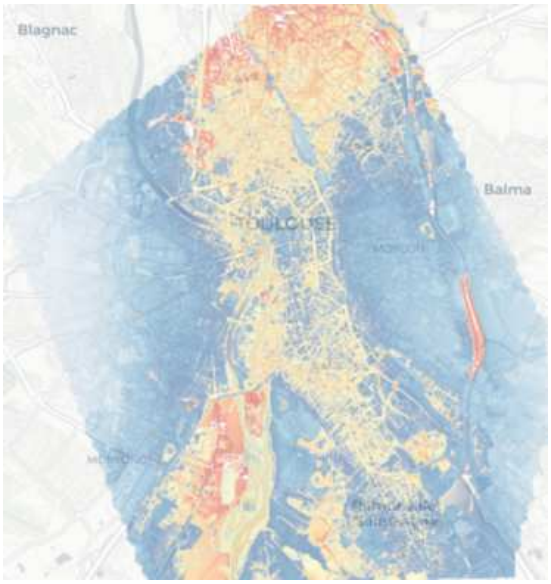


(c) Neural Net.

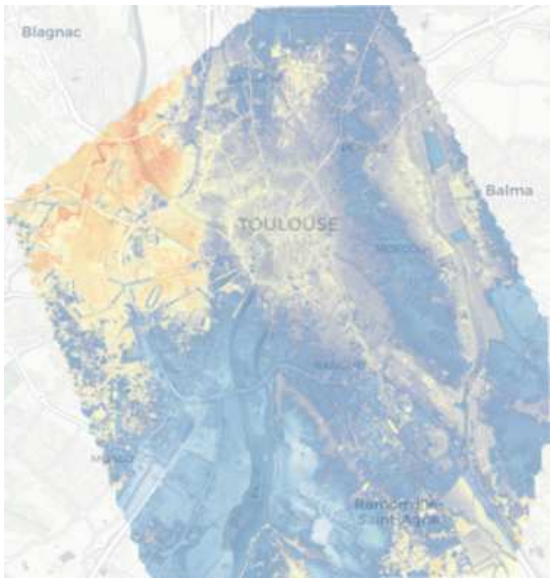
FIGURE 4.20 – Cartes de pollution en CO pour la ville de Toulouse, estimées (a) par Krigeage, (b) par LUR et (c) par réseau de neurones sur la base de nos mesures réelles. Résolution de 25 m × 25 m. Les prédictions sont centrées-réduites.



(a) Krigeage



(b) LUR



(c) Neural Net.

FIGURE 4.21 – Cartes de pollution en NO_2 pour la ville de Toulouse, estimées (a) par Krigeage, (b) par LUR et (c) par réseau de neurones sur la base de nos mesures réelles. Résolution de 25 m × 25 m. Les prédictions centrées sont normalisées par la valeur maximale.

4.5 Conclusion

Dans ce chapitre, nous avons exposé la mise en œuvre concrète d'un réseau de capteurs sur vélo pour estimer les niveaux de pollution dans la ville.

Pour cela, nous avons déployés l'instrument de mesure (dont la conception a été présentée au chapitre 3) auprès d'une association de location de vélos et auprès de vélo-taffeurs de notre laboratoire. Le premier déploiement a été un échec en raison du trop haut niveau

d'automatisation requis dans le temps imparti (3 ans pour une personne seule).

Le second déploiement a fourni de biens meilleurs résultats. En effet, le fait de nous rediriger vers une base d'utilisateurs scientifiques, soucieux du protocole à suivre, nous a permis d'abandonner l'autonomie en énergie du capteur via la dynamo au profit d'une batterie à charger.

A la suite de ce déploiement, nous avons collecté un jeu de donnée de déplacements et de mesures en pollution de l'air (NO_2 et CO) sur une période de 7 mois, avec 11 systèmes embarqués utilisés par 17 personnes. Afin d'analyser ces données, nous avons filtré les positions indésirables (initialisation du GPS, trajets en voiture) et reconstitué les trajets à vélo. Ces déplacements nous ont permis de valider le réalisme du modèle de cycliste construit au chapitre 2 au regard de l'utilisation que nous en faisons. Ils nous ont également permis de constater que les heures de ces déplacements correspondent aux heures des pics de pollution (d'origine automobile).

Puis, nous avons étudié la pollution de l'air dans la ville de Toulouse à l'aide de ces mesures. Tout au long de cette thèse, nous nous sommes intéressés à la notion de *Rendez-Vous*. Dans cette dernière analyse à partir de données réelles, nous sommes partis d'un cas idéal de *Rendez-Vous* (deux vélos qui roulent côte à côte) pour arriver à un cas réaliste et généralisable.

Par la suite, nous avons discuté de l'application d'un étalonnage par *Rendez-Vous* avec les stations de référence et entre capteurs sur vélo. Sur la base des *Rendez-Vous* avec les stations fixes **et** de référence, nous en avons conclu que la relation qui peut être établie est dépendante du lieu, et notamment de l'homogénéité de la zone autour de la station. Ainsi, s'il semble possible d'étalonner nos capteurs à l'aide des stations de fond, mais ce n'est pas le cas avec la station surveillant le périphérique.

De surcroît, nous nous sommes intéressés aux mesures dans leur ensemble. Cela nous a permis de déterminer deux niveaux moyens de pollution pour le CO, en ville et en zone rurale, et de retrouver l'échelle de variabilité théorique. Pour le NO_2, nous avons observé une distribution à queue lourde, signe que notre capteur sature en milieu urbain. En conséquence, les maxima de pollution en NO_2 dans la ville de Toulouse dépassent localement 10 ppm (environ $20\,mg/m^3$) soit 100 fois le seuil en moyenne horaire recommandé. Il s'agit probablement de cas où le vélo suit une voiture.

Enfin, bien que le nombre de trajets journaliers collectés soit inférieur au nombre nécessaire estimé par notre simulation numérique afin d'établir des cartes quotidiennes de la pollution de l'air, nous avons spatialisé les mesures correspondant aux heures des pics de pollution (entre 16h et 18h) durant toute la période de l'expérience, pour déterminer les points chauds de Toulouse. Nous avons retrouvé des lieux caractéristiques de la ville, ce qui met en évidence la vraisemblance de nos prédictions. De plus, au travers de ce cas réel, nous avons à nouveau constaté les forces et faiblesses des méthodes d'estimation étudiées (Krigeage, LUR et réseau de neurones) au chapitre 2. Nous concluons finalement que l'évaluation de concentrations réalistes et des erreurs associées restent encore inaccessibles avec ce type de capteur.

Bibliographie

Althoff, T. , White, R. W. et Horvitz, E. . Influence of Pokémon Go on Physical Activity : Study and Implications. J Med Internet Res, 2016. DOI : 10.2196/jmir.6759.

Belkhiri, A. , Bechkit, W. et Rivano, H. . Virtual Forces based UAV Fleet Mobility Models for Air Pollution Monitoring. 2018 IEEE 43rd Conference on Local Computer Networks (LCN). IEEE, 2018. DOI : 10.1109/LCN.2018.8638231.

Cook, M. . It Takes Two Neurons To Ride a Bicycle. Neural Information Processing Systems, 2004. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.88.3781.

He, J. , Zhao, M. et Stasinopoulos, S. . Constant-velocity steering control design for unmanned bicycles. 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO). IEEE, 2015. DOI : 10.1109/ROBIO.2015.7418805.

Kosmidis, E. , Syropoulou, P. , Tekes, S. , Schneider, P. , Spyromitros-Xioufis, E. , Riga, M. , Charitidis, P. , Moumtzidou, A. , Papadopoulos, S. , Vrochidis, S. , Kompatsiaris, I. , Stavrakas, I. , Hloupis, G. , Loukidis, A. , Kourtidis, K. , Georgoulias, A. et Alexandri, G. . hackAIR : Towards Raising Awareness about Air Quality in Europe by Developing a Collective Online Platform. ISPRS International Journal of Geo-Information, 2018. DOI : 10.3390/ijgi7050187.

LaMarca, A. , Brunette, W. , Koizumi, D. , Lease, M. , Sigurdsson, S. B. , Sikorski, K. , Fox, D. et Borriello, G. . Making Sensor Networks Practical with Robots. Pervasive Computing. Springer Berlin Heidelberg, 2002. DOI : 10.1007/3-540-45866-2_13.

Conclusion et perspectives

Voir est un acte.

– René Magritte

La ville de demain sera intelligente et connectée. Elle visera à un développement harmonieux avec son environnement naturel, en respectant des attentes environnementales élevées, et particulièrement en ce qui concerne l'air respiré. La problématique de la qualité de l'air en zone urbaine est complexe et connexe aux questions de développement, d'urbanisme, de mobilité et de comportements. L'utilisation stratégique des infrastructures, des services de l'information et de la communication peut permettre une gestion intelligente de ces questions et d'aboutir à une qualité de l'air plus saine.

La question abordée dans cette thèse traite de l'utilisation de systèmes répartis et mobiles pour acquérir une meilleure compréhension de la qualité de l'air à des échelles spatiales et temporelles qui ne sont pas accessibles à l'aide de systèmes existant à l'heure actuelle. La démarche retenue décompose cette question en trois sous-problèmes plus simples et relativement disjoints : (i) l'acquisition de trajets d'utilisateurs, (ii) la mesure de l'information le long de ces trajets et (iii) la spatialisation de l'information dans la zone étudiée.

La première contribution est l'utilisation de cette décomposition du problème pour proposer une méthode théorique de détermination de deux paramètres du réseau de capteurs mobiles, sa taille et son pas d'échantillonnage de mesure. Cette méthode repose sur la génération de jeux de mesures à partir d'une carte de pollution de référence et sur la comparaison de la spatialisation des jeux de mesures à la carte de référence. Dans le cas d'un jeu de données idéal pour une méthode de spatialisation, la carte estimée est identique à la carte de référence.

Nous avons étudié trois méthodes de spatialisation (Krigeage, Land-Use Regression et réseau de neurones). Nous avons fait varier successivement les deux paramètres identifiés pour la génération des jeux de données et nous avons observé l'erreur entre la carte estimée et la carte de référence à l'aide de trois métriques : le coefficient de corrélation, la RMSE et la MAE.

Nous avons appliqué cette méthode pour deux polluants, le NO_2 et les PM_{10}. Les émissions de ces 2 polluants sont principalement primaires et reliées au trafic automobile, mais leur temps de résidence est différent. L'expérience portait sur 14 jours consécutifs traités de façon indépendante pour la ville de Marseille. Nous en avons déduit qu'un réseau de l'ordre de quelques dizaines de capteurs mobiles dont les mesures sont échantillonnées tous les 200 m est satisfaisant pour la modélisation de la pollution de l'air. Par la suite, nous avons étudié les principales sources d'erreur de spatialisation au regard de variables explicatives décrivant la ville. Nous en avons conclu que la zone de l'espace est davantage source d'erreurs que la méthode de spatialisation ou que le polluant. Effectivement, les lieux mal prédits sont généralement les mêmes : soit des lieux où les vélos ne vont pas (forêt, routes privées, réserve naturelle, zone industrielle, mer), soit des lieux à forte variabilité (réseau routier, routes et zones résidentielles). Les lieux les mieux prédits sont des lieux particuliers liés au trafic (station essence, station de taxi, parking, rond-pond,

parking à vélo, stop, métro) et les pistes cyclables. Enfin, nous avons étudié les erreurs d'estimation des méthodes de spatialisation en comparant les prédictions avant et après introduction d'une perturbation spatiale sur la carte de référence. Les trois méthodes sont robustes à l'introduction d'un bruit blanc, mais seul le Krigeage permet de retrouver correctement une perturbation sphérique (type explosion). La reconstitution d'une perturbation linéique (sur une route) est la plus difficile à retrouver.

La seconde contribution est le retour d'expérience sur le prototypage de l'instrument de mesure et sur les stratégies de déploiements. Nous avons conçu un système de mesure dédié au vélo et quasi-autonome vis-à-vis de l'utilisateur.

Il embarque un micro-capteur low-cost MOx de NO_2 et CO, le MiCS-4514, un multi-capteur des conditions météorologiques (température, pression, humidité), le BM280, un accéléromètre et un récepteur GPS.

La technologie MOx est peu sélective et très sensible aux conditions de pression, température et humidité, ce qui rend l'évaluation en laboratoire assez peu concluante. *In situ*, nous avons disposé 4 systèmes embarqués en deux lieux de la région toulousaine (technopole, zone rurale) pendant plusieurs jours. La corrélation avec les paramètres météorologiques est importante et il semble possible de séparer l'influence de ces paramètres de manière statistique. Les réponses normalisées des différents capteurs sont très corrélées entre elles (coefficient de 0,95 pour le NO_2 et pour le CO). Par ailleurs, nous avons retrouvé des statistiques globales comparables à celles fournies par les stations de référence d'ATMO Occitanie. Ceci montre que les capteurs ont un comportement assez similaire et permet d'envisager un suivi de l'étalonnage collaboratif par *Rendez-Vous* entre les systèmes eux-mêmes et lors de passage à proximité de stations de référence.

De nombreuses difficultés sont apparues dans cette étape de réalisation et ont nécessité un travail beaucoup plus long qu'initialement prévu. En l'état actuel, le système ne donne pas entièrement satisfaction par rapport à la qualité des observations fournies en raison d'erreurs de conception (typiquement la position des différents capteurs sur la carte, ou la communication en temps réel) qui n'ont pas pu être résolues dans le cadre de ce travail. Néanmoins, deux déploiements en conditions réelles ont pu être réalisés.

Dans un premier temps, ces systèmes ont été déployés sur les vélos d'une association de location. Ce premier déploiement n'a pas été concluant. En effet, les systèmes étaient alimentés à l'aide de la dynamo du vélo pour minimiser l'intervention des utilisateurs. Dans ces conditions, le temps de synchronisation du récepteur GPS, supérieur à la durée moyenne d'un trajet, ne permet pas de collecter des données. De plus, nous avons observé un effet canyon, sous les allées de platanes le long du canal du Midi et par temps très nuageux, qui empêche la synchronisation GPS. L'échec de cette expérience montre également les limites d'un modèle d'observation totalement transparent d'un point de vue de l'utilisateur.

Dans un deuxième temps, nous avons déployé le système auprès de 18 vélo-taffeurs de notre laboratoire en adjoignant une batterie complémentaire au système. Cette batterie permet à l'utilisateur d'allumer le système 30 minutes avant le départ pour la synchronisation GPS et le pré-chauffage du capteur MOx. Cela nous a permis de simplifier grandement notre système et ainsi de collecter une base significative de données de déplacements et de mesures en NO_2 et en CO dans la ville de Toulouse et ses alentours.

La troisième contribution de cette thèse concerne la construction et l'analyse de ce jeu de données. L'utilisation de vélos comme vecteur du capteur, contrairement à l'utilisation de transports en commun par exemple, fait rentrer en ligne de compte des éléments relevant du comportement individuel. La durée, la vitesse, le nombre d'arrêts, l'itinéraire suivi sont variables et quasi aléatoires. La reconstruction des trajets à partir du positionnement GPS s'est donc avérée assez complexe. A partir des données GPS ponctuelles, nous avons reconstruit 599 trajets à vélo pour 11 systèmes embarqués actifs sur une période de 7 mois.

Les systèmes ont fonctionné en moyenne un jour sur deux, sans traiter de façon particulière les week-ends et les vacances. De plus, lors d'un jour d'utilisation des systèmes, environ un quart de la flotte était active. Les trajets ont duré environ 20 minutes. La vitesse moyenne des vélos est de 16,6 km/h sur une distance de 5,7 km. Sans surprise, les pistes cyclables et les axes secondaires de circulation routière sont sur-représentés. Il apparaît également qu'une étape supplémentaire impliquant un map-matching permettrait d'améliorer le positionnement de certains trajets. Enfin, nous avons dénombré 1992 *Rendez-Vous* entre les vélos et 192 *Rendez-Vous* avec les stations de référence d'ATMO Occitanie, pour un *Rendez-Vous* dans une fenêtre de 1 h à une distance de moins de 250 mètres. Les mesures en polluants de ce jeu de données semblent exploitables puisqu'elles ont permis de facilement retrouver une forte variabilité à l'echelle de la rue, entre la route et les trottoirs ; simplement au regard des variations des mesures en NO_2.

A partir de ce jeu de données, nous avons estimé les niveaux de pollution en CO et NO_2 pour la ville de Toulouse. Pour cela, nous nous sommes appuyés sur les conclusions du chapitre 3 et nous avons rasterisé toutes les mesures de la période de l'expérimentation (7 mois) correspondant aux pics de pollution de fin d'après-midi (entre 16h et 18h). Puis, nous avons appliqué les méthodes de spatialisation étudiées au chapitre 2. Les prédictions (NO_2 et CO) ne reflètent pas les concentrations réelles mais leurs variations. Nous en déduisons que les points chauds en NO_2 de Toulouse sont le cœur de la ville et certains quartiers populaires, dont les Izards, ses axes principaux (route de Narbonne, route d'Espagne, boulevard Silvio Trentin), Montaudran (zone industrielle), Labège (technopole) et la gare. De plus, l'echelle théorique de variabilité s'observe dans nos mesures et dans les cartes estimées.

Enfin une dernière contribution concerne la mise à disposition des scripts développés pour chacun de ces sous-problèmes précédemment énoncés et pour le traitement de leur enchaînement (cf. annexe 9). Il n'est cependant pas concevable de liberer l'accès à la donnée brute telle que nous l'avons collectée durant ce travail. Il reste à évaluer si les produits délivrés via le traitement des données (les niveaux de polluants spatialisés notamment) ne présentent pas de faille vis-à-vis de la protection de la vie privée du groupe de personnes ayant acquis ces données. Le Krigeage, qui est un estimateur qui minimise la variance spatiale du résidu d'estimation, conserve approximativement les extrema observés par les vélos. De plus, les différences entre les estimations théoriques des niveaux de polluants et celles évaluées en utilisant les trajets sont susceptibles de délivrer des informations sur les trajets utilisés et donc sur les individus.

Cette thèse, de la conception de l'instrument de mesure à la modélisation, souligne l'importance de la collecte de données dans le processus de modélisation. A la différence de

nombreuses études en sciences des données, nous avons été producteur et consommateur de la donnée ce qui nous a permis d'effectuer des va-et-vient entre ces deux rôles.

Les perspectives de ce travail sont nombreuses. L'étude du guidage des vélos afin de sélectionner les informations à collecter nous semble primordiale. Ce guidage peut-être programmé (robots), suggéré (ludification) ou énoncé (expérimentateurs). En effet, être acteur de la collecte de l'information permet :

— de garantir des *Rendez-Vous* utiles à l'étalonnage automatique des capteurs,
— d'assurer que toutes les variables explicatives des mesures du modèle sont échantillonnées, voire de maximiser une fonction d'information (au sens de la théorie de l'information) et minimiser le nombre de vélos nécessaires,
— d'explorer la zone d'étude pour rechercher des zones particulières – telles que les points chauds – ou déterminer l'emplacement de stations fixes, à l'aide d'une descente de gradient « dans la vraie vie » que le cycliste suit à l'aide de son smartphone par exemple,
— de confirmer/infirmer la modélisation, notamment pour les zones jamais explorées ou prédites avec confiance,
— de visiter les zones prédites avec peu de confiance, pour améliorer les modélisations successives.

Autrement dit, cela permet de former un jeu de données plus réduit, donc moins coûteux en terme d'acquisition (temps, budget, participants), mais qui possède des propriétés intéressantes au regard de la modélisation qui en découle.

De plus, nous pensons que cette question des *small data*, entre les campagnes de collecte de données et les *big data*, deviendra centrale dans le futur, à cause de l'impact écologique des ces dernières. En effet, le contrôle intelligent de l'acquisition de la donnée permettrait une approche plus parcimonieuse et économe en terme de traitement de l'information, en accord avec les principes de l'innovation frugale. Pour cela, nous pouvons concevoir un nouveau cycle de création de connaissances pour y intégrer pleinement la phase de collecte de données, sur la base du cycle look–compute–move et de la notion de temporalité de ces phases (passé, présent, futur).

Cette thèse, et son déroulement au sein de l'équipe Tolérance aux fautes et Sûreté de Fonctionnement côté LAAS-CNRS, conduit également à une réflexion plus large sur la vie privée dans les smart city, et plus particulièrement les safe city, thème qui n'a pas été abordé dans ce travail. Ce sont des smart city particulièrement dirigées vers la sécurité. Qui peut dire être contre la *safety* ? Personne. Mais la safety de qui (citoyens) ? de quoi (institution, infrastructure) ? Puisque la perception est forcément subjective (cf. préambule), le fait observé dépend de l'observateur. Il paraît alors illusoire de prétendre que la recherche peut être agnostique quant aux applications et il paraît nécessaire d'être vigilant aux détournements possibles, notamment en terme de surveillance de la ville et de ses habitants.

En guise de mot de fin, voici une explication des citations figurant en en-tête : la modélisation n'est pas la réalité mais une représentation (« Le dessin n'est pas la forme, il est la manière de voir la forme »). Donc pour modéliser, il est préférable d'avoir plusieurs sources d'informations pour les confronter (« Revois deux fois pour voir juste, ne vois qu'une fois pour voir beau »). Or si ces sources d'informations n'ont pas la même réponse

pour une même information, c'est qu'il faut opérer des transformations pour accorder leurs réponses (« Les choses ne changent pas. Change ta façon de les voir, cela suffit »). En outre, c'est grâce aux diverses sources d'informations qu'il est possible de déceler celles qui sont erronées (« Ce n'est qu'avec les yeux des autres que l'on peut bien voir ses défauts ») et ainsi améliorer le réalisme de la modélisation. Enfin, une nouvelle perspective d'amélioration du réalisme de la modélisation est de diriger la collecte d'informations pour confirmer/infirmer celle-ci (« Voir est un acte »).

Annexes

1 Anekãntavãda

L'essence de l'Anekãntavãda, doctrine non-absolutiste du jaïnisme, est synthétisée par la parabole suivante (Andhgajanyãyah), traduite par le poète John Godfrey Saxe en 1872, puis remise au goût du jour sous forme de caricatures. L'une d'elle est présentée Figure 1.

It was six men of Indostan
To learning much inclined,
Who went to see the Elephant
(Though all of them were blind),
That each by observation
Might satisfy his mind.

The First approached the Elephant,
And happening to fall
Against his broad and sturdy side,
At once began to bawl:
God bless me!—but the Elephant
Is very like a wall!"

The Second, feeling of the tusk,
Cried: "Ho!—what have we here
So very round and smooth and sharp?
To me 't is mighty clear
This wonder of an Elephant
Is very like a spear!"

The Third approached the animal,
And happening to take
The squirming trunk within his hands,
Thus boldly up and spake:
"I see," quoth he, "the Elephant
Is very like a snake!"

The Fourth reached out his eager hand,
And felt about the knee.
"What most this wondrous beast is like
Is mighty plain," quoth he;
"'T is clear enough the Elephant
Is very like a tree!"

The Fifth, who chanced to touch the ear,
Said: "E'en the blindest man

Can tell what this resembles most;
Deny the fact who can,
This marvel of an Elephant
Is very like a fan!"

The Sixth no sooner had begun
About the beast to grope,
Than, seizing on the swinging tail
That fell within his scope,
"I see," quoth he, "the Elephant
Is very like a rope!"

And so these men of Indostan
Disputed loud and long,
Each in his own opinion
Exceeding stiff and strong,
Though each was partly in the right,
And all were in the wrong!

So, oft in theologic wars
The disputants, I ween,
Rail on in utter ignorance
Of what each other mean,
And prate about an Elephant
Not one of them has seen!

– John Godfrey Saxe, *The Blind Men And The Elephant.*

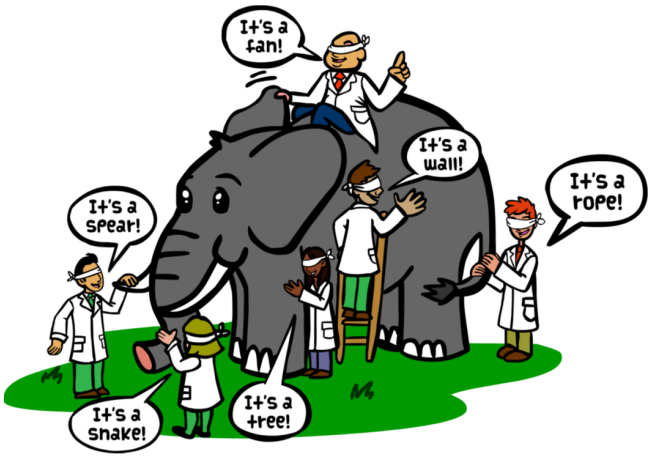


FIGURE 1 – Caricature de l'Andhgajanyãyah par Gnazmul, créateur de contenu sur ImgBin. Licence : usage non-commercial.

2 Rapport d'expérience : exploration de fichiers de journalisation à l'aide du traitement en langage naturel et application à la détection d'anomalies

Experience Report: Log Mining using Natural Language Processing and Application to Anomaly Detection

Christophe Bertero, Matthieu Roy, Carla Sauvanaud and Gilles Tredan
LAAS-CNRS, Université de Toulouse, CNRS, INSA, Toulouse, France
Email: firstname.name@laas.fr

***Abstract*—Event logging is a key source of information on a system state. Reading logs provides insights on its activity, assess its correct state and allows to diagnose problems. However, reading does not scale: with the number of machines increasingly rising, and the complexification of systems, the task of auditing systems' health based on logfiles is becoming overwhelming for system administrators. This observation led to many proposals automating the processing of logs. However, most of these proposal still require some human intervention, for instance by tagging logs, parsing the source files generating the logs, etc.**

In this work, we target minimal human intervention for logfile processing and propose a new approach that considers logs as regular text (as opposed to related works that seek to exploit at best the little structure imposed by log formatting). This approach allows to leverage modern techniques from natural language processing. More specifically, we first apply a word embedding technique based on Google's `word2vec` algorithm: logfiles' words are mapped to a high dimensional metric space, that we then exploit as a feature space using standard classifiers. The resulting pipeline is very generic, computationally efficient, and requires very little intervention.

We validate our approach by seeking stress patterns on an experimental platform. Results show a strong predictive performance ($\approx 90\%$ accuracy) using three out-of-the-box classifiers.

***Keywords*—Anomaly detection, logfile, NLP, word2vec, machine learning, VNF**

I. Introduction

Gathering feedback about computer systems states is a daunting task. To this aim, it is a common practice to have programs report on their internal state, for instance through journals and logfiles, that can be analyzed by system administrators.

However, as systems tend to grow in size, this traditional logging method does not scale well. Indeed, scattered software components and applications produce heterogeneous logfiles. For instance, logging methods such as the common *syslog*, are extremelly flexible in their syntax (see the RFC [7]). Also, different logfiles may gather information with distinct types of information. For instance rule-based logging [4] traces the start and the termination of applications functions, while *syslog* event logging collects system activity. Each of them tends to describe a partial view of the whole system. In particular, [3] shows that event logging, assertion checking, and rule-based logging are orthogonal sources for system monitoring. Moreover, each partial view of the system, even when using the same logging method (or protocol), may not use the same keywords to express normal or erroneous behaviors. This plethora of available logfiles burdens log summarization.

As a result, source code analyzes and communications with application developpers are necessary for troubleshooting or auditing systems [17]. Notwithstanding, such non automatic processes are not acceptable in large computing system because troubleshooting for reconfiguration must be handled online. To address these challenges, a large number of studies proposed approaches to automate and scale up log analysis ([5], [8], [17], [23], [24]). Most approaches require however cumbersome log processing, for instance by manually tagging important events, or by parsing the source code functions to assess the fixed and variable parts of log events.

The contribution of this paper is to propose a new approach departing from this research line and considering log mining as a natural language processing task.

This approach has two main consequences, i) we lose a part of the context by under-exploiting the specificities of each structured sentence according to a predefined pattern and, most importantly, ii) our approach is agnostic to the format of the logfiles. Thus, while considering sets of logfiles as languages, we gain the ability to use modern Natural Language Processing (NLP) methods. In other words, we trade accuracy for volume, preferring the ability to inaccurately process large volumes of logfiles instead of accurately processing some tediously preprocessed logs.

As such, the question we explore in this work is: "What can off-the-shelf Natural Language Processing algorithms bring to log mining? ". We more particularly focus on such questions as "is my system in state A or state B? ". The proposed approach is rather simple and brutal. Instead of precisely tracking the events related to a transition from A to B, we collect large amounts of log events related to systems in states A and B. We then transform the logs into multidimensional vectors of features (using NLP algorithms) and train a classifier on the resulting data. The resulting pipeline is a relatively standard big data application, where we target the realization of classifiers providing accurate information about the target system state. We believe this approach is specifically interesting due to the expensive expertise usually required to preprocess the logs.

We show in this paper, through a series of experiments, that with minimum setup effort and standard tools, it is possible

to automatically extract relevant information about a system state. We more particularly use the `word2vec` algorithm of Google [16] for log mining, which is an algorithm for learning high-quality vector representations of words. It notably has been used for NLP in some previous works but not for the analysis of logfiles.

Through experiments, we illustrate the potential benefits of our approach, by providing answers to system administrators' questions when data is massively available. As an illustrative example, we focus on the detection of stress related anomalies over a broad range of configurations. More specifically, we deployed on a virtual cloud environment a virtual network function running a panel of three applications, namely a proxy, a router, and a database, to which we applied a large variety of stress patterns by means of fault injection (high CPU and memory consumption, high number of disk accesses, increase of network latency and network packet losses). We show that by simply analyzing the results of NLP processed logfiles, it is possible to detect stressed behaviors with $\approx 90\%$ accuracy.

In the following, we first present in Section II the rationale of our log mining approach, and describe our use of fault injection for validation purposes in Section II. Then, in Section III we define our case study, the experimental platform on which we deployed it, and the implementation of our approcah on this platform. Section IV presents some promising experimental results. In Section V we discuss our results, and analyze their threats to validity. Section VI describes related works regarding NLP and log mining for detection purposes. Finally, we conclude this paper in section VII.

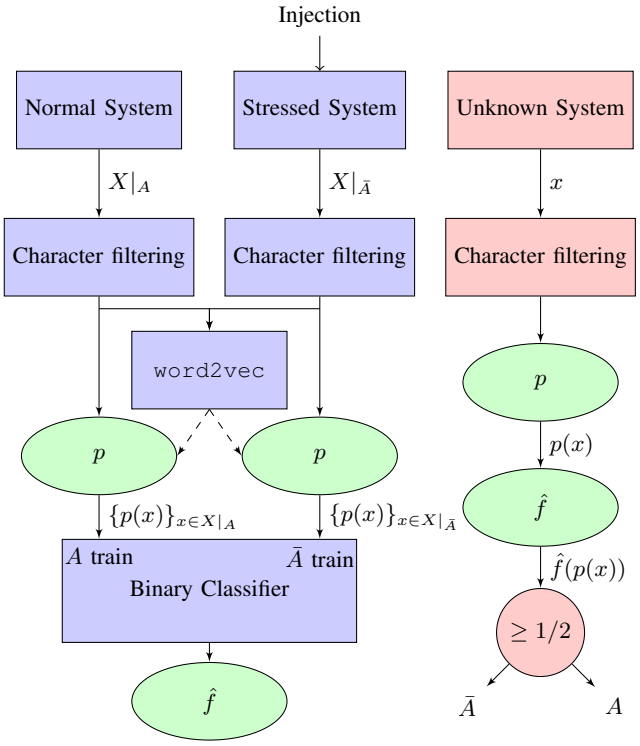


Fig. 1: General approach overview. *Left:* Training. *Right:* Inference.

II. Approach

A. General approach overview

The approach proposed as the contribution of this paper is presented in Figure 1.

Consider a set of logfiles related to a given system. Each of these logfiles contains a varying amount of lines, each line consisting of one application of the system reporting an event. Each log event (line) is a list of words.

As we consider logfiles as a natural language, we analyze these logfiles using Natural Language Processing tools. As such, we first remove all non alphanumeric characters (as required by `word2vec`) and replace them by spaces, namely `sed 's/[^a-zA-Z0-9]/ /g'`.

Secondly, we use `word2vec` from [16], a popular embedding tool employed by Google to process natural language. In a nutshell, `word2vec` produces a mapping from the set of words of a text corpus (a set of logfiles in our case) to an euclidean space say T. In the case of a 20-dimensions space $T \subset \mathbb{R}^{20}$. Thus, each word of an event gets assigned coordinates in a vector space. The enjoyable property of `word2vec` is its ability to produce *meaningful* embeddings, where similar words end up close, whereas words that are not related to each other end up far away in the embedding space.

Once each word has been mapped to the embedding space T, we define the position of a log event as the barycenter of its words. Following a similar scheme, once all log events from a given logfile have been mapped to points, we define the position of this logfile as the barycenter of the position of its log events. Hence, at the end of the process, each logfile is mapped to a single point in T. This drastic compression has one major interest: it produces a compact and useful input to traditional classifiers. Assuming $\mathcal{X}$ represents the set of all possible logfiles, such mapping can be represented as a function:

$$\begin{aligned} p : \mathcal{X} &\to T \\ x &\mapsto p(x). \end{aligned}$$

Now, assume that one has access to a large set X of observations (logfiles) on the system, corresponding to two states that we would like to characterize, say A and $\bar{A}$. Let $X|_A$ and $X|_{\bar{A}}$ be the corresponding logfiles sets. By the above described process, every observation $x \in X = X|_A \cup X|_{\bar{A}}$ can be assigned to a coordinate $p(x) \in T$.

In a third step, we train a classifier, named $\hat{f}$ hereafter, on $p(x|x \in X|_A)$. A typical such classifier $\hat{f}$ is an approximation of the ideal separation function:

$$\begin{aligned} f : T &\to [0,1] \\ p(y) &\mapsto \mathbb{P}(A|y). \end{aligned}$$

The training of a classifier requires an available set of labeled data. These labels may be for instance: normal and anomalous. In cases that labeled data is not available, one can generate them by monitoring a system while experiencing normal and anomalous behaviors. Since anomalous behaviors are undesired events and, as such, usually not frequent in

recent systems, they need to be synthesized using techniques such as fault injection. In this paper, we generate sets of normal and anomalous behaviors in a controlled manner using fault injection techniques for all anomalous behaviors, as represented in Figure 1.

Once the training is finished, the resulting classifier is used to provide, given any new production logfile x, an inferred state (anomalous or not) $\hat{f}(p(x))$ that we claim is a good approximation of the actual stress status of the system, i.e., $\mathbb{P}(A|x) \simeq \hat{f}(p(x))$. It is actually expressed as a probability and we need to set a limit over which a system is categorized as stressed, say $1/2$ as in Figure 1. In the case x contains unencountered words, those are simply ignored.

III. Case study and experimental platform

A. Case study

We hereby present our case study on virtual network function (VNF) called Clearwater[1] as well as the workload generator used during our experiments to simulate actual users of this target system. This case study was used in our preivous work [19] for anomaly detection based on monitoring data.

It constitutes a meaningful case study in that it deploys several components of different roles (e.g., router, proxy and database). While we apply our approach with no specific configuration nor a priori knowledge of the implementations for each component, we consider that our approach has good chances to generalize to various case studies.

1) Description: The service is an open source VNF named Clearwater. It provides voice and video calls based on the Session Initiation Protocol (SIP), and messaging applications. Clearwater encompasses several software components and we particularly focus our work on Bono, Sprout, Homestead shown in Figure 2.

Bono is the SIP proxy implementing the Proxy-Call/Session Control Functions. It handles users' requests and routes them to Sprout. It also performs Network Address Translation traversal mechanisms.

Sprout is the IMS SIP router, receiving requests from Bono and routing them to the adequate endpoints. It implements some Serving-CSCF and Interrogating-CSCF functions and gets the required users profiles and authentication data from Homestead. Sprout can also call application servers and actually contains itself a multimedia telephony (MMTel) application server, whose data is stored in another Clearwater component not presented in this work (when calls are configured to use its services).

Homestead is a HTTP RESTful server. It either stores Home Subscriber Server (HSS) data in a Cassandra database and masters data (i.e., information about subscribed services and locations), or pulls data from another IMS compliant HSS.

Bono, Sprout, and Homestead work together to control the sessions initiated by users and handle the entire CSCF. Our case study encompasses these three components, each one being deployed on a dedicated virtual machine (VM) of our virtualized experimental platform (see Section III-B).

[1] http://www.projectclearwater.org/about-clearwater/

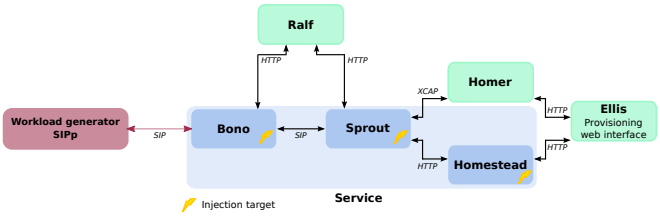


Fig. 2: Clearwater deployment.

2) Workload: IMS workloads can be emulated by means of the SIPp benchmark[2]. The benchmark contains a workload that can be configured with a number of calls per second to be sent to the IMS, and a scenario. The execution of a *scenario* corresponds to a call. A scenario is described in terms of SIP transactions in XML. A SIP transaction corresponds to a SIP message to be sent and an expected SIP response message. A call fails when a transaction fails. A transaction may fail for two reasons: either a message is not received within a fixed time window (i.e., the timeout), or an unexpected message is received. Unexpected messages are identified by the HTTP error codes 500 (Internal Server Error), 503 (Service Unavailable) and 403 (Forbidden).

The scenario run for our experimentations simulates a standard call between two users and encompasses the standard SIP REGISTER, INVITE, UPDATE, and BYE messages. The scenario is available online[3]. Timeouts are set to 10 sec as in similar experimental campaigns [2].

3) Fault injection for training and validation: Fault injection is used in our study for collecting logfiles representing both normal behaviors and stressed behaviors of a target system, in order to provide them as inputs for the training and validation of the classifiers. We emulate errors by means of injection tools that implement systems stressing. These tools were used in our previous work [19].

We call the orchestration of several executions of the target system in presence or not of error emulations an *experimental campaign*. In the following we present the errors that our injection tools emulate and describe the execution of an experimental campaign.

Error emulation. We emulate the following five types of errors, which we will be referring to as *CPU, memory, disk, network packet loss, and network latency* errors respectively:

(1) high CPU consumption,
(2) misuse of memory, i.e., increase of memory consumption,
(3) abnormal number of disk accesses, i.e., large increase of disk I/O accesses and synchronizations,
(4) network packet loss,
(5) network latency increase.

CPU errors. Abnormal CPU consumptions may arise from programs encountering impossible termination conditions leading to infinite loops, busy waits or deadlocks of competing actions, which are common issues in multiprocessing and distributed systems.

[2] http://sipp.sourceforge.net/index.html

[3] https://homepages.laas.fr/csauvana/sipp_scenario/issre2016_sipp_scenario.xml

Memory errors. Abnormal memory usages are common and happen when allocated chunks of memory are not freed after their use. Accumulations of unfreed memory may lead to memory shortage and system failures.

Disk errors. A high number of disk accesses, or an increase of disk accesses over a short period of time, emulate disks whose accesses often fail and lead to an increase in disk access retries. It may also result from a program stuck in an infinite loop of data writing.

Network packet loss and latency errors. Such errors may arise from network interfaces of the target system or from the network interconnection of the virtualized infrastructure hosting the system. We emulate packet losses and latency increases. Packet losses may arise from undersized buffers, wrong routing policies or even firewall misconfigurations. Latency errors may originate from queuing or processing delays of packets on gateways or at the target system level.

From the definition of these error types, an important experimental parameter is the injection intensity, i.e., the expected impact magnitude of the different injections from users points of view. In our study, we present results for the detection of errors with high intensities. In other terms, experimental campaigns perform injections that strongly affect the target system capability to answer users requests.

Table I presents the intensity levels that we calibrated for our Clearwater case study.

Error type	Unit	Intensity level
CPU	%	90
Memory	%	97
Disk	#process	50
Network packet loss	%	8.0
Network latency	ms.	80

TABLE I: Injection intensity levels.

Regarding the memory, disk and CPU injections, the intensity values of errors are constrained by the capacity of the operating systems (OSs) on which are deployed the applications of our case study. In other words, the intensity levels correspond to the maximum resource consumption allowed by the OS before killing the execution of the injection agent.

Considering the remaining types of injections, the corresponding intensity levels is set so as to lead to around 99% of unsuccessfully answered requests when applied in at least one VM. The unsuccessfully answered requests rate can be known from the workload logfiles.

Experimental campaigns. The experimental campaign is conducted using a customizable main script that either launches normal or anomalous executions of the target system. The experimental campaign either launches normal or stressed executions of the target system. An execution, be it normal or anomalous, produces one logfile for each VM of our target system.

We define a campaign to run as many normal executions as the number of stressed executions. The selected number of stressed executions is configured to represent all combinations of different injections (i.e., the injection of each error type, in each VM).

When running an anomalous execution, the configured injection starts after t seconds from the target system boot time, where t is randomly selected in a preconfigured interval. This process adds randomization to the set of collected logfiles, a prerequisite for the generalization of our results.

Additionally, consecutive executions of a campaign are separated by the reboot of all VMs of the target system and the workload in order to be sure to restart from a clean and unpolluted state.

As a result, the parameters of an experimental campaign are as follows: i) target VMs listed in *l_vm*, ii) error types listed in *l_type*, iii) an injection duration set in *inject_duration*, iv) a clean run duration set in *clean_run_duration*, v) an interval of values defining after which time an injection can start after a reboot set in *interval*.

Moreover, a campaign is executed as follows. Each error type is injected in a first VM, then in a second VM, etc. with reboots of the target system and the workload before each new execution. The stressed executions are orchestrated as explained in algorithm 1. Then the same number of normal executions are performed.

Algorithm 1 Orchestration of stressed executions of the target system in an experimental campaign

Input: l_vm, l_type, inject_duration, interval, clean_run_duration

```
start_workload()                                   ▷ Clean run
for vm in l_vm do                                  ▷ Runs with injections
    for err in l_type do
        start_workload()
        rand_time = random_int(interval)
        sleep(rand_time)
        inject = Injection(err, inject_duration)
        inject_in_vm(vm, inject)
        stop_workload()
        reboot_vms()
    end for
end for
```

B. Experimental platform

In the following, we first present the platform on which we run experiments. Then we describe the implementation required to carry out our experiments namely the injection agents, experimental campaign parameters, and the collection of logfiles.

1) Platform: We deployed our target system on a virtualized platform. The platform is composed of a cluster including two hypervisors and several VMs. Four VMs are deployed for our target system: one VM runs the workload and the other three respectively host the components Bono, Sprout and Homestead of Clearwater. The workload VM also has the means to control the experimental campaign launch. Two other VMs are respectively used to store logfiles collected from the target system and to analyze the stored logfiles. The deployment of the VMs is illustrated in Figure 3.

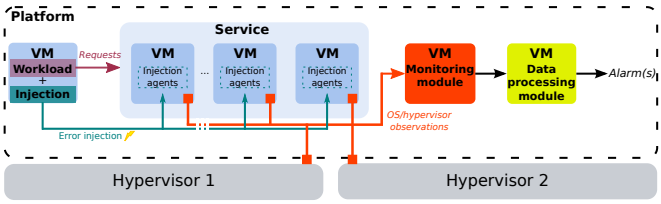


Fig. 3: Virtualized platform.

The platform is a VMware vSphere 5.1 private cloud composed of 2 servers Dell Inc. PowerEdge R620 with Intel Xeon CPU E5-2660 2.20 GHz and 64 GB memory. Each server has a VMFS storage. Each VM deployed for the target system implementation has 2 CPUs, a 10 GB memory, a 10 GB disk and runs the Ubuntu OS. VMs are connected through a 100 Mbps network.

2) Fault injection: Injections in the target system are carried out by injection agents installed in these VMs. There is one injection agent for each error type in each VM of a target system. Agents are run and stopped through an SSH connection orchestrated by the campaign main script. They emulate errors presented in Section III-A3 by means of a software implementation.

CPU and disk errors are emulated using the stress test tool `stress-ng`[4]. CPU injections run 2 processes (there are 2 cores in each VM) running all the stress methods listed in the tool documentation. The percentage of loading is set according to the intensity level of the injection.

Disk injections start several workers writing 50 Mo and 50 workers continuously calling the *sync* command, with an *ionice level* of 0. The number of writing workers is set according to the intensity level of the injection.

Memory injections are run by means of a python script reserving memory space while continuously checking whether the amount of memory space reserved by the script corresponds to the amount set by the intensity level of the injection.

Finally, we use the Linux kernel tools *iptables* and *tc* for the injection of network latencies on the POSROUTING chain, and *iptables* on the INPUT chain for the injection of packet losses. All network protocols are targeted.

3) Experimental campaigns parameters: An experimental campaign corresponds to the execution of a customizable main script that starts the workload of our target system, and either makes clean run of this target system or makes runs while performing injections in the target system VMs.

The parameters of the experimental campaigns we run are as follows. The injection duration is calibrated so as to affect several instances of workload executions (an execution lasts less than 1 sec). We calibrated the injection duration to be 10 min long in order to collect around 5000 lines of logfile for each clean run and injection. Also, we calibrated the clean run duration to be 30 min. Finally, we calibrated the start of injections to be randomly selected in the interval from 1 to 10 min. This interval allows the VMs to stabilize after a reboot.

[4]http://kernel.ubuntu.com/~cking/stress-ng/

```
Apr 18 06:44:37 cw-011 restund[1368]: stun
server ready
Apr 18 06:44:37 cw-011 bono[1284]: 2005 -
Description: Application started. @@Cause:
The application is starting. @@Effect:
Normal. @@Action: None.
Apr 18 06:45:01 cw-011 CRON[1521]:
(root) CMD (/usr/lib/sysstat/sadc 1 1
/var/log/sysstat/clearwater-sa`date +%d` >
/dev/null 2>&1)
```

Fig. 4: Example of *syslog* events.

Our experimental campaign parameters are summarized in Table II.

Campaign parameters
• l_vm $= \{Bono, Sprout, Homestead\}$
• l_type $= \{CPU, memory, disk, latency, packet\_loss\}$
• injection_duration $= 10$ min
• clean_run_duration $= 10$ min
• interval $= [1 : 10]$ min

TABLE II: Injection campaign parameters of the four experimentations.

4) Logfiles collection: The logfiles that we use in this study are generated by the Linux-based Ubuntu OS using *syslog*, the standard tool for message logging. Events are logged with a predefined pattern containing in that order the date of the event issue, the hostname of the equipment delivering the event, the process delivering the event, a priority level, the id of the process delivering the event and finally the message containing free-formatted information. For instance, no performance metrics of the system are logged. A example of *syslog* events is provided in Figure 4.

Results of previous studies [3] show that *syslog* event logging is the more suitable method to use in this context, although a combination of the several methods increases the failure coverage. The syslog facility has the advantage to gather several applications events.

During experimental campaigns, logfiles are collected by means of agents (they are represented by orange squares in Figure 3) and stored in a database for later analysis.

IV. Results

In this section, we quantitatively study the effectiveness of the presented approach by presenting the analysis results over 660 logfiles. After briefly introducing the considered metrics, we will detail the obtained results.

The main research question we seek to answer is: *Using only syslog files as input, how accurately can our algorithm distinguish Stressed and non Stressed systems?* The secondary questions are i) how sensitive are the results to the parameters used to calibrate the models of our approach? and ii) what is the ability of our approach to issue quick decision on a system state?

A. Materials and Metrics

Using the testbed presented in Section III-B we generate a set of 660 logfiles that will constitute the basis of our models training. Exactly half of these (330) originate from normal unstressed system executions. The other half captures systems with injected faults. More precisely, we ran 22 replications for each of the 5 injection campaigns over each of the 3 target VMs of our case study, for a total of $(22*3*5) = 330$ stressed logfiles.

Word2Vec training: To establish the `word2vec` training set, we use the concatenation of all 660 logfiles from which we removed all non alphanumeric characters.

`word2vec`, originally designed for NLP tasks, can be tuned with a number of different options. The most important parameter is the embedding space dimension $dim(T)$, its impact is detailed in Section IV-B2. The other parameters mostly allow to setup filters in order to optimize the computation. We deactivated all of them to keep the maximum amount of information available to the classifier. Finally, from the two methods proposed in the implementation of `word2vec`, namely `skip-gram` and `cbow` (defining whether the source context words should be predicted from target words or the opposite[5]), we chose `cbow` because of its simplicity, in order to provide an "as-simple-as-possible" solution.

Given the relatively small size of our text corpus (compared to all the English texts available on the web, namely `word2vec`'s original usecase), and the well known efficiency of the `word2vec` implementation, the overall computation is tractable on a standard computer (see Section IV-B3). Therefore, the philosophy behind implementation choices is the following: keep it simple, and keep the maximum amount of information.

From word coordinates to logfile coordinates: The output of `word2vec` is a file containing the coordinates of the $233k$ distinct words of our training corpus in T. To transform logfiles into coordinates in T, we explored two standard strategies:

bary In the barycenter approach, we first compute the position of each line of a logfile, defined as the average position of all the words it contains. Then, the position of the file is defined as the average of all its line:

$$p(f) =_{\text{def}} 1/|f| \sum_{l \in f} 1/|l| \sum_{w \in l} p(w).$$

tfidf Term frequency - inverse document frequency is a standard metric of information retrieval. Compared to the barycenter approach, words are weighted by their frequency in the document. That is, a frequent (common) word will proportionally have less weight than a rare word when computing the average position of a logfile. We relied on the `scikit-learn`[6] standard implementation of the function.

The output of this step is a matrix of $660 \times dim(T)$ entries decorated with their corresponding target labels (stressed, unstressed system).

[5]See one implementation explaination https://www.tensorflow.org/tutorials/word2vec. Last read on 13/08/2017.

[6]http://scikit-learn.org/

Classifiers: Binary classifiers are amongst the most common and understood classifiers in machine learning. We restricted our study to three simple and state of the art approaches: Naive Bayes, Random Forests and Neural Networks. We relied on the following `scikit-learn` library implementations: Random Forest Classifier, MLP Classifier, and Gaussian NB. All these algorithms belong to the class of supervised algorithms. In other words, they require labeled training data, although we could have used unsupervised approaches such as the ones tested in [8], i.e., Principal Components Analysis and Invariant mining.

Again, the philosophy of our approach is to refrain from fine tuning those implementations and to assess the global strategy as a hole. We therefore used the default parameters on all these algorithms.

Classifier Assessment: To assess the classification accuracy, we used the standard 10-fold validation approach. We first randomly divided the training set in 10 equal sized chunks. Each possible group of 9 chunks was used to train our classifier while the remaining chunk was used as a test.

Let $\{X_i\}_{1 \leq i \leq 10}$ be a partitioning of X into 10 chunks. Let X_j be the tested chunk, and let T_j (resp. F_j) be the subset of stressed (resp. unstressed) logs of X_j. The set of true positives TP_j for X_j is defined as:

$$TP_j = \{x \in X_j \text{ s.t. } \hat{f}_j(x) \geq 1/2 \wedge x \in T_j\}.$$

Logs that belong to stressed machines and to which the classifier $\hat{f}_j$ (trained using $\cup_{i \neq j} X_i$) assigned a probability greater than 1/2 of being stressed are true positives for X_j. Similarly, the set of false positives FP_j for X_j (logs belonging to unstressed machines but detected as more likely stressed) is defined as:

$$FP_j = \{x \in X_j \text{ s.t. } \hat{f}_j(x) \geq 1/2 \wedge x \in F_j\}.$$

Notice that the true negative and false negative sets are symmetrically defined.

To get a closer look at $\hat{f}_j$, one can use Receiver Operating Characteristics (ROC). That is, let $s \in [0, 1]$ be a "safety level" one wants to apply to $\hat{f}$-based decisions. Let $X_j^s = \{x \in X_j, \hat{f}_j(x) \geq s\}$ be the subset of X_j containing only the logs detected as stressed with probability at least s. For each value of s, it is thus possible to define a true positive rate $TPR_s = |X_j^s \cap T_j|/|T_j|$ and a false positive rate $FPR_s = |X_j^s \cap F_j|/|F_j|$. The graphical representation of the obtained $\{FPR_s, TPR_s\}$ couples provides a precise visual description of $\hat{f}$'s performance, as in Figure 5 that will be presented shortly hereafter.

B. Results analysis

In the following, after exploring the detailed results obtained using a typical trained classifier, we study the impact of the embedding host space dimension. We then study the runtime overhead of our approach.

1) Accuracy: Figure 5 presents the ROCs obtained on a typical configuration. More precisely, in this setup, we used $dim(T) = 20$ and explored various aggregation/classifier configurations. The results are very good, with Neural Network

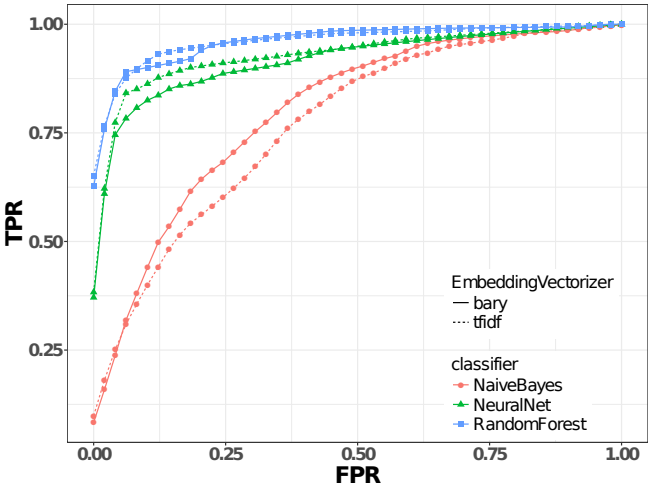


Fig. 5: Receiver Operating Characteristic of 3 classifiers, for $dim(T) = 20$. This plot shows the True Positive Rate of every classifier as a function of the False Positive Rate of the same classifier.

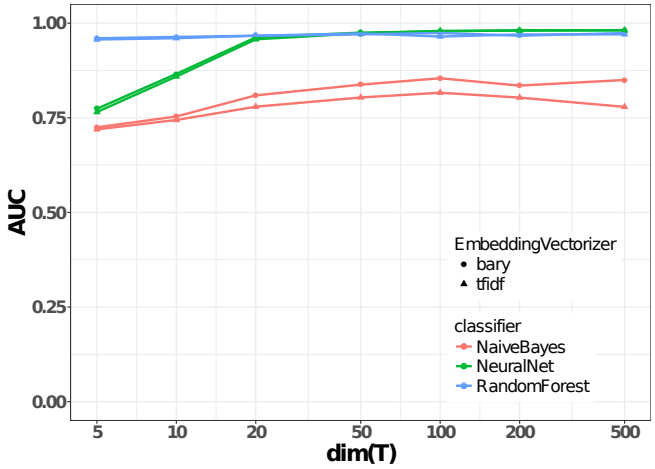


Fig. 6: Area Under the ROC Curves (AUC) capturing the performance of our classifiers, as a function of the number of dimensions of the embedding space

and Random Forest exhibiting a strong classification accuracy ($> 95\%$ AUC). The aggregation technique (i.e., based on tf-idf or barycenter) has little impact. Naive Bayes performs considerably better than random (77% and 81% AUC for tf-idf and barycenter resp.), but is visibly less precise than the other two classifiers. These very good results confirm the soundness of the approach.

One can have a more detailed look at the origin of misclassifications. Table III exhibits the confusion matrix of Neural Network (using barycenter and $dim(T) = 20$). Although around 90% of the targets get correctly categorized, one can see that the errors are slightly leaning towards false positives (that is, an unstressed system is wrongfully categorized as stressed). Although this is not the purpose of this study, it is possible to exploit this imbalance for an overall better classification accuracy (for instance by raising a $1/2$ limit over which a system is categorized as stressed). The stress patterns are not very homogeneously detected, with Latency stress being 7 times more efficiently detected than CPU stress. However, because of the accuracy of the considered classifier, these results only concern a small number of events, and therefore have a low statistical power. Table IV presents the misclassified entries by application: all three applications (namely Bono, Sprout and Homestead) yield to similar classification accuracy.

TABLE III: Confusion matrix for the Neural Network classifier, using $dim(T) = 20$, and barycenter: detailed by stress type

Stress Type	Detected As Stressed (True)	Detected as Unstressed (False)
No Stress	0.115	0.885
Packet loss	0.939	0.061
Latency	0.985	0.015
Memory	0.939	0.061
Disk	0.970	0.030
CPU	0.893	0.106

TABLE IV: Confusion matrix for the Neural Network classifier, using $dim(T) = 20$, and barycenter: detailed by application

Target Machine	Requests	Number of misclassifications	Success Rate (%)
Bono	220	19	91.4
Sprout	220	17	92.3
Homestead	220	20	90.9

2) Parameters sensitivity: We here focus on two choices of importance: the dimension of the embedding space $\dim(T)$, and the classifier algorithm. To compare our classifiers, we use the Area Under Curve (AUC) measure. In a nutshell, it measures the area under the ROC of a classifier. That is, an AUC of 1 denotes a perfect classification, while an AUC of 0 denotes a worse than random prediction. It is also commonly presented, given a random positive (stressed) and random negative (unstressed) example, as the probability for the classifier to rank the negative example below (that is, less stressed) the positive example. The ROC AUC is know to well summarizes ROC curves [1].

Figure 6 provides the AUC measures for our 3 considered classifiers for various embedding space dimensions. As expected, increasing the number of dimensions increases the classification accuracy: more information helps. This increase is however very limited: apart from Neural Network, where increasing dimensions from 5 to 20 has a visible impact, classifier accuracies all stay stable for $dim(T) > 20$. This is good news, as such parameter can be hard to tune a priori.

More generally, this figure confirms the previous observations: classification is very accurate, especially using Neural Network and Random Forest, with AUCs consistently scoring above 0.95.

3) Timing performance: When selecting a classifier, the expected classification accuracy is the most important criteria. However, in operational contexts, another crucial criteria is the computational complexity of both training and prediction.

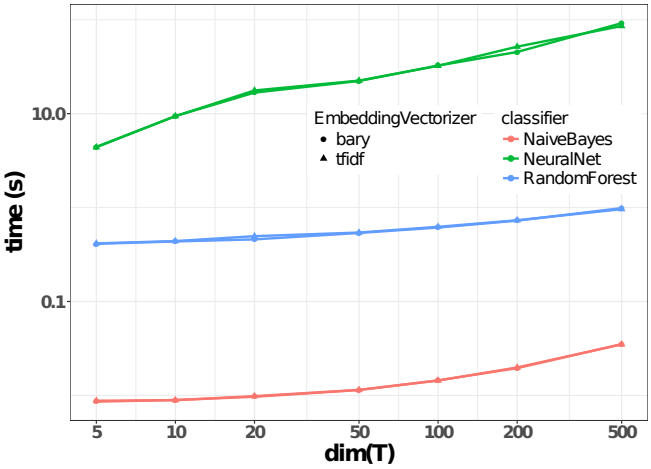


Fig. 7: Training wall time of the classifiers on 660 instances, for varying embedding space dimensions. Notice the log-log scale.

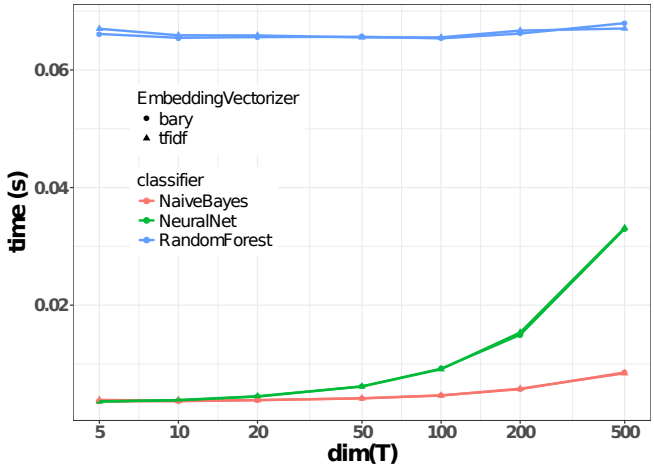


Fig. 8: Time taken for a trained model to issue one prediction. Notice the log-lin scale.

To provide some insights, we recorded wall clock times of the training of machine learning models (Figure 7) and of individual prediction of these models (Figure 8) operations. Those were performed on classical Macbook Pro with 16 GB of RAM and a quad-core Intel i7.

Interestingly, these figures provide a new perspective on our classifiers. Results confirm the reputation of each of those models: Naive Bayes is very simple, it is quickly trained and provides fast answers. Neural Network is a considerably more complex model whose training requires significantly more time. However, once trained it is able to answer reasonably fast. Contrariwise, Random Forest is quickly trained but requires considerably more time to issue predictions. Issuing a prediction requires on average $66ms$ (resp. $5ms$ and $11ms$) for Random Forest (resp. Naive Bayes and Neural Network).

Not surprisingly, increasing $dim(T)$ comes with a computational cost (as it increases the number of features on which each model is trained), but since Section IV-B1 shows that $dim(T) = 20$ is already sufficient to obtain accurate results, we conclude that this approach is computationally tractable. The most prominent decision is the choice of the classifier: although the simplest possible classifier (Naive Bayes) provides cheap and reasonable answers, more efficient classifiers like Random Forest or Neural Network will cost a bit more, either at training time, or at prediction time.

To conclude, this results section explored the performance of three state of the art classifiers exploiting the log positions. These classifiers exhibit a strong performance for a reasonable cost. The most important parameter, the dimension of the host space $dim(T)$, is not very sensitive: values ranging from 20 to 200 will roughly deliver the same performance. Although many parameters could be precisely tuned to optimize the classifiers, we believe these good results obtained using mostly default values of COTS tools already validate the soundness of our approach. More precisely, these show the extremely powerful effect of the `word2vec` embedding applied to logs: it allows to summarize each logfile to a single point in T while keeping enough information to allow an efficient classification.

V. Discussion

Our approach leaves one common question of all machine learning approaches intact: how general are the learned models? In other words, are the classifiers built in this context able to provide accurate answers in different contexts, application environments, under different injection campaigns? Although this question is definitely of interest, we argue its scope goes well beyond this paper. Philosophically, this study shows that it is easy to train efficient classifiers. But informally, a classifier is only as good as its training data. The availability of labelled training data can clearly limit the applicability of our approach. The advantage of fault injection if to gather relevant labeled datasets in a short time period. Although it enables to evaluate our approach in a straghforward manner this implemention can be cumbersome. However, while we rely on fault injection to gather datasets, other sources exist : user-based feedback, crowed sourced datasets, and crash reports of large scale deployments.

In our previous work [19] we analyzed monitoring counters such as CPU consumption or number of disk accesses for anomaly detection. Results from counter-based detection showed a good predictive performance that is yet not fully aligned with the results of this study. For instance, latency errors were significantly harder to detect. In this study, we show that by solely mining *syslog* files we could detect anomalies with high accuracy for all types of anomalies. Consequenlty, we believe our approach is largely promising. As for future work, we plan to study an hybrid approach leveraging both logging and counter-based data in order to further evaluate their potential complementarity. what type of logs enhance or weaken the efficiency of our approach.

Finally, results presented in this paper show that our approach detects with the same accuracy the stresses injected in either type of application of our case study (i.e., proxy, router and database). In other words, the analysis of system related logs such as syslog is an efficient way to summarize

application behaviors for stress detection with no regard to the type of application. We believe however that syslog events are not enough to derive application dataflows that may allow to detect other types of anomalies or more importantly for administrators, to diagnose the origin of an anomaly. Consequently, we need to explore in future work other types of logs, notably the ones generated by our case study application.

VI. Related work

In this study, we use a `word2vec`-based method for log mining with a validation-purposed application of detecting stressed behaviors in computing systems. `word2vec` is a method for learning high-quality vector representations of words. It has been used for NLP in some previous works but not for the analysis of logfiles. In comparison, our previous work [19] focuses on anomaly detection based on monitoring data collected by means of a specific software agent, deployed beforehand on target machines, and providing numerical metrics on the system behavior. Here we exploit the default system-produced textual logs to predict stress. Beside the deep technical differences, our approach allows different use-cases, like post-mortem analysis of the behavior of the several processes being executed in the targeted systems.

Consequently, in the following we present separately several works related to NLP and other works related to logfiles analysis for detection purposes.

NLP applications. In the literature, most of the NLP algorithms are used for document processing [26] to isolate references of a given subject in a document and detect the sentiments of the writer, or to exploit tweets [11] to detect cyber-attacks such as distributed denial of service.

To the best of our knowledge, relatively few works exploit NLP for a different purpose than document analysis. We provide here a quick summary of these non-traditional uses of NLP. In [15], the authors use a NLP technique called Latent Semantic Indexing to identify source code documents that match a user query expressed in natural language. They use the same technique in [14] to detect similar piece of code (i.e., duplicated functions) in software systems code. In addition, Latent Dirichlet Allocations are used for a similar purpose in [20]. NLP is also applied on network packet payloads for network intrusion detection in [18]. In [10], customers accesses to businesses URLs are analyzed using a `word2vec`-based method to propose better services to customers. Finally, NLP is also used to detect design and requirement debts [13] from comments of ten open source projects.

Log mining for detection purposes. Although some works propose new methods to generate relevant log events as in [4], logfiles still gather a wide range of events and evaluating their information in the execution context or weighting their gravity is still intricate. For instance, the authors of [17] analyze a wide range of logs with engineers and compare events signaling failures to the engineers feedback on actual failures. It turns out that the number of actual failures is lower than the failures reported by logs. Also they point out that syslog message severity level is of "dubious value", and that it is essential to take into account the operational context during which log events are collected. Nevertheless, logfiles analysis for anomaly (e.g., crash, fault, OS stressing...) detection in computing systems has been widely studied and it is still an active research field, in particular when considering the ever more complex recent computing systems.

Execution traces of streaming applications are analyzed in [9] in order to detect anomalies. The authors analyze traces by means of the merging pattern mining method applied on patterns of events (i.e., lines of traces). Then they build a graph representing the dataflow between the different computing units of the application. Likewise, in [21] the authors analyze the temporality of execution traces in order to derive system states from their estimated control flows. The authors of [25] also work on the ordered nature of logfiles. They exploit time series potentially hidden behind logs events for failure symptoms detection. They use a probabilistic modeling using a mixture of Hidden Markov Models (HMM) to represent different time windows (i.e., sessions) of logs event. They propose a new method for the learning of the HMM mixture working online.

Automatic techniques based on machine learning or statistics algorithms have been widely used for this matter, as in [6] where the authors propose a new approach for disk failure prediction. More precisely, they analyze by means of a Support Vector Machine (SVM) model, sequences of syslog events based on syslog tag numbers sequences or key strings in events. In [22], the author proposes a new algorithm for the clustering of log events and implements a tool based on it named SLCT. Logfiles parsing is exploited in [24]. The parsing uses log patterns identified from a static analysis of source code. Then, two types of features are computed from the entire available logfiles, and they are fed to the PCA-based anomaly detection algorithm for an offline detection. A log extractor for anomaly detection is studied in [12]. The extractor uses log clustering based on the Levenshtein editing distance to evaluate the similarities amongst log events strings (i.e., two strings are close together if there is a minimal number of actions to change the first string into the other). Templates are then extracted from log clusters. Finally, a sequence of log events matching patterns is created and feed to a machine learning algorithm. The Naive Bayes, and Recurrent Neural Networks are evaluated.

VII. Conclusions and Future work

In this paper, we tackled the problem of anomaly detection by mining logs produced by running systems. Differently to previous studies, we develop a *linguistic* approach by considering logs as regular plain text documents. This enables to exploit recent NLP techniques to extract information from the grammatical structure and context of log events. Logfiles are represented as a set of features that can be processed by standard machine learning algorithms. As such this approach shifts the burden of log preprocessing toward the collection of representative datasets. It is a good trade when data is massively available like in recent distributed systems.

Our experimental campaigns on different components of a VNF rely on fault injection to synthetize anomalous behaviors and collect relevant datasets on demand. We more particularly focus on the case of stress detection and show that strong predictors ($\approx 90\%$ accuracy) are easily trained with no human intervention in the loop. Even though we focus on stress

detection in this work, our approach is fitted for computing systems administrators for the online detection of any type of anomaly.

As for future work, we plan to explore unsupervised classifiers that would not restrain our approach scope to labelled training data and mostly known anomalies. *Syslog* files are used in this study, however we plan to inquire about what type of logfiles (e.g., `dmesg`, application logs...) enhance or weaken the efficiency of our approach. Also, we plan to extend our study to more precise online event troubleshooting while combining this detection approach with our previous work on counter-based detection [19].

REFERENCES

[1] A. P. Bradley, "The use of the area under the roc curve in the evaluation of machine learning algorithms," *Pattern recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.

[2] L. Cao, P. Sharma, S. Fahmy, and V. Saxena, "Nfv-vital: A framework for characterizing the performance of virtual network functions," in *Network Function Virtualization and Software Defined Network (NFV-SDN), 2015 IEEE Conference on*, Nov 2015, pp. 93–99.

[3] M. Cinque, D. Cotroneo, R. D. Corte, and A. Pecchia, "Characterizing direct monitoring techniques in software systems," *IEEE Transactions on Reliability*, vol. 65, no. 4, pp. 1665–1681, Dec 2016.

[4] M. Cinque, D. Cotroneo, and A. Pecchia, "Event logs for the analysis of software failures: A rule-based approach," *IEEE Transactions on Software Engineering*, vol. 39, no. 6, pp. 806–821, June 2013.

[5] M. Farshchi, J. G. Schneider, I. Weber, and J. Grundy, "Experience report: Anomaly detection of cloud application operations using log and cloud metric correlation analysis," in *Software Reliability Engineering (ISSRE), 2015 IEEE 26th International Symposium on*, Nov 2015, pp. 24–34.

[6] R. W. Featherstun and E. W. Fulp, "Using syslog message sequences for predicting disk failures," in *Proceedings of the 24th International Conference on Large Installation System Administration*, ser. LISA'10. Berkeley, CA, USA: USENIX Association, 2010, pp. 1–10.

[7] R. Gerhards, "The Syslog Protocol," RFC Editor, RFC 5424, March 2009.

[8] S. He, J. Zhu, P. He, and M. R. Lyu, "Experience report: System log analysis for anomaly detection," in *2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE)*, Oct 2016, pp. 207–218.

[9] O. Iegorov, V. Leroy, A. Termier, J. F. Mehaut, and M. Santana, "Data mining approach to temporal debugging of embedded streaming applications," in *2015 International Conference on Embedded Software (EMSOFT)*, Oct 2015, pp. 167–176.

[10] R. Kanagasabai, A. Veeramani, H. Shangfeng, K. Sangaralingam, and G. Manai, "Classification of massive mobile web log urls for customer profiling analytics," in *2016 IEEE International Conference on Big Data (Big Data)*, Dec 2016, pp. 1609–1614.

[11] R. P. Khandpur, T. Ji, S. Jan, G. Wang, C.-T. Lu, and N. Ramakrishnan, "Crowdsourcing cybersecurity: Cyber attack detection using social media," *arXiv preprint arXiv:1702.07745*, 2017.

[12] C. Liu, "Data analysis of minimally-structured heterogeneous logs : An experimental study of log template extraction and anomaly detection based on recurrent neural network and naive bayes." Master's thesis, KTH, School of Computer Science and Communication (CSC), 2016.

[13] E. Maldonado, E. Shihab, and N. Tsantalis, "Using natural language processing to automatically detect self-admitted technical debt," *IEEE Transactions on Software Engineering*, vol. PP, no. 99, pp. 1–1, 2017.

[14] A. Marcus and J. I. Maletic, "Identification of high-level concept clones in source code," in *Proceedings 16th Annual International Conference on Automated Software Engineering (ASE 2001)*, Nov 2001, pp. 107–114.

[15] A. Marcus, A. Sergeyev, V. Rajlich, and J. I. Maletic, "An information retrieval approach to concept location in source code," in *11th Working Conference on Reverse Engineering*, Nov 2004, pp. 214–223.

[16] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[17] A. Oliner, "What supercomputers say: A study of five system logs," in *Proceedings of DSN 2007*, 2007.

[18] K. Rieck and P. Laskov, "Detecting unknown network attacks using language models," in *Proceedings of the Third International Conference on Detection of Intrusions and Malware & Vulnerability Assessment*, ser. DIMVA'06. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 74–90.

[19] C. Sauvanaud, K. Lazri, M. Kaâniche, and K. Kanoun, "Anomaly detection and root cause localization in virtual network functions," in *27th IEEE International Symposium on Software Reliability Engineering, ISSRE 2016, Ottawa, ON, Canada, October 23-27, 2016*, 2016, pp. 196–206.

[20] T. Savage, B. Dit, M. Gethers, and D. Poshyvanyk, "Topicxp: Exploring topics in source code using latent dirichlet allocation," in *2010 IEEE International Conference on Software Maintenance*, Sept 2010, pp. 1–6.

[21] J. Tan, X. Pan, S. Kavulya, R. Gandhi, and P. Narasimhan, "Salsa: Analyzing logs as state machines," in *Proceedings of the First USENIX Conference on Analysis of System Logs*, ser. WASL'08. Berkeley, CA, USA: USENIX Association, 2008, pp. 6–6.

[22] R. Vaarandi, "A data clustering algorithm for mining patterns from event logs," in *Proceedings of the 3rd IEEE Workshop on IP Operations Management (IPOM 2003) (IEEE Cat. No.03EX764)*, Oct 2003, pp. 119–126.

[23] Y. Watanabe, H. Otsuka, M. Sonoda, S. Kikuchi, and Y. Matsumoto, "Online failure prediction in cloud datacenters by real-time message pattern learning," in *Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference on*, Dec 2012, pp. 504–511.

[24] W. Xu, L. Huang, A. Fox, D. Patterson, and M. I. Jordan, "Detecting large-scale system problems by mining console logs," in *Proceedings of the ACM SIGOPS 22Nd Symposium on Operating Systems Principles*, ser. SOSP '09. New York, NY, USA: ACM, 2009, pp. 117–132.

[25] K. Yamanishi and Y. Maruyama, "Dynamic syslog mining for network failure monitoring," in *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, ser. KDD '05. New York, NY, USA: ACM, 2005, pp. 499–508.

[26] J. Yi, T. Nasukawa, R. Bunescu, and W. Niblack, "Sentiment analyzer: extracting sentiments about a given topic using natural language processing techniques," in *Third IEEE International Conference on Data Mining*, Nov 2003, pp. 427–434.

3 Leçons tirées de la conception d'un capteur réparti pour la recherche en aérologie

CLUE: Lessons Learned from Designing a Distributed Sensor for Aerology Research

Mimicking data science process for agile development of sensing system

Christophe Bertero <christophe.bertero@laas.fr>
Advisors : Matthieu Roy, Gilles Tredan (system and data science) & Jean-Francois Léon (aerology)

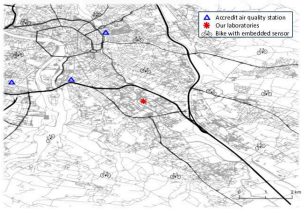
Context

- Curse of dimensionality
- Low cost, miniaturized gas sensors

Bike fleet: a solution

- Long distance tracks
- High diversity: cycle paths, sideways ...
- Human-centered

A promising carrying platform !



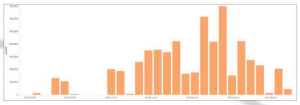
Challenges

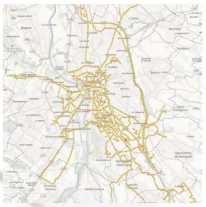
- Low quality sensor (low accuracy, time drifting)
  - ➢ Calibration of sensor fleet
  - ➢ Data density vs data quality
- Interpretability of system perception at urban scale
  - ➢ Air pollution
  - ➢ Human mobility
- User privacy and data sharing

6. Second version and collected data

5'801'307 data points

Toulouse, South of France, 4th city





System improvements :
- LoRa + USB → micro SD
- Sensor in the open air
- Dynamo → Powerbank (10x5x2cm)

2. User-invisible system

MSP430 + BLE + 2Go micro SD
LoRa + USB
3D printed enclosure with suitable airflow
Hub dynamo
1Hz sensors :
- Pollution markers: CO, NO2 (MiCS4514)
- Temperature, Pressure, Humidity (BME280)
- Accelerometer
- GPS

5. A system for experimenters

Experience report :
- Long engineering process and assistance
- Sensor heaters delay
- Underestimation of the energy consumption of the sensor
- GPS init. delay leading to spares data for short trips

→ Cut the complexity thanks to careful users without losing interpretability

Call to volunteers

From 09/01/2018 to 03/15/2019

10 sensors on personal bikes

14 LAAS CNRS experimenters:

ease automatic calibration (see 7.) since they meet at the laboratory

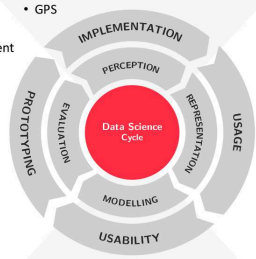
Huge motivation of bicycle commuting researchers

1. A massively distributed system

Crowdsensing → used by many different careless users
- User-invisible system
- Robust design

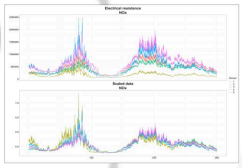
Carried by rental bikes
- Need distant communication

LA MAISON DU VELO TOULOUSE



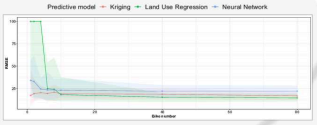
3. Sensor Calibration

- Laboratory calibration of each sensor is unrealistic
- Lab experiments confirm gradient evaluation capabilities (better seen after scaling)

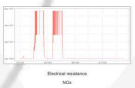


4. Dimensioning

Only few quality measurements collected
→ How many distributed system do we need ?



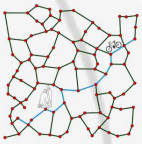
- Determine usage conditions, e.g. without high humidity



7. Work in progress

- Automatic calibration

Vicarious calibration

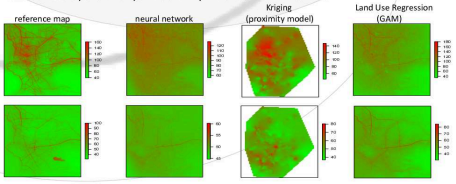


Rendezvous calibration

- Confront interpretation workflow suggest by the feasibility study with real measurements

4bis./8. Exploitability

How to interpret data produced by a distributed sensor ?



With 10 simulated bikes

4 Table de conversion ppm – μg/m^3

TABLE 1 – Table de conversion entre ppm et μg/m^3. 1 ppm = 1000 ppb. Les valeurs de conversion sont fournies par la Commission Européenne et l'Organisation Mondiale de la Santé. Elles dépendent de la température et de la pression.

Polluant	Commission Européenne 20 °C – 1013mb	OMS 25 °C – 1013mb
Ozone (O3)	1 ppb = 1,9957 μg/m^3	1 ppb = 1,96 μg/m^3
Dioxyde d'azote (NO_2)	1 ppb = 1,9125 μg/m^3	1 ppb = 1,88 μg/m^3
Monoxyde de carbon (CO)	1 ppm = 1,1642 μg/m^3	1 ppm = 1,15 μg/m^3
Dioxyde de soufre (SO2)	1 ppb = 2,6609 μg/m^3	1 ppb = 2,62 μg/m^3
Benzène	1 ppb = 3,2430 μg/m^3	1 ppb = 3,19 μg/m^3
1,3-butadiene	1 ppb = 2,2452 μg/m^3	1 ppb = 2,21 μg/m^3

5 Compléments sur le Krigeage

Autres types de Krigeage Plusieurs types de Krigeage existent en fonction de la forme de μ choisie. Les plus courants sont :

- simple : $\mu(s) = m$ avec m constante connue
- ordinaire : $\mu(s) = \mu$ avec μ constante inconnue
- universel : $\mu(s) = \sum_i \beta_i f_i(s)$ (avec généralement f_i polynomial), i.e. une tendance de μ est estimée à partir d'un modèle additif qui dépend de la position spatiale.
- avec dérive externe : $\mu(s) = \sum_i \beta_i f_i(\omega(s))$ avec $\omega(s)$ un vecteur de méta-données de la position. Il s'agit d'une généralisation du Krigeage universel.

Remarques

- Dans le cas du Krigeage avec dérive, si les coordonnées géographiques font partie des méta-données, l'indépendance spatiale entre μ et δ n'est plus assurée. La méthode d'estimation du semi-variogramme par les moments n'est plus possible. Des moyens de passer outre existent mais ne seront pas présentés ici.
- Les hypothèses ne supposent pas une distribution gaussienne de Y. Pourtant, de meilleurs résultats sont observés (Janssen et *al.*, 2008). Par similarité avec les Modèles Additifs Généralisés (dits GAM [4]), nous pouvons utiliser une fonction de lien (link function) pour viser d'autres distributions, mais la transformation inverse peut biaiser le résultat final, ce qui déroge au Krigeage. Quelques adaptations sont donc nécessaires, mais possibles.

Semi-variogramme δ est une fonction aléatoire stationnaire, d'espérance nulle et de forme supposée connue. Une hypothèse courante est de supposer δ isotopique, c'est-à-dire une invariance de δ en fonction de la direction.

4. `https://en.wikipedia.org/wiki/Generalized_additive_model`

Le semi-variogramme γ de Y est estimé en fonction de la distance h entre deux positions :

$$\begin{aligned}\gamma(h) &= \frac{1}{2}Var[Y(s+h) - Y(s)] \\ &= \frac{1}{2}Var[(\mu(s+h) + \delta(s+h)) - (\mu(s) + \delta(s))]\end{aligned}$$

Pour cela, nous utilisons la méthode des moments qui fournit un estimateur de la variance : $Var(A, B) = \frac{1}{n}\sum_i (A(i) - B(i))^2$, avec A et B deux positions spatiales où la variable explicative Y est connue. En effet, la forme de μ est connue et dépend du Krigeage choisi (en fonction du type de Krigeage) ; celle de δ est supposée connue.

En pratique, nous essayons plusieurs modèles pour δ (Gaussian, Exponential...), nous les paramétrisons avec les données et nous gardons le plus pertinent. [5]

La forme générale d'un semi-variogramme est présentée Figure 2. Les notions de pépite, palier et portée sont expliquées sur le schéma, mais ne figurent pas nécessairement dans tout semi-variogramme (cf. non existence de palier expliquée en note).

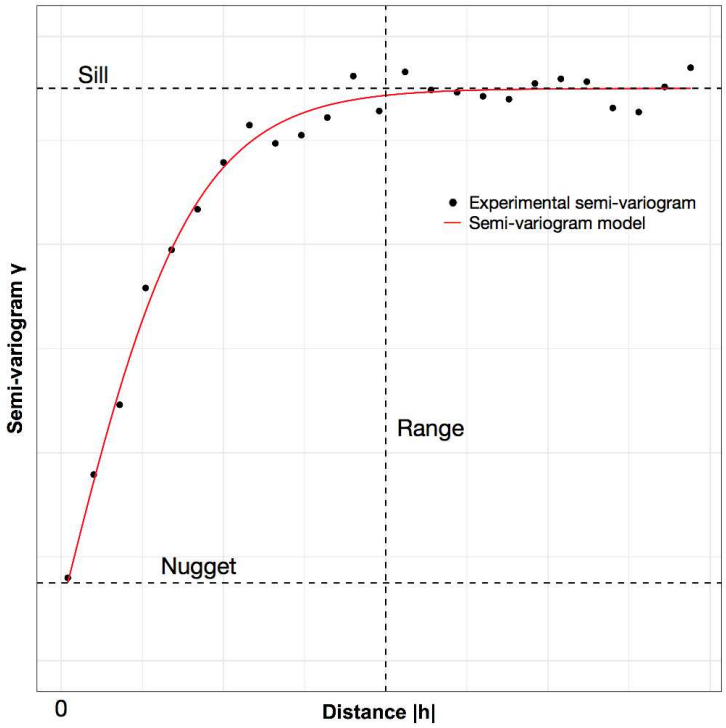


FIGURE 2 – Forme d'un semi-variogramme.

Hypothèses d'existence mathématique de la variance du résidu et du biais Deux jeux d'hypothèses sur δ sont couramment utilisés pour assurer l'existence de la

5. L'hypothèse d'existence mathématique de la variance du résidu et du biais impose certaines hypothèses sur la forme de γ. Par exemple, l'hypothèse de Stationnarité de second ordre, plus contraignante, implique l'existence d'un palier (borne supérieure) de la valeur de la semi-variance, et autorise de nouveaux modèles (Spherical, Nugget...).

variance du résidu et du biais :

- Stationnarité de second ordre :
  - $\forall s, E[\delta(s)] = 0$,
  - $\forall s, s+h, Cov[\delta(s), \delta(s+h)]$ ne dépendant que de h,
- Stationnarité intrinsèque :
  - $\forall s, s+h, E[\delta(s+h) - \delta(s)] = 0$,
  - $\forall s, s+h, Var[\delta(s+h) - \delta(s)]$ ne dépendant que de h.

La Stationnarité de second ordre est ainsi plus contraignante et implique même la Stationnarité intrinsèque.

6 Figures complémentaires pour notre simulation sur la ville de Marseille (chapitre 2)

6.1 Schéma de notre réseau de neurones

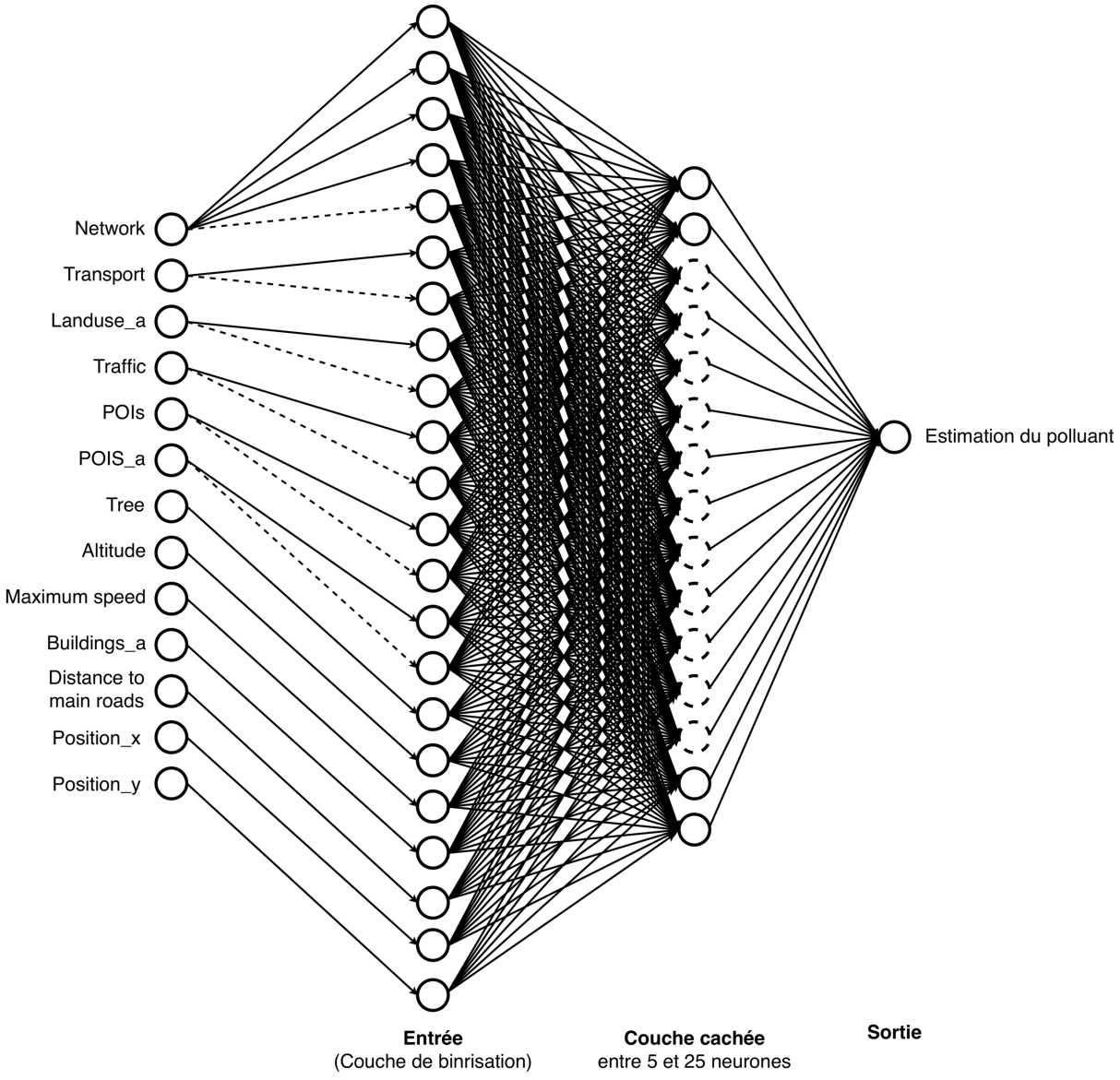


FIGURE 3 – Forme de nos réseaux de neurones.

6.2 Taux de couverture du jeu de mesures synthétiques en fonction du nombre de trajets

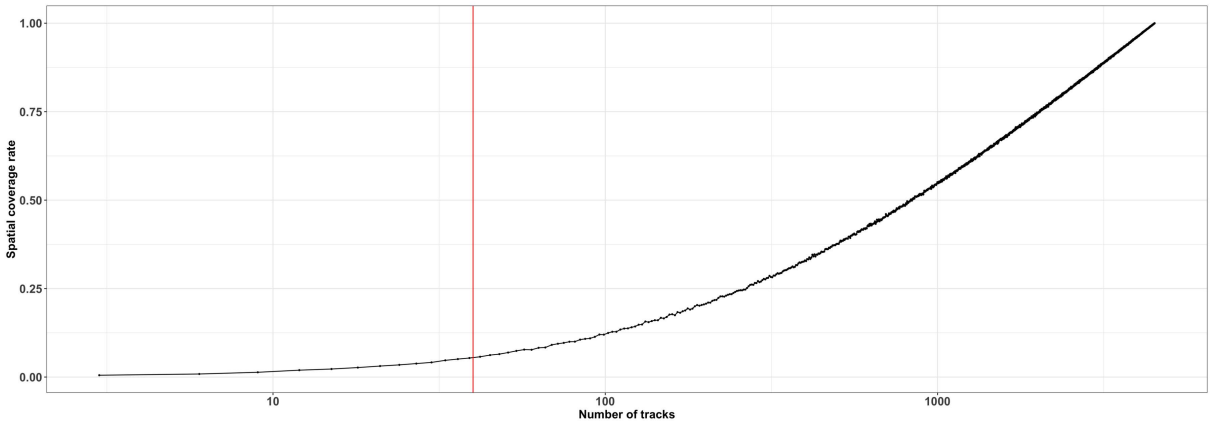


FIGURE 4 – Taux de couverture spatiale en fonction du nombre de trajets, à un échantillonnage tous les 100 mètres.

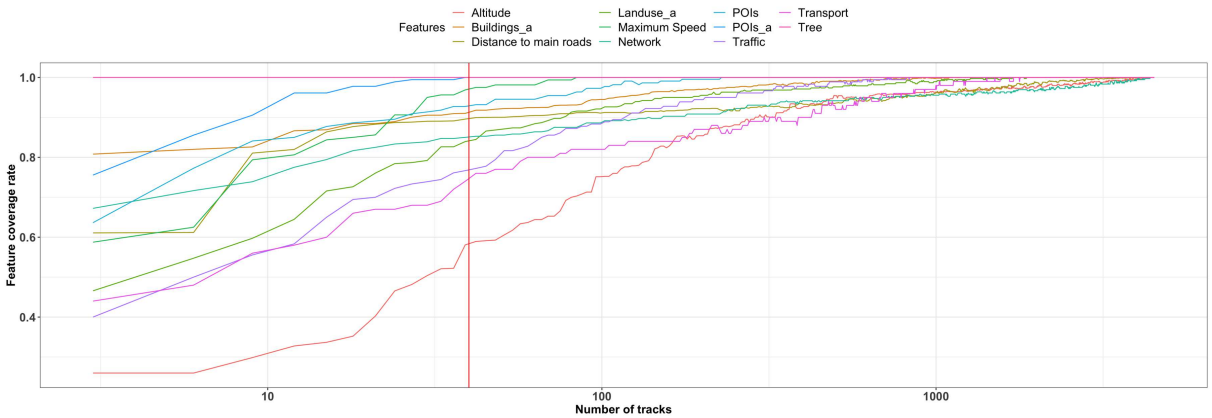


FIGURE 5 – Taux de couverture des méta-données en fonction du nombre de trajets, à un échantillonnage tous les 100 mètres.

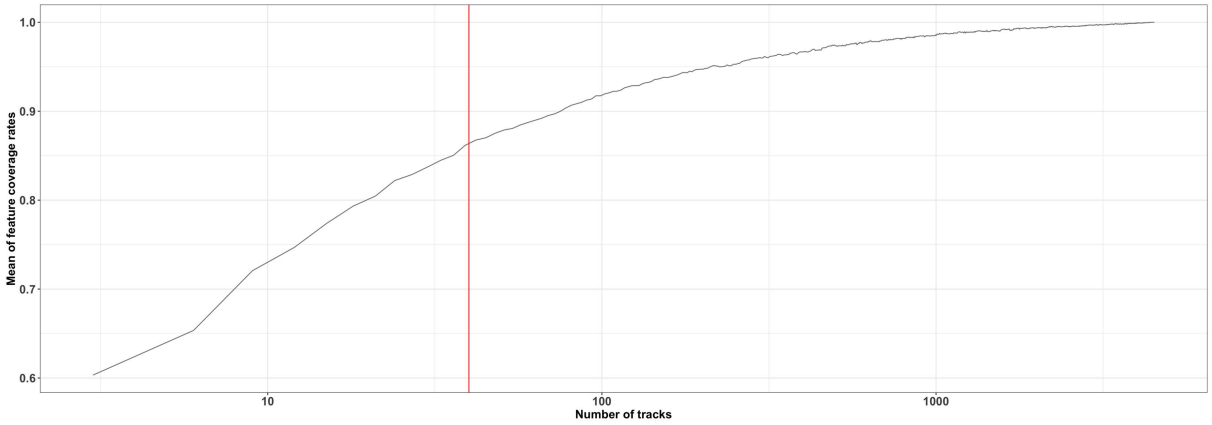


FIGURE 6 – Taux de couverture moyen des méta-données en fonction du nombre de trajets, à un échantillonnage tous les 100 mètres.

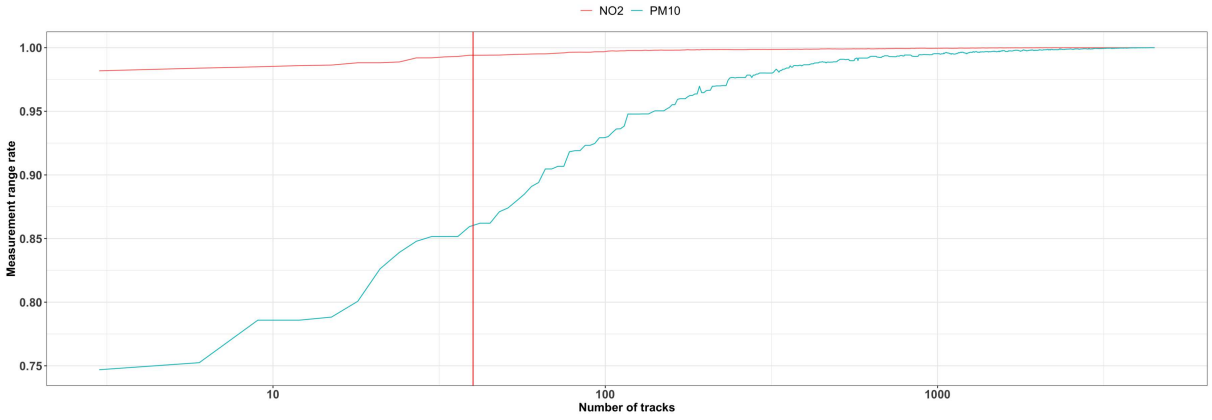


FIGURE 7 – Taux de couverture de l'étendue du polluant en fonction du nombre de trajets, à un échantillonnage tous les 100 mètres.

6.3 Taux de couverture du jeu de mesures synthétiques en fonction de l'échantillonnage

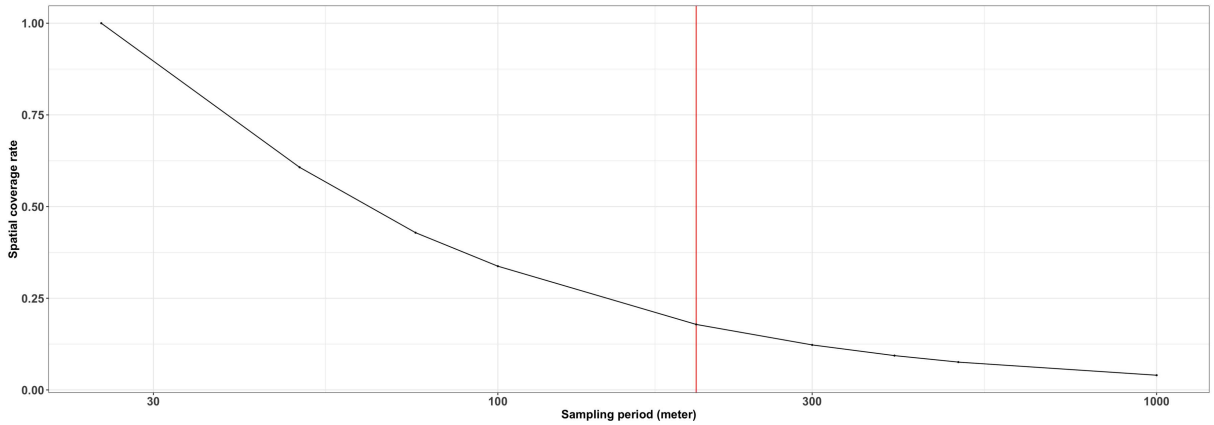


FIGURE 8 – Taux de couverture spatiale en fonction de l'échantillonnage, pour 40 trajets.

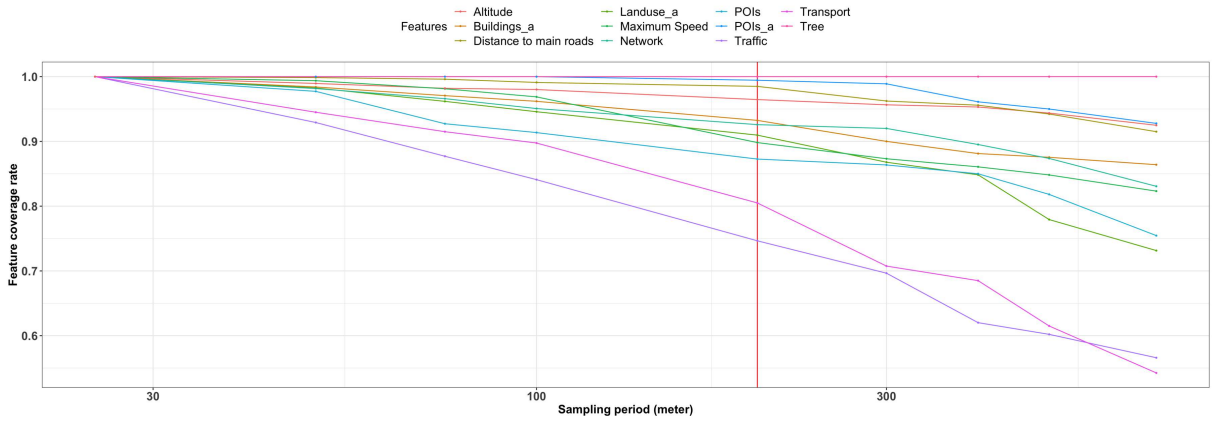


FIGURE 9 – Taux de couverture des méta-données en fonction de l'échantillonnage, pour 40 trajets.

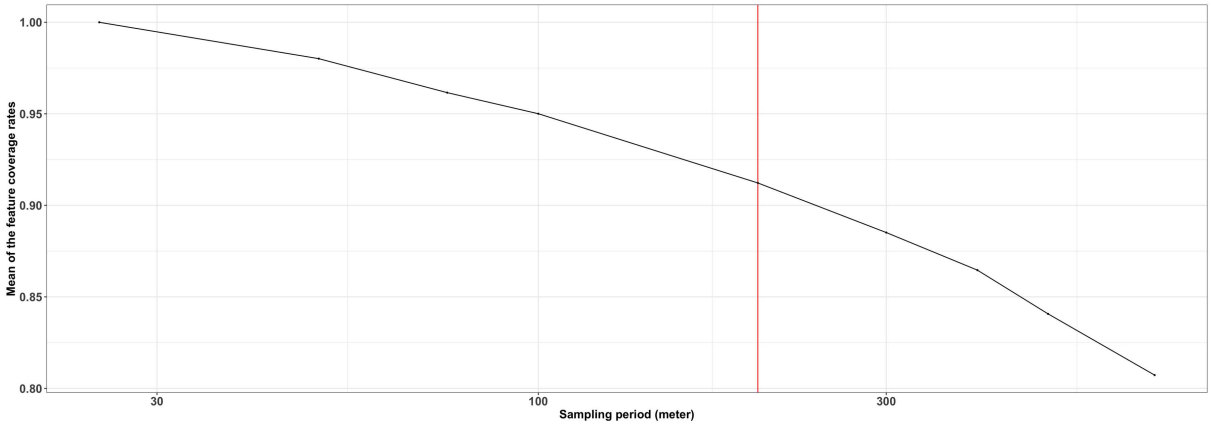


FIGURE 10 – Taux de couverture moyen des méta-données en fonction de l'échantillonnage, pour 40 trajets.

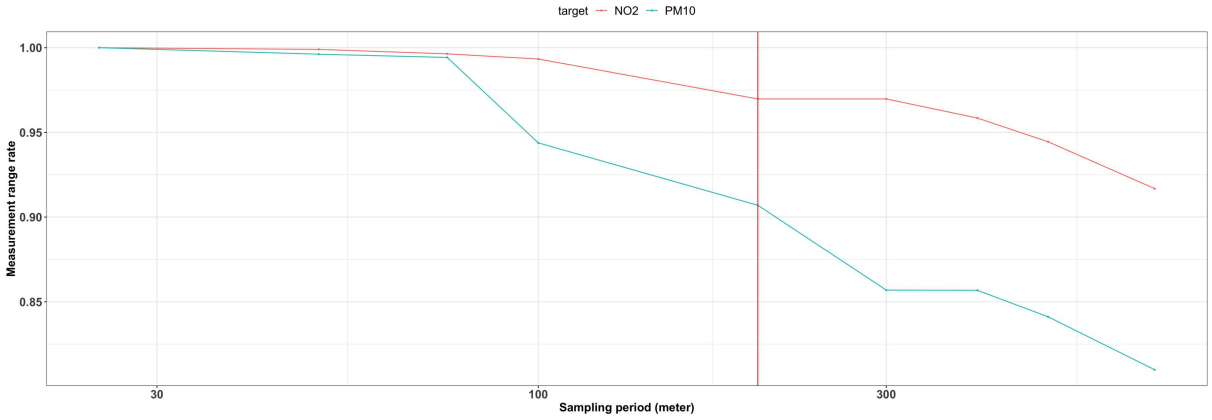


FIGURE 11 – Taux de couverture de l'étendue du polluant en fonction de l'échantillonnage, pour 40 trajets.

6.4 Bruit blanc et perturbation spatiale sphérique

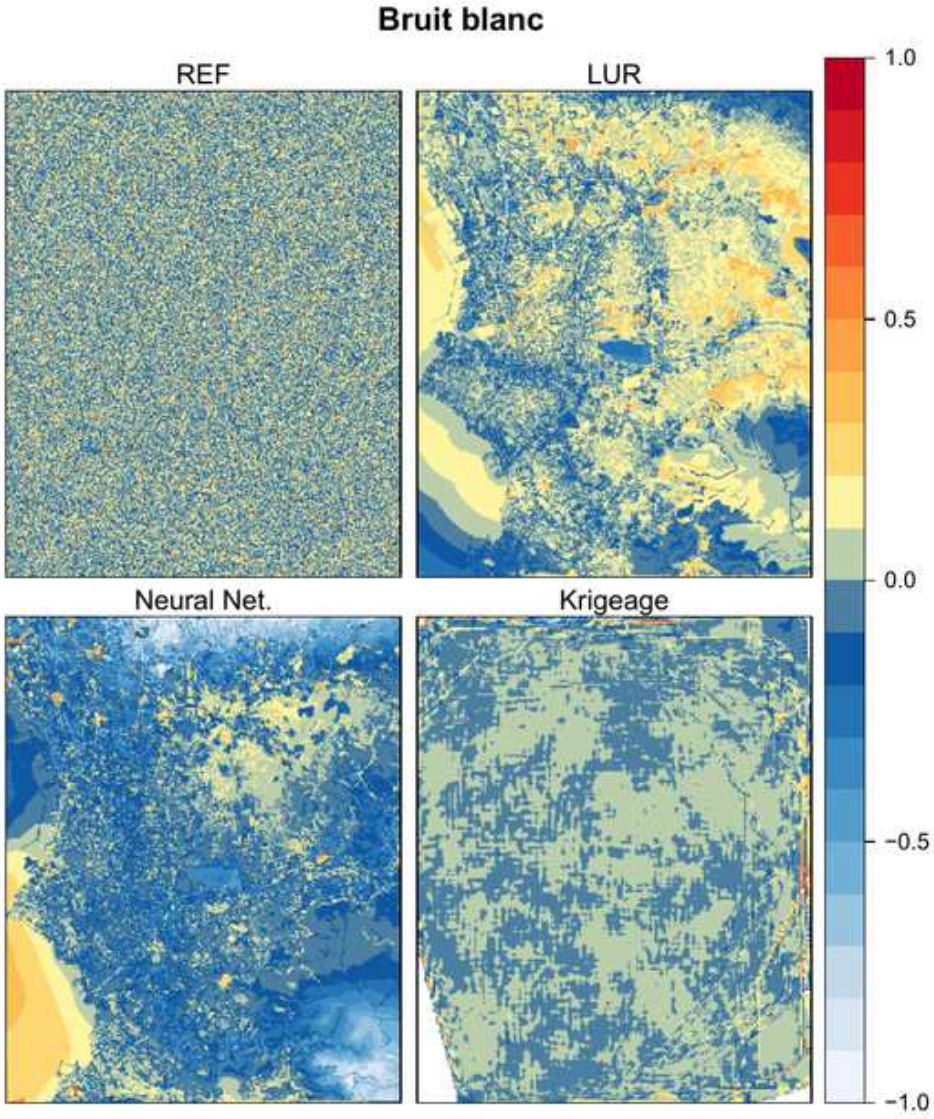


FIGURE 12 – Perturbation introduite sur la carte de référence d'un bruit blanc (REF) et détection de la perturbation par les méthodes de spatialisation. Application pour le NO_2, résultats moyennés pour les 14 jours étudiés. Les perturbations sont normalisées par leur maximum (4,8 $\mu g/m^3$ pour la référence, 0,15 $\mu g/m^3$ pour le LUR, 0,8 $\mu g/m^3$ pour le réseau de neurones, 2,5 $\mu g/m^3$ pour le Krigeage).

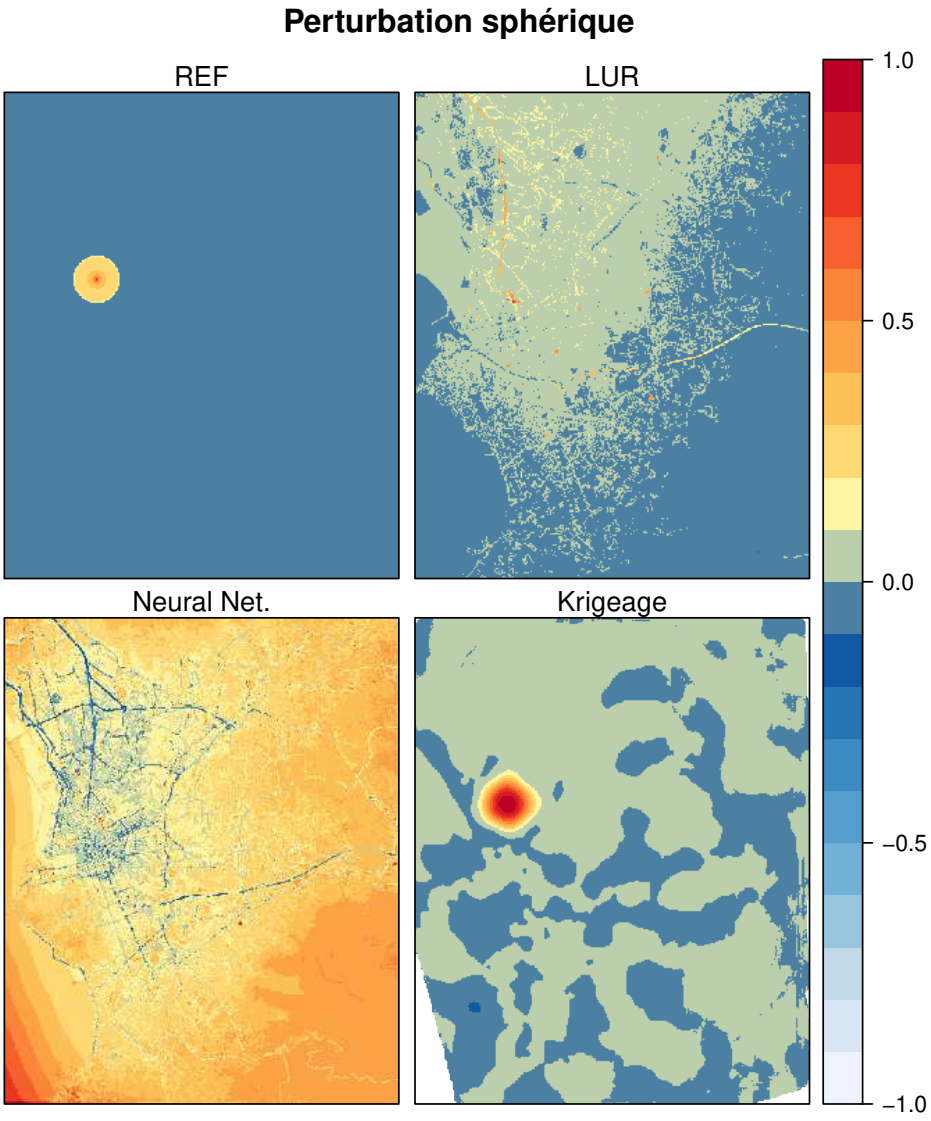


FIGURE 13 – Perturbation sphérique introduite sur la carte de référence (REF) et détection de la perturbation par les méthodes de spatialisation. Application pour le NO_2, résultats moyennés pour les 14 jours étudiés. Les perturbations sont normalisées par leur maximum (444 pour la référence, 116 $\mu g/m^3$ pour le LUR, 23 $\mu g/m^3$ pour le réseau de neurones, 180,4 $\mu g/m^3$ pour le Krigeage).

7 Exemple de séquence d'enregistrement d'un de nos systèmes embarqués

16−01−01 00:00:00 LOG REBOOT SD init 22
16−01−01 00:00:00 LOG GPSINIT LP_MODE_EXIT 1 0
16−01−01 00:00:00 LOG GPSON LP_MODE_EXIT 1 0
16−01−01 00:00:00 LOG MOI GOTO GPS_OSP_SET_MODE_DEGRAD
16−01−01 00:00:00 LOG START 161000110007 v18080301
16−01−01 00:00:00 LOG DESC debug − velo mode − batterie − mox − ble − led
16−01−01 00:00:00 LOG HEAT CO 1 −1 −4
16−01−01 00:00:00 LOG HEAT NOx 1 −1 −4
16−01−01 00:00:00 LOG STATE WORK
16−01−01 00:00:00 MAIN LOOP
16−01−01 00:00:01 BME 100153 3260 40643
16−01−01 00:00:01 MOX 5762 322657 65537 4259840
16−01−01 00:00:02 LOG MOI app_gps_processOSP DEFAULT stopOspMsg
16−01−01 00:00:02 OSPMSG A6005D0000000000
16−01−01 00:00:02 LOG MOI app_gps_processOSP DEFAULT stopOspMsg
16−01−01 00:00:02 OSPMSG A6005D0000000000
16−01−01 00:00:02 BME 100161 3259 40699
16−01−01 00:00:02 MOX 5941 337193 65537 4259840
16−01−01 00:00:03 BME 100158 3260 40759
16−01−01 00:00:03 MOX 5881 339180 65537 4259840
16−01−01 00:00:04 BME 100156 3260 40746
16−01−01 00:00:04 MOX 5904 338848 65537 4259840
16−01−01 00:00:05 BME 100153 3262 40713

[...]

16−01−01 00:06:01 BME 100178 3429 36763
16−01−01 00:06:01 MOX 10375 328396 65537 4259840
16−01−01 00:06:02 LOG MOI app_gps_processOSP DEFAULT stopOspMsg
16−01−01 00:06:02 OSPMSG A6000D0000000000
16−01−01 00:06:02 LOG UpdateRTC
16−01−01 00:06:02 DEBUG 4 0 1 2 0 3862
16−01−01 00:06:02 BME 100183 3428 36785
18−09−21 07:26:21 MOX 10430 326472 65537 4259840
18−09−21 07:26:22 BME 100175 3429 36809
18−09−21 07:26:22 MOX 10508 323607 65537 4259840

[...]

18−09−21 07:33:48 BME 100338 3275 39077
18−09−21 07:33:48 MOX 14200 413589 65537 4259840

18−09−21 07:33:49 BME 100344 3275 39030
18−09−21 07:33:49 MOX 14156 387929 65537 4259840
18−09−21 07:33:50 LOG MOI app_gps_processOSP DEFAULT stopOspMsg
18−09−21 07:33:50 OSPMSG A600380000000000
18−09−21 07:33:50 LOG MOI app_gps_processOSP DEFAULT stopOspMsg
18−09−21 07:33:50 OSPMSG A600380000000000
18−09−21 07:33:50 LOG MOI app_gps_processOSP DEFAULT stopOspMsg
18−09−21 07:33:50 OSPMSG A600380000000000
18−09−21 07:33:50 BME 100346 3271 39108
18−09−21 07:33:50 MOX 14141 407508 65537 4259840
18−09−21 07:33:51 LOG MOI app_gps_processOSP DEFAULT stopOspMsg
18−09−21 07:33:51 OSPMSG A600E10000000000
18−09−21 07:33:51 GPS 436044848 14560377 12323 4
18−09−21 07:33:51 BME 100346 3271 38992
18−09−21 07:33:51 MOX 14068 414817 65537 4259844
18−09−21 07:33:52 LOG MOI app_gps_processOSP DEFAULT stopOspMsg
18−09−21 07:33:52 OSPMSG A6005D0000000000
18−09−21 07:33:52 GPS 436044472 14560599 12906 4
18−09−21 07:33:52 BME 100343 3271 38957
18−09−21 07:33:52 MOX 14318 419354 65537 4259844
18−09−21 07:33:53 GPS 436044332 14560839 13189 4
18−09−21 07:33:53 BME 100349 3271 38876
18−09−21 07:33:53 MOX 14141 417285 65537 4259840
18−09−21 07:33:54 DEBUG 4 0 1 2 1 3928
18−09−21 07:33:54 GPS 436044415 14561019 13248 4
18−09−21 07:33:54 BME 100343 3269 38840
18−09−21 07:33:54 MOX 14229 418111 65537 4259840
18−09−21 07:33:55 GPS 436044456 14561200 13346 4

8 Figures complémentaires pour notre expérience dans la ville de Toulouse (chapitre 4)

8.1 Diagramme de dispersion entre les réponses normalisées de nos capteurs et les concentrations réelles

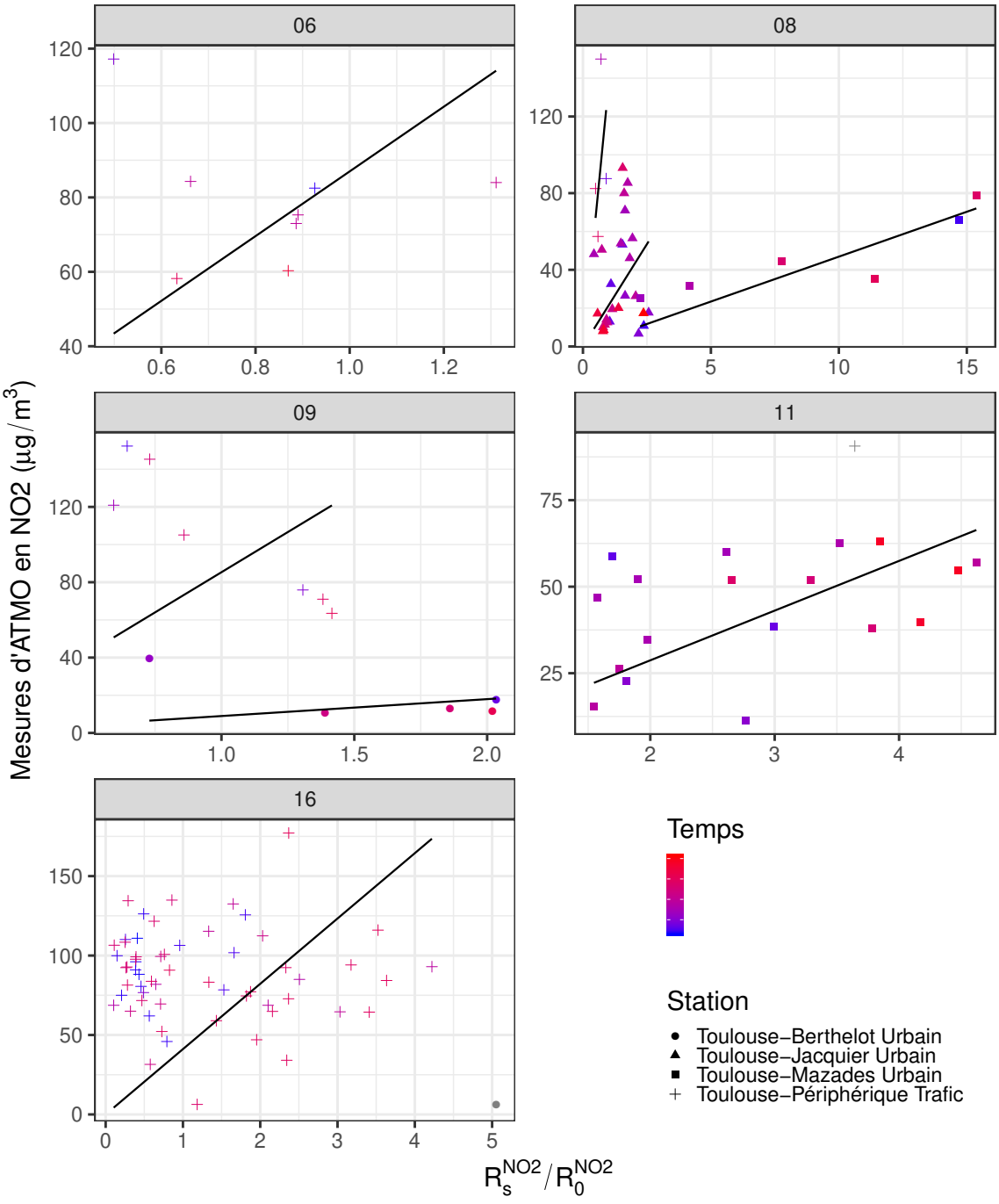


FIGURE 14 – Diagramme de dispersion entre les réponses normalisées des capteurs MiCS-4514 et les concentrations réelles fournies par les stations d'ATMO Occitanie.

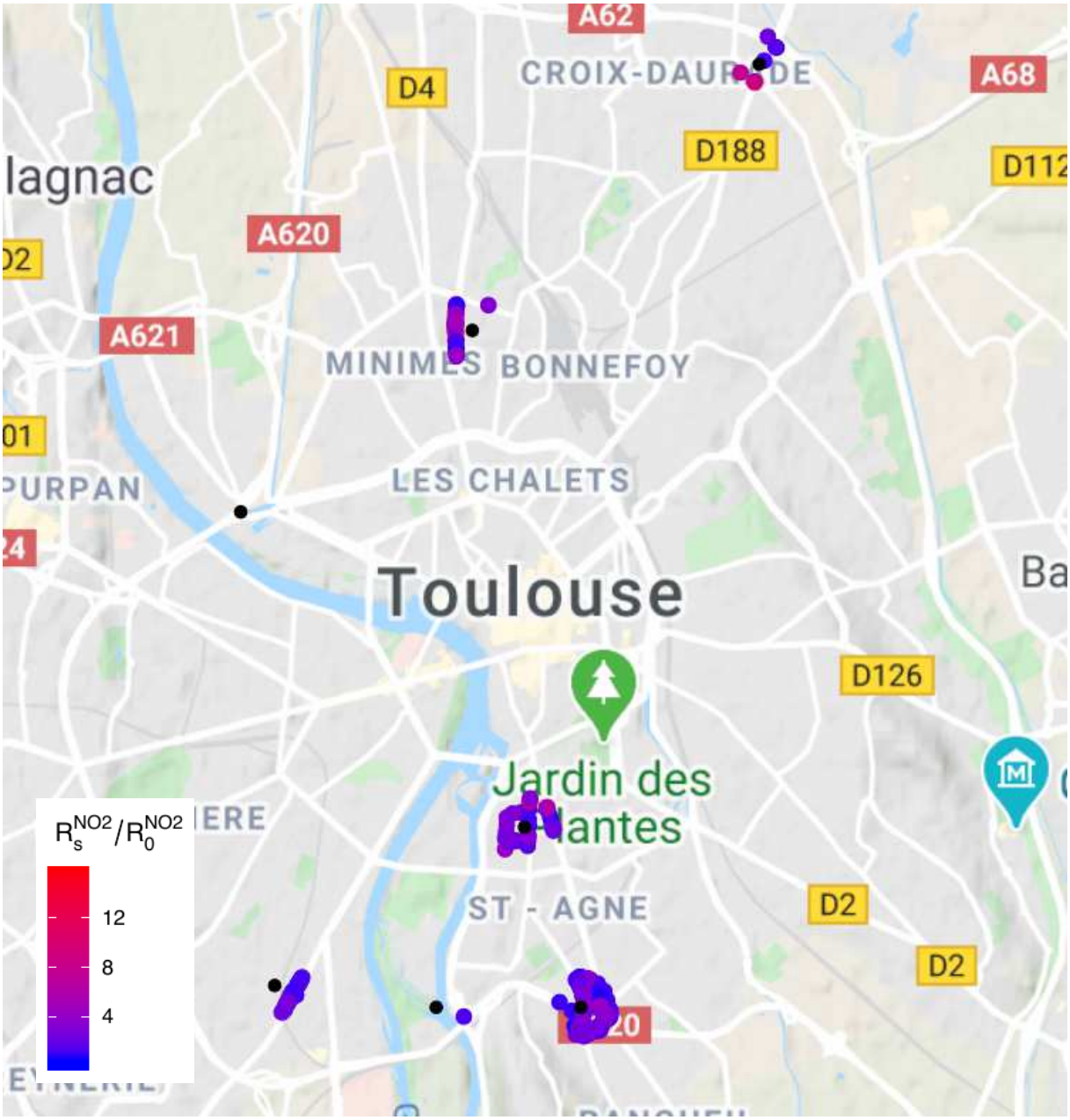
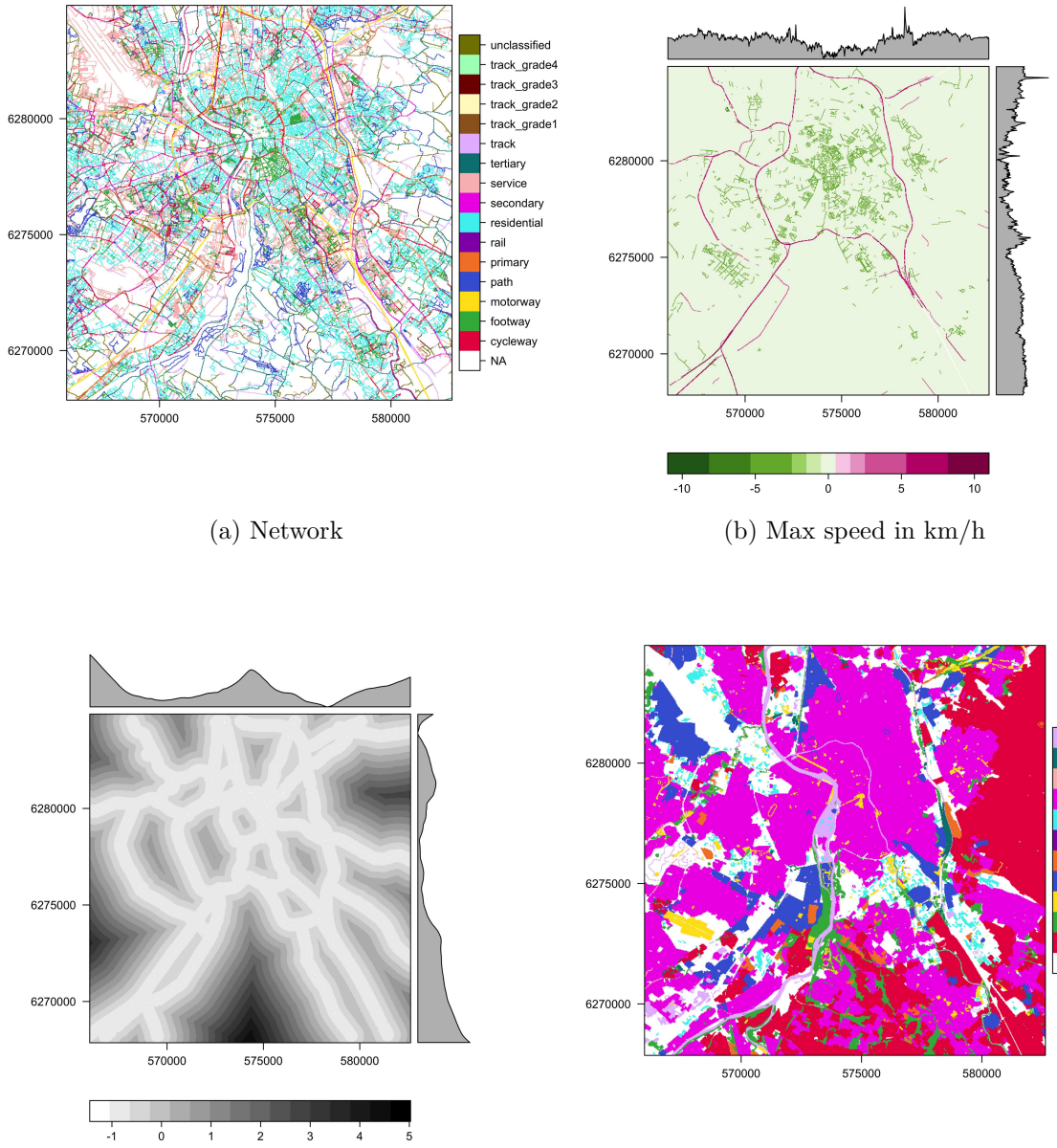


FIGURE 15 – Visualisation des *Rendez-Vous* entre les vélos et les stations d'ATMO Occitanie.

8.2 Variables explicatives pour la ville de Toulouse



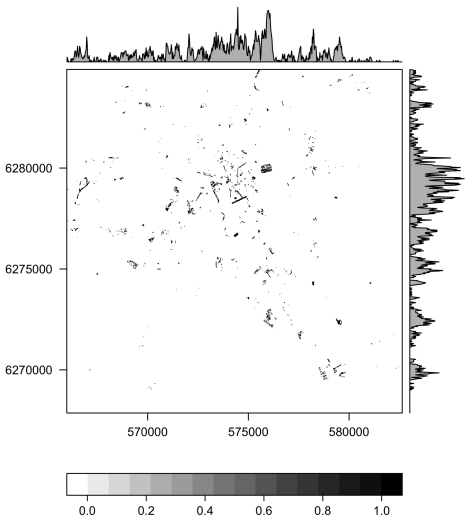
(a) Network

(b) Max speed in km/h

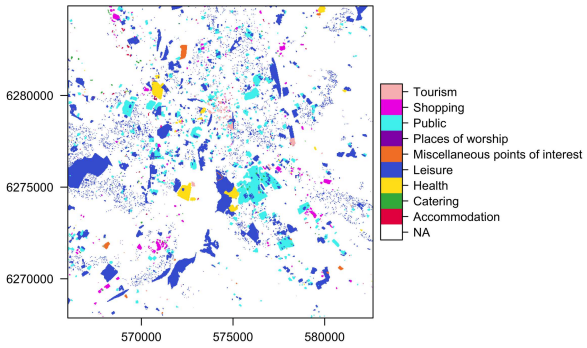
(c) Distance to main road

(d) Landuse

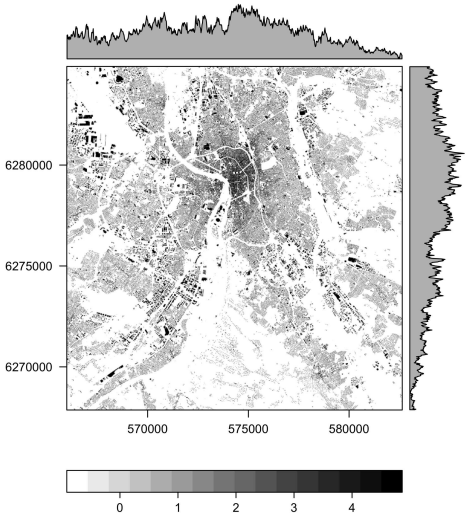
FIGURE 16 – Variables explicatives extraites de la base de données de OpenStreetMap et pré-traitées représentant (a) le réseau routier, (b) la limite de vitesse autorisée, (c) la distance aux principaux axes de circulation, et (d) la carte d'occupation du sol pour la ville de Toulouse.



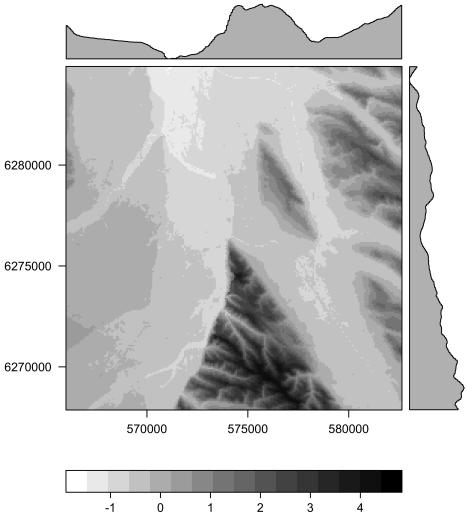
(a) Tree



(b) Point of interest Area



(c) Building Area



(d) Altitude in meters

FIGURE 17 – Même légende que Figure 16 pour (a) la présence d'arbres, (b) les zone de points d'intérêt, (c) la densité de construction et (d) l'altitude du MNT (issue de la base de données de l'IGN).

9 Implémentation informatique : README GitHub

Air pollution spatialization

This R project deals with air pollution spatialization, from external field measurments or external field simulation, and is associated with a thesis which took place at LAAS-CNRS and at Aerology Lab (France).

About The Thesis

The summary of the thesis is as follows: "This thesis takes place in the context of smart cities, where the information processing improves the quality of life. It studies the perception of the environment and especially the perception of air pollution in the city using sensors on bikes. The first chapter introduces the technical and scientific challenges in terms of information collection and modeling applied to aerology. The second chapter presents the design of a fleet of mobile instruments for measuring air pollution. We characterize the shape of the sensor network needed for modeling, on the one hand using the literature and on the other hand using a simulation. The third chapter deals with the development of such an instrument. We have built our instrument around a semiconductor metal oxide micro-sensor (MOx sensor) of NO2 and CO, the MiCs-4514, and evaluated its performance in controlled environments. The fourth chapter presents the two deployments of this instrument in the city of Toulouse in France, first with a bicycle rental association and then with bikers from our laboratory, and the dataset collected. Finally, we estimate the pollution levels in NO2 and CO in the city."

Thesis link : publication pending.

Getting Started

To get a local copy up and running follow these simple steps.

Prerequisites

This project requires R, version 3.4.2 or above.

```
sudo apt-get install r-base
```

Installation

1. Clone the air-pollution-spatialization repository

   ```
   git clone https://github.com/cbertero/air-pollution-spatialization.git
   ```

2. Install R packages from R

   ```
   install.packages(c('automap','caret','data.table','doMC','e1071','geosphere',
   'ggmap','ggplot2','ggthemes','gstat','HDInterval','leaflet','lubridate','magi
   ```

```
ck','magrittr','mapview','metR','mgcv','osmar','plyr','purrr','raster','raste
rVis','Rcpp','reshape2','rgdal','rgeos','RJSONIO','RSNNS','scales','sp','star
s','stringi','stringr','tidyverse','xtable','zoo'))
```

Usage

Two use cases are defined.

In the first case, the spatialization of air pollution is made from an external pollution dataset. This dataset must be in the form of a `data.frame`, saved in `.RDS` format and placed in the `data` directory under the name `Field_Campaign.RDS`. The columns of the `data.frame` are as follows: `longitude`, `latitude`, `value` (concentration of the pollutant). Optionally, if these data correspond to bicycle trips, they can be reconstructed in order to filter out unwanted positions.

In the second case, the spatialization of air pollution is made from simulated bicycle tracks. The generation of the random tracks is managed by this project. It is possible to generate several sets of tracks in order to perform a Monte-Carlo simulation. The simulation of pollution measurements is based on an external simulation of pollution over the study area. The result of this external simulation must be a raster saved in the `data` directory under the name `Field_Simulation.TIFF` (or other extension managed by the `raster` package).

In both cases, it is possible to complete the information used for spatialization by adding metadata over the study area. These data layers must be in the shapefile format and saved in the `data/field` directory. If they are downloaded from GeoFabrick, they are pre-processed automatically.

Finally, the configuration file `project.conf` must be modified accordingly to the use case.

To launch the project, do :

```
Rscript spatialize.R useCase
```

where useCase is either `1` or `2`.

Since the computation may take a long time (especially in the case of the Monte-Carlo simulation), it is think to be process on another machine thanks to Docker. You may modify `host.conf` to perform the project on another machine or in parallel.

Contributing

Any contributions you make are **greatly appreciated**.

1. Fork the Project
2. Create your Feature Branch (`git checkout -b feature/yourFeature`)
3. Commit your Changes (`git commit -m 'Add some feature'`)
4. Push to the Branch (`git push origin feature/yourFeature`)
5. Open a Pull Request

License

Distributed under the MIT License. See `LICENSE` for more information.

Contact

Christophe Bertero - bertero.ch@gmail.com

Project Link: https://github.com/cbertero/air-pollution-spatialization
(https://github.com/cbertero/air-pollution-spatialization)

Acknowledgements

- [PhD advisor : Jean-Francois Léon, Matthieu Roy]
- [Other supervisor : Gilles Tredan]
- Co-funding : NéoCampus (https://www.irit.fr/neocampus/fr/)

Bibliographie

Janssen, S. , Dumont, G. , Fierens, F. et Mensink, C. . Spatial interpolation of air pollution measurements using CORINE land cover data. Atmospheric Environment, 2008. DOI : 10.1016/j.atmosenv.2008.02.043.

Résumé : Cette thèse s'inscrit dans le contexte des « villes intelligentes », où le traitement de l'information améliore la qualité de vie. Elle étudie la perception de l'environnement, et plus particulièrement la perception de la pollution de l'air en ville, à l'aide de capteurs sur vélos. Le premier chapitre introduit les défis techniques et scientifiques, en terme de collecte de l'information et de modélisation, appliqués au domaine de l'aérologie. Le deuxième chapitre s'intéresse à la conception d'une flotte d'instruments mobiles de mesure de la pollution de l'air. Nous caractérisons la forme du réseau de capteurs nécessaire à la modélisation, d'une part à l'aide de la littérature et d'autre part via une simulation. Le troisième chapitre expose notre réalisation d'un tel instrument. Nous l'avons articulé autour d'un micro-capteur à métal-oxyde semi-conducteur (capteur MOx) de NO_2 et CO, le MiCS-4514, et évalué ses performances en milieux contrôlés. Le quatrième chapitre présente les deux déploiements de cet instrument dans la ville de Toulouse, d'abord auprès d'une association de location de vélos puis avec des « vélo-taffeurs » de notre laboratoire, et le jeu de données collecté. Enfin, nous estimons les niveaux de pollution en NO_2 et en CO dans la ville.

Mots clés : Vélo intelligent, Réseau de capteurs et systèmes répartis, Modélisation, Pollution de l'air, Pollution urbaine, Capteur MOx, Mobilité humaine

---

Abstract : This thesis takes place in the context of "smart cities", where the information processing improves the quality of life. It studies the perception of the environment and especially the perception of air pollution in the city using sensors on bikes. The first chapter introduces the technical and scientific challenges in terms of information collection and modeling applied to aerology. The second chapter presents the design of a fleet of mobile instruments for measuring air pollution. We characterize the shape of the sensor network needed for modeling, on the one hand using the literature and on the other hand using a simulation. The third chapter deals with the development of such an instrument. We have built our instrument around a semiconductor metal oxide micro-sensor (MOx sensor) of NO_2 and CO, the MiCS-4514, and evaluated its performance in controlled environments. The fourth chapter presents the two deployments of this instrument in the city of Toulouse in France, first with a bicycle rental association and then with bikers from our laboratory, and the dataset collected. Finally, we estimate the pollution levels in NO_2 and CO in the city.

Keywords : Smart bike, Sensor network and distributed systems, Modeling, Air pollution, Urban pollution, MOx sensor, Human mobility