

Modélisation des flux du dernier kilomètre : reprise et évolution de l'outil Sogaflow pour l'aide à la décision en logistique urbaine durable

Rapport de stage Étude & Recherche
pour le diplôme d'ingénieur de l'École des Ingénieurs de la Ville de Paris

Jules PADE

Septembre 2025

Sous la direction de Reece FISHER, doctorante au Laboratoire Ville, Mobilité, Transport/Chaire Logistics City en CIFRE chez Sogaris.

Notice bibliographique

AUTEUR du mémoire	
NOM	PADE
Prénom	Jules
ORGANISME de stage	
NOM organisme	SOGARIS
NOM, Prénom du maître de stage	FISHER Reece
NOM, Prénom du tuteur EIVP	MOIGNOT Benjamin
ANALYSE	
TITRE du mémoire	Modélisation des flux du dernier kilomètre : reprise et évolution de l'outil Sogaflow pour l'aide à la décision en logistique urbaine durable
TITLE	Last-mile flow modelling: redevelopment and enhancement of the Sogaflow tool for decision-making in sustainable urban logistics
RÉSUMÉ (15 à 20 lignes)	Ce mémoire présente les travaux menés dans le cadre d'un stage de fin d'études au sein de Sogaris, foncière spécialisée dans la logistique urbaine. La mission a porté sur la remise en fonctionnement et l'évolution de Sogaflow, un outil de simulation des flux du dernier kilomètre. Après un diagnostic du projet initial, l'outil a été réactivé, structuré et enrichi de nouvelles fonctionnalités, notamment la comparaison multi-scenarios et l'intégration de calculs environnementaux. Une étude de cas sur le site de Beaugrenelle a permis de confronter les simulations aux données issues d'un bureau d'études, validant la pertinence de l'outil pour des usages opérationnels et prospectifs. Le rapport met en évidence le rôle que peuvent jouer les outils de simulation dans la gouvernance logistique et la planification urbaine, ainsi que les perspectives d'évolution vers une version élargie de Sogaflow.
ABSTRACT	This report presents the work carried out during a final-year internship at Sogaris, a real estate company specialising in urban logistics. The project focused on restarting and upgrading Sogaflow, a tool for simulating last-mile flows. After diagnosing the initial project, the tool was reactivated, restructured, and enhanced with new functionalities, including multi-scenario comparison and integration of environmental impact calculations. A case study on the Beaugrenelle site compared simulation outputs with data from an engineering consultancy, validating the tool's relevance for both operational and prospective applications. The report highlights the role simulation tools can play in logistics governance and urban planning, as well as the development prospects for an expanded version of Sogaflow.
Mots-clés du thésaurus Archirès	Logistique urbaine, dernier kilomètre, simulation, aide à la décision, modélisation, bilan carbone
Keyword thesaurus (anglais, 8 maxi)	Urban logistics, last mile, simulation, decision support, modelling, carbon footprint

Références			
	Nb de pages du mémoire	Annexes (nombre de p.)	Bibliogr. : nb de références
	35	10	14

Remerciements

Je tiens tout d'abord à remercier Adeline Heitz, mon encadrante durant le stage Etude & Recherche, pour sa confiance et pour m'avoir mis en relation avec Reece Fisher, qu'elle co-encadrerait dans le cadre de sa thèse. Un grand merci à Reece Fisher, qui, malgré une expérience limitée en encadrement de stagiaire, a su me guider avec justesse, en m'accordant l'autonomie nécessaire pour développer mes compétences tout en assurant un cadre clair et stimulant.

Je remercie également chaleureusement l'ensemble du service DSEI – Suzanne Friedrich, Clément Roussel et Juliette Berthon – auprès duquel j'ai énormément appris. Cette équipe à taille humaine m'a permis d'approfondir des thématiques essentielles comme la gestion de la donnée, l'innovation, la RSE et la biodiversité, toutes au cœur des enjeux urbains contemporains.

Je tiens aussi à remercier l'ensemble des collaborateurs de Sogaris, avec qui j'ai eu la chance d'échanger, quel que soit leur service ou leur niveau hiérarchique. Ces interactions m'ont offert une vision complète et concrète du fonctionnement d'une foncière logistique.

Un grand merci également à l'équipe du coworking COMET, qui m'a accueilli dans un environnement de travail agréable et fonctionnel durant toute la durée du stage.

Enfin, je remercie sincèrement mon tuteur académique, Benjamin Moignot, pour la richesse de nos échanges, qui m'ont permis de structurer ma réflexion et de prendre du recul tout au long de cette expérience.

Sommaire

Remerciements	1
Liste des illustrations.....	3
Liste des tableaux	3
Glossaire	4
Introduction	5
1. La logistique urbaine : défis, outils et acteurs	6
1.1. Présentation de Sogaris	6
1.2. Contraintes du dernier kilomètre en ville	8
1.3. Positionnement de la simulation comme outil d'aide à la décision	9
1.4. Besoins des acteurs publics, privés et académiques.....	9
2. Reprise et structuration de Sogaflow.....	11
2.1. Diagnostic initial du projet	11
2.2. Réactivation du moteur de simulation.....	11
2.3. Structuration logicielle et conception de l'interface utilisateur.....	13
3. Évolutions fonctionnelles et cas d'usage.....	14
3.1. Nouvelles fonctionnalités intégrées.....	14
3.2. Export des résultats, visualisation cartographique et ajustements	15
3.3. Premier test et cas d'usage	16
4. Simulation sur le cas de Beaugrenelle : retour d'expérience complet	18
4.1. Choix du cas d'étude et intégration des données	18
4.2. Résultats de simulation.....	18
4.3. Comparaison avec l'étude et discussion.....	18
4.4. Enseignements pour le calibrage du modèle	19
5. Positionnement et perspectives	20
5.1. Apports de Sogaflow dans les démarches territoriales.....	20
5.2. Complémentarité avec les outils existants.....	20
5.3. Vers une version V3 ?	21
Conclusion.....	23
Bibliographie/Webographie-Sitographie.....	24

Annexes

Annexe I :	25
Annexe I.1 : Document technique Sogaflow – Source : PADE Jules	26
Annexe I.2 : Interface utilisateur avec résultats – Source : PADE Jules	29
Annexe II :	30
Annexe II.1 : Script start_sogaflow.sh avec commentaires - Source : PADE Jules	32
Annexe II.2 : Exemple de réponse JSON simplifiée renvoyée par l’API VROOM – Source : PADE Jules	33
Annexe II.3 – Exemple de configuration config.py pour un déploiement local de Sogaflow	35

Liste des illustrations

Figure 1: Organisation du comité exécutif de Sogaris	6
Figure 2 : typologie des sites logistiques urbains (Sogaris)	6
Figure 3 : répartition du capital de Sogaris par actionnaire	7
Figure 4 : Sectorisation équilibrée de Nanterre par k-means contraint — Source : Jules Pade (2025)	12
Figure 5 : Ordonnancement optimal par secteur (échantillons) — Source : Jules Pade (2025)	12
Figure 6 : Interface utilisateur de Sogaflow — Source : Jules Pade (2025)	13
Figure 7 : Résultats des simulations dans Sogaflow — Source : Jules Pade (2025)	15
Figure 8 : Résultats globaux d’une simulation Sogaflow — Source : Jules Pade (2025)	16
Figure 9 : Visualisation cartographique d’une tournée — Source : Jules Pade (2025)	16
Figure 10 : Exemple de détour parasite — Source : Jules Pade (2025)	19
Figure 11 : Schéma du processus O+ → Sogaflow — Source : Jules Pade (2025)	21
Figure 12 : Impact du nombre de colis et de véhicules sur le temps de calcul dans VROOM — Source : Jules Pade (2025)	22
Figure 13 : Complexité VROOM en fonction du nombre de points et de véhicules — Source : Jules Pade (2025)	22

Liste des tableaux

Tableau 1 : Comparaison annuelle distances et émissions entre Sogaflow et l’étude Jonction — Source : Jules Pade (2025)	18
---	----

Glossaire

API : Interface de programmation permettant à deux systèmes de communiquer.

Asset management : Gestion d'actifs immobiliers visant à optimiser la valeur, la performance et la rentabilité des sites logistiques sur le long terme.

Backend : Partie serveur et calcul d'un logiciel, qui traite les données, exécute les opérations en arrière-plan et fournit les résultats à l'interface utilisateur.

DAM : Direction Asset Management.

DD : Développement durable, conciliation économie, social et environnement.

DPM : Direction Property Management.

IT : Technologies de l'information (outils, systèmes, réseaux).

K-means constraint : Algorithme de regroupement qui forme des clusters (ou secteurs) en imposant une contrainte d'équilibre sur le nombre de points dans chaque groupe.

Position : Unité de comptage en logistique correspondant à une adresse ou un point de livraison, quelle que soit la quantité de colis.

Property management : Gestion technique, administrative et opérationnelle quotidienne des sites logistiques (maintenance, conformité, relations locataires, etc.).

Service Docker pour OSRM : Instance du moteur open-source OSRM (Open Source Routing Machine) exécutée dans un environnement conteneurisé Docker, utilisée pour calculer les distances et durées de trajets à partir de données géographiques.

Thèse CIFRE : Doctorat financé en partenariat entre entreprise et laboratoire de recherche.

VROOM : Moteur open-source d'optimisation des tournées de véhicules (Vehicle Routing Problem), utilisé pour calculer l'ordre de passage optimal selon les contraintes définies.

Introduction

La logistique urbaine occupe une place croissante dans les enjeux des villes contemporaines. Elle conditionne l'approvisionnement des commerces, des entreprises et des habitants, tout en pesant sur la congestion, l'occupation de l'espace public et les émissions de gaz à effet de serre. L'essor du e-commerce et la fragmentation des flux accentuent encore ces défis, en particulier sur le segment du dernier kilomètre, souvent le plus complexe et le plus coûteux à organiser.

Face à cette situation, la modélisation des flux du dernier kilomètre apparaît comme un levier stratégique. Elle permet de représenter, simuler et comparer différents scénarios logistiques afin d'anticiper leurs effets opérationnels et environnementaux, et d'éclairer la décision publique comme privée.

C'est dans ce contexte qu'a été retenue la problématique suivante : *Comment modéliser les flux du dernier kilomètre pour aider à la décision en matière de logistique urbaine durable ?*

Le terme « modéliser » fait ici référence à la volonté de représenter de manière formalisée les flux logistiques du dernier kilomètre, à travers un outil numérique capable de simuler différents scénarios d'organisation. Il s'agit de structurer des données, de traduire des logiques d'exploitation et d'anticiper les effets de choix logistiques, afin de produire une information exploitable, compréhensible et comparable.

Les « flux du dernier kilomètre » désignent l'ensemble des mouvements de marchandises effectués sur le segment final de la chaîne logistique, depuis un entrepôt ou un site de distribution jusqu'au destinataire, qu'il s'agisse d'un commerçant, d'une entreprise ou d'un particulier. Bien que cette portion soit courte en distance, elle concentre une grande partie des difficultés logistiques en ville : congestion, accessibilité réduite, faible mutualisation, pressions foncières et nuisances diverses.

L'enjeu est donc d'utiliser cette modélisation pour éclairer les arbitrages des acteurs publics comme privés. Cela peut concerner des choix d'implantation, de flotte ou de stratégie de desserte, mais aussi des politiques de planification urbaine ou d'investissement. Enfin, la notion de logistique urbaine durable renvoie à l'ambition de concilier efficacité économique, qualité de service et réduction des externalités négatives, dans une approche systémique et territorialisée.

Ce rapport s'ouvre sur la présentation de l'entreprise d'accueil et du contexte dans lequel s'inscrit le projet. Il se poursuit par la description du travail de reprise et de structuration de l'outil Sogaflow, avant d'exposer les évolutions fonctionnelles et de les illustrer par une étude de cas. Enfin, il se conclut sur le positionnement de l'outil et ses perspectives d'évolution.

1. La logistique urbaine : défis, outils et acteurs

1.1. Présentation de Sogaris

Sogaris est une foncière immobilière spécialisée dans la logistique urbaine, dont l'objectif est de rapprocher les espaces logistiques des centres-villes et de développer un foncier adapté aux contraintes urbaines actuelles. La gouvernance de l'entreprise s'articule autour d'un comité exécutif structuré en pôles fonctionnels et opérationnels (Figure 1), ce qui lui permet de coordonner à la fois les enjeux stratégiques et les actions opérationnelles sur le terrain.



Figure 1: Organisation du comité exécutif de Sogaris

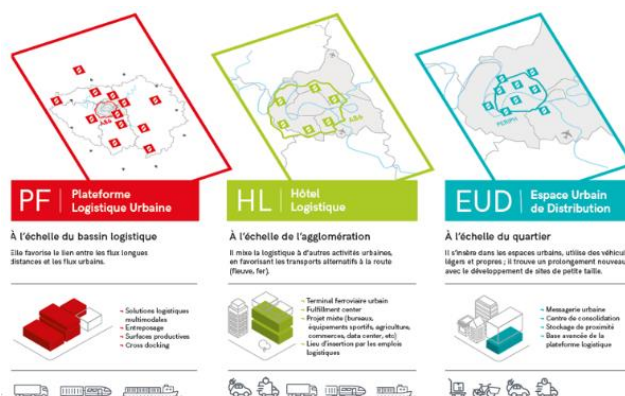


Figure 2 : typologie des sites logistiques urbains (Sogaris)

L'activité de Sogaris se décline en trois grands types de projets (Figure 2), pensés pour structurer la chaîne logistique urbaine depuis les grandes couronnes jusqu'à l'hypercentre : les plateformes logistiques urbaines, les hôtels logistiques et les espaces urbains de distribution. Dans un contexte où le foncier logistique disponible se raréfie en zone dense, la densification des usages permet d'optimiser l'occupation des espaces, notamment grâce aux hôtels logistiques qui accueillent plusieurs fonctions sur un même site.

Sogaris est une société anonyme d'économie mixte (SAEM) à capital majoritairement public (Figure 3). Ce statut hybride lui confère à la fois l'agilité d'un opérateur privé et la capacité de répondre à des objectifs d'intérêt général. Ce statut place la société comme une interface public/privé ce qui lui confère une agilité opérationnelle ainsi qu'une capacité d'innovation intéressante. De plus, l'entreprise est également engagée dans une démarche de Société à Mission, intégrant des objectifs sociaux et environnementaux dans sa stratégie.

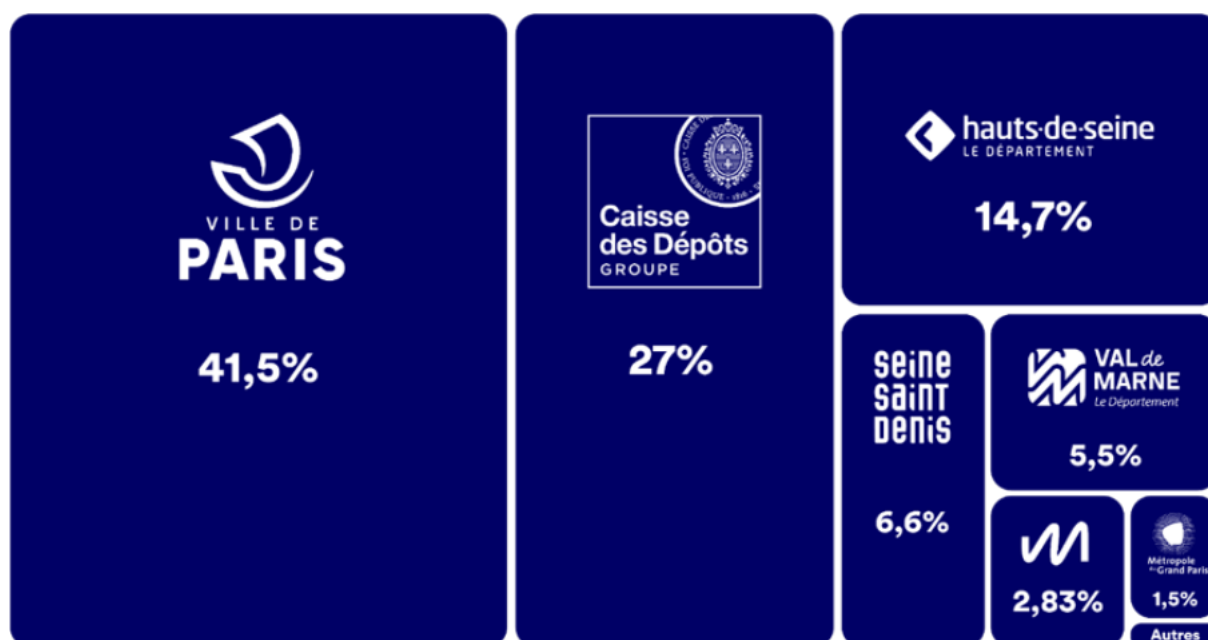


Figure 3 : répartition du capital de Sogaris par actionnaire

En parallèle, Sogaris renforce ses liens avec le monde de la recherche et de l'innovation pour participer au mieux aux transformations de la logistique urbaine. Dans le cadre de sa stratégie en faveur d'une logistique urbaine durable, Sogaris s'investit activement dans la recherche appliquée, en lien étroit avec les milieux académiques et institutionnels. Elle est notamment mécène fondateur de la Chaire Logistics City, soutenue par la Fondation Université Gustave Eiffel et portée par le Laboratoire Ville Mobilité Transport (LVMT), unité mixte de recherche de l'Université Gustave Eiffel et de l'École des Ponts ParisTech. Créée en 2019 et renouvelée en 2023, cette chaire est dédiée aux effets de la logistique urbaine, des entrepôts et du e-commerce sur la ville. Aux côtés de la Région Île-de-France, de la Métropole du Grand Paris et de la Fondation Gustave Eiffel également mécènes de la chaire, Sogaris contribue à une dynamique de recherche-action mobilisant chercheurs, collectivités et acteurs privés.

C'est dans ce cadre que le projet Sogaflow a été créé. Cet outil de modélisation des flux du dernier kilomètre a une double vocation : il sert d'abord aux équipes opérationnelles de Sogaris (bilan carbone, développement, gestion d'actifs immobiliers — asset management — et gestion technique et administrative des sites — property management) pour les aider dans leurs décisions. Tout cela en s'inscrivant dans un projet de recherche porté par la thèse CIFRE de Reece Fisher, doctorat financé par une convention associant Sogaris et un laboratoire académique, intégrée à la Chaire Logistics City, qui fédère chercheurs, institutions et acteurs privés autour des enjeux de la logistique urbaine. Cette double approche, opérationnelle et scientifique, est au cœur du stage présenté dans ce rapport.

1.2. Contraintes du dernier kilomètre en ville

Malgré une distance parcourue bien plus courte que pour les autres phases de la chaîne logistique, la livraison du dernier kilomètre peut représenter 28 % à 53 % des coûts totaux de livraison dans les chaînes d'approvisionnement. (Cardenas et al., 2017 ; Yang et al., 2023 ; Lopez, 2017 ; Wang et al., 2016).

Un des problèmes principaux auquel fait face la livraison au client final est l'encombrement de l'espace public toujours plus important. En effet, la ville est de plus en plus dense, les zones de livraisons se font rare et sont fréquemment occupées par des particuliers à la recherche d'une place de stationnement. Cette recherche de place pour effectuer sa livraison est à multiplier pour chaque position à livrer. C'est un problème commun pour les livreurs mais qui refait surface à chaque tournée.

Une contrainte plus structurelle cette fois est le manque de foncier dédié à la fonction logistique, surtout dans les zones urbaines. La Chaire Logistics City démontre ainsi que moins de 5% du foncier d'activités parisien est alloué à des fonctions logistiques. Sachant qu'à Paris 15-20% des flux sont associés à une activité logistique, cela paraît assez peu. C'est notamment dans ce contexte que Sogaris développe les hôtels logistiques qui permettent l'utilisation du foncier à des fins logistiques tout en densifiant l'usage grâce à d'autres activités en parallèle.

Enfin, une contrainte plutôt conjoncturelle et liée à l'essor du e-commerce est la fragmentation des flux. Selon l'ADEME, le nombre d'arrêt par tournée de livraison a doublé en 10 ans alors même que la charge moyenne par véhicule a diminué dans le même temps. Cela complexifie l'organisation des tournées, réduit la mutualisation, et augmente le taux de livraison échouées, avec 10-15% d'échec en zone dense. Cela représente une baisse de rentabilité non négligeable pour le secteur.

Ces difficultés auxquelles fait face le monde logistique justifient le développement d'outil de simulation et d'aide à la décision. La marge d'erreur faible ainsi que le caractère opérationnel 302 jours par an rendent les expérimentations de terrain très compliquées à mettre en place. La simulation comme outil d'aide à la décision semble donc être un moyen efficace d'aider le monde de la logistique urbaine à optimiser ses opérations et à réduire son empreinte carbone.

1.3. Positionnement de la simulation comme outil d'aide à la décision

Les flux logistiques devenant toujours plus complexe et cachés (*Dablan et al., 2017*), les outils de simulation se placent comme des leviers stratégiques pour aider à prendre des décisions liées à l'organisation, l'aménagement ou la régulation. Leur but ne se résume pas à la représentation de l'existant, mais plutôt à la quantification de l'impact de choix futurs au niveau spatial et organisationnel. La facette logistique restant assez absente des documents de planifications urbaines, ce genre d'outil permet de mettre en lumière et de commencer à mesurer et prévoir ces flux trop souvent oubliés.

Sogaflow a été développé dans cette optique. Cet outil vise à fournir des ordres de grandeur fiables et comparables en fonction du choix de l'emplacement des entrepôts, de la composition de la flotte et de l'organisation des itinéraires. Conçu dès le départ comme un outil opérationnel destiné à aider les développeurs et les clients dans leurs décisions d'implantation et la commercialisation de sites logistiques, il a depuis été réactivé pour soutenir la recherche appliquée, intégré à la thèse CIFRE de Reece Fisher, co-dirigée par Sogaris et l'université Gustave Eiffel. Sogaflow illustre ainsi le rôle que peuvent jouer les modèles de simulation dans la gouvernance logistique : rendre visibles des flux souvent invisibles, tester des hypothèses avant leur mise en œuvre et rapprocher les mondes de l'urbanisme, de la mobilité et de la logistique pour construire une ville plus durable.

1.4. Besoins des acteurs publics, privés et académiques

Le domaine de la logistique urbaine, comme celui de la ville, rassemble énormément d'acteurs ayant tous des rôles bien distincts dans cet écosystème. De fait, les attentes vis-à-vis de Sogaflow sont très hétérogènes. L'intérêt porté à l'outil et son utilité perçue peuvent donc varier énormément selon les acteurs et le contexte.

Sogaflow est une idée issue de la branche DSEI (Direction Stratégie, Engagement, Innovation) de Sogaris dont je fais partie, qui porte une culture d'innovation et de recherche forte. L'outil est vu comme une exploration du possible grâce aux algorithmes et aux données open-source. L'utilisation opérationnelle est une des problématiques abordées, mais l'exploration des possibles reste le sujet principal dans cet environnement. Que ce soient les moteurs d'optimisation utilisés, les indicateurs créés ou les améliorations possibles, ce sont tant d'avancées qui intéressent le milieu de la recherche et notamment la Chaire Logistics City.

Après une réunion de présentation de l'outil à la DD (Direction du Développement) et la DAM (Direction Asset Management) de Sogaris, il était évident que les enjeux étaient ici différents. Pour les acteurs privés et opérationnels, c'est l'absence de friction et la facilité d'utilisation qui va permettre ou non le déploiement de l'outil. Autrement dit, l'interface doit être intuitive, les résultats rapidement exploitables et l'intégration dans leurs processus existants immédiate. L'exploration des possibles n'est pas le sujet ici. Il s'agit dans ce contexte de proposer un outil rapide, fiable, ergonomique. Le rôle de Sogaflow serait donc ici d'incarner une aide simple et sans complication associée pour aider à la décision mais la description outre-mesure de l'outil n'est pas cohérente ici.

Pour finir, les acteurs publics sont ceux avec lesquels j'ai eu le moins d'échanges directs au cours du stage. Ils se positionnent néanmoins à la croisée des mondes de la recherche et du secteur privé. Les résultats produits par Sogaflow peuvent constituer un appui précieux pour éclairer des décisions de régulation ou d'aménagement. L'outil gagnerait toutefois à être adapté à certains contextes ou territoires spécifiques, ce qui impliquerait une phase de développement complémentaire, mais correspond bien aux attentes et aux priorités des collectivités.

Ces besoins variés montrent qu'un même outil peut servir de passerelle entre acteurs publics, opérateurs privés et chercheurs. En rendant les flux visibles et mesurables, il facilite un langage commun autour des enjeux logistiques. C'est précisément dans cette optique qu'a été conçu Sogaflow, dont l'état des lieux initial a constitué la première étape de ce travail.

2. Reprise et structuration de Sogaflow

2.1. Diagnostic initial du projet

Au moment de mon arrivée, Sogaflow sortait d'une longue période d'inactivité, le dernier développement remontant à environ trois ans. Le code source, bien qu'encore présent et complet, ne pouvait plus être exécuté tel quel en raison de la vétusté de ses dépendances logicielles. Certaines bibliothèques appelées dans les scripts avaient disparu ou évolué, rendant plusieurs fonctions obsolètes. Les fichiers de configuration n'étaient plus adaptés à l'environnement actuel, ce qui bloquait toute tentative de lancement sans correction préalable.

Une documentation existait, mais elle se présentait davantage comme un guide de travail en phase exploratoire que comme un manuel opérationnel. Plusieurs étapes clés étaient simplement évoquées ou supposées connues, ce qui obligeait à reconstituer une partie de la logique par lecture directe du code. L'architecture logicielle portait encore la trace de cette dimension expérimentale, avec des modules en cours d'assemblage, des redondances entre scripts et une absence de structuration claire par fonctionnalités.

À côté de cette version locale, une installation de Sogaflow existait sur un serveur interne de Sogaris. Elle avait elle aussi été laissée à l'abandon pendant la même période et présentait les mêmes problèmes de compatibilité. Sa remise en fonctionnement n'a été possible qu'en toute fin de stage, à l'issue d'un travail conjoint avec un membre de l'équipe IT pour résoudre les blocages liés à la configuration réseau et aux services de calcul en arrière-plan, le backend.

Ce contexte rendait indispensable un véritable travail de reprise en main, consistant à comprendre la logique des modules existants, à identifier les blocages techniques et à établir un plan de remise en état avant d'envisager le moindre ajout fonctionnel.

2.2. Réactivation du moteur de simulation

La phase de réactivation s'est concentrée sur les trois briques principales qui composent le cœur fonctionnel de Sogaflow. La première concernait la sectorisation, fondée sur un algorithme de k-means contraint, qui regroupe les adresses en secteurs tout en imposant un nombre équilibré de points par secteur. Dans le modèle actuel, chaque secteur est pris en charge par un seul véhicule, ce qui rend cette étape déterminante pour répartir équitablement la charge de travail et conditionner l'efficacité globale des tournées.

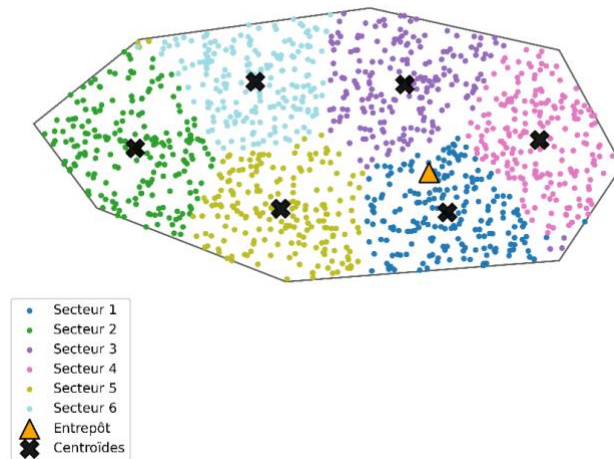


Figure 4 : Sectorisation équilibrée de Nanterre par k-means contraint — Source : Jules Pade (2025)

La deuxième brique portait sur l'optimisation au sens de l'attribution des points de livraison à un véhicule donné. Même si la sectorisation fixe déjà ce cadre, VROOM, moteur open-source d'optimisation de tournées de véhicules, intervient pour préparer les données nécessaires au calcul précis des trajets en tenant compte des contraintes et des distances calculées en amont. Cette partie avait été fortement impactée par les évolutions de l'API VROOM depuis la dernière utilisation de l'outil, rendant indispensable une mise à jour complète des appels et des paramètres.

La troisième brique relevait de l'ordonnancement, c'est-à-dire la détermination de l'ordre de passage optimal à l'intérieur de chaque tournée. Dans ce cadre, il ne s'agit plus de décider quel véhicule dessert quelles adresses, mais de choisir les arrêts pour réduire les distances et durées, en respectant les contraintes définies.

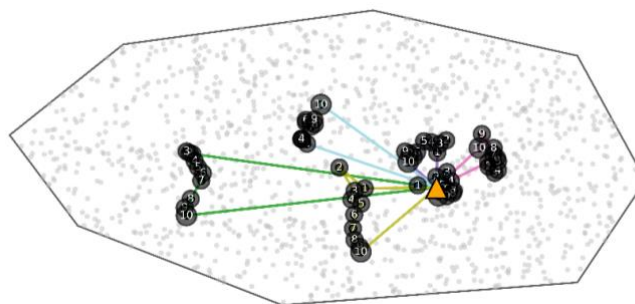


Figure 5 : Ordonnancement optimal par secteur (échantillons) — Source : Jules Pade (2025)

Pour remettre en route ce dispositif, il a fallu corriger les appels aux services Docker pour OSRM, moteur open-source de calcul d'itinéraires exécuté dans un environnement spécifique, chargé du calcul des distances et des durées. De même pour VROOM, responsable des étapes d'optimisation et d'ordonnancement. Les changements d'API, la mise à jour des conteneurs et la configuration réseau ont été passés en revue. Chaque brique a été relancée séparément, puis testée en chaîne afin de valider la continuité du traitement. Ce travail a permis de retrouver une exécution stable et reproductible, indispensable pour envisager des développements supplémentaires.

2.3. Structuration logicielle et conception de l'interface utilisateur

Une fois le moteur interne relancé, l'enjeu a été de repenser la structure du projet pour en faire un outil fiable et évolutif. La réorganisation a consisté à séparer clairement les modules selon leur rôle, en nettoyant le code, supprimant les redondances et harmonisant la nomenclature des scripts. Cette nouvelle architecture rend désormais le projet plus lisible, plus simple à maintenir et prêt à accueillir de nouvelles fonctionnalités.

En parallèle, une interface utilisateur a été développée avec Streamlit pour rendre Sogaflow exploitable sans compétences en programmation. Elle permet de configurer et de lancer une simulation depuis un navigateur, de choisir la zone, l'entrepôt et la flotte, de lancer les calculs de sectorisation et d'optimisation, et de visualiser les résultats sous forme d'indicateurs chiffrés et de cartes interactives. L'export d'un fichier Excel regroupant les paramètres et les résultats synthétiques a été intégré afin de faciliter le partage et l'archivage.

Sogaflow (Simulation logistique)

Mode de simulation ⓘ

- ☐ Étude rapide (1 simulation)
- ☒ Étude poussée (100 simulations)
- ☐ Comparaison d'entrepôts - rapide (1 simulation)
- ☐ Comparaison d'entrepôts - poussée (100 simulations)
- ☐ Comparaison de flottes - rapide (1 simulation)
- ☐ Comparaison de flottes - poussée (100 simulations)

Zone de desserte à simuler (plusieurs codes séparés par une virgule) ⓘ

92000

Nombre de colis à livrer par jour

2000

Choix de la flotte principale (Flotte A)

Mode de définition de la flotte ⓘ

- ☒ Flotte type
- ☐ Flotte personnalisée

Choisir une flotte type

VUL mixte - 50% diesel / 50% électrique (France)

Flotte type sélectionnée : VUL mixte - 50% diesel / 50% électrique (France)

Flotte actuelle (Flotte A) :

	nom	Capacité (nb colis)	Emissions (gCO ₂ /km)	% flotte totale
0	VUL_diesel		90	270
1	VUL_elec		90	60

Sélection de l'entrepôt

Mode de sélection de l'entrepôt

- ☒ Entrepôt Sogaris
- ☐ Saisie manuelle

Choisir un entrepôt Sogaris ⓘ

Beaugrenelle (Espace urbain de distribution)

Lancer la simulation

Figure 6 : Interface utilisateur de Sogaflow — Source : Jules Pade (2025)

Cette remise en état technique et cette refonte structurelle ont permis de repartir sur une base solide. L'outil, redevenu pleinement fonctionnel et plus accessible, pouvait désormais accueillir de nouvelles fonctionnalités pensées pour répondre aux besoins opérationnels et stratégiques de Sogaris et de ses partenaires. C'est cette phase de développement, axée sur l'enrichissement des capacités de simulation et sur la diversification des cas d'usage, qui constitue l'objet de la partie suivante.

3. Évolutions fonctionnelles et cas d'usage

3.1. Nouvelles fonctionnalités intégrées

L'une des principales évolutions apportées à Sogaflow au cours du stage concerne l'élargissement des modes de simulation, afin de rendre l'outil plus flexible et plus proche des besoins réels des utilisateurs. Six configurations sont désormais disponibles :

- **Étude rapide** : une simulation unique sur un jeu de données donné, permettant d'obtenir un résultat immédiat à partir des paramètres choisis.
- **Étude poussée** : cent simulations réalisées avec un tirage aléatoire des adresses, ce qui permet de mesurer la variabilité des résultats et de disposer d'ordres de grandeur plus robustes.
- **Comparaison d'entrepôts – rapide** : évalue deux localisations distinctes sur la base d'une simulation unique, à paramètres de flotte identiques.
- **Comparaison d'entrepôts – poussée** : décline la comparaison précédente sur cent simulations pour affiner l'analyse et neutraliser l'effet d'un tirage unique.
- **Comparaison de flottes – rapide** : compare deux compositions de flotte sur un même territoire et avec un même entrepôt, à partir d'une simulation unique.
- **Comparaison de flottes – poussée** : réalise cette comparaison sur cent simulations pour renforcer la fiabilité des résultats.

Pour rendre la comparaison plus fiable, le système garantissant que chaque scénario repose sur le même jeu d'adresses est le suivant : concrètement, un premier tirage d'adresses est utilisé pour tester successivement l'entrepôt 1 puis l'entrepôt 2 (ou la flotte 1 puis la flotte 2). On passe ensuite au deuxième tirage, appliqué de la même manière, et ainsi de suite. Ce fonctionnement permet d'éliminer l'effet du hasard dans la répartition spatiale des livraisons, afin que les écarts observés reflètent uniquement les paramètres comparés — que ce soit la localisation de l'entrepôt ou la composition de la flotte.

L'outil prend désormais en compte la diversité des configurations possibles grâce à la fonctionnalité de flotte variable, qui permet d'ajuster précisément la composition des véhicules selon le scénario étudié. Chaque type de véhicule est défini par trois paramètres : sa capacité de chargement, son facteur d'émission, et la part de colis qu'il prend en charge sur le total. Cette souplesse rend possible l'évaluation de situations contrastées, allant d'une flotte entièrement thermique à un mix intégrant des véhicules électriques, des poids lourds ou encore des vélos-cargos. En complément, un calcul environnemental a été intégré, exprimé en grammes de CO₂ par kilomètre et par type de véhicule. Les facteurs d'émission retenus, modifiables directement dans le code, ont été choisis pour refléter des ordres de grandeur couramment utilisés dans les analyses de cycle de vie. Leur présence native permet de compléter l'évaluation logistique par une estimation environnementale.

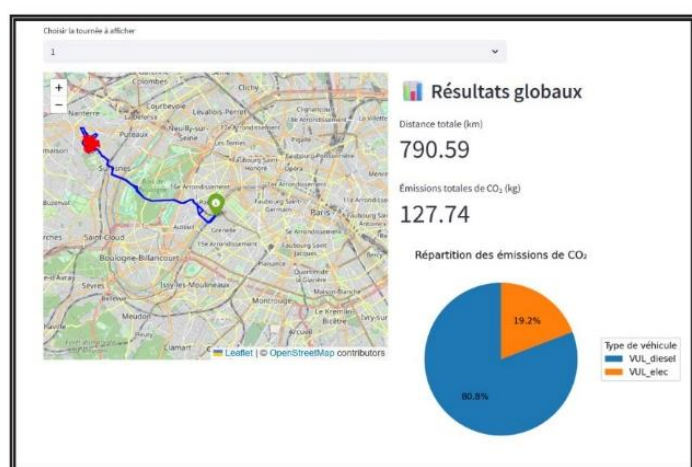
La mise en place de ces fonctionnalités répond à un double objectif : rendre l'outil capable de traiter des cas simples avec rapidité, mais aussi fournir des analyses solides pour les projets nécessitant un niveau de précision plus élevé. Cette souplesse ouvre la voie à des usages variés, allant du test ponctuel pour un opérateur à la simulation prospective pour une collectivité ou un projet de recherche.

3.2. Export des résultats, visualisation cartographique et ajustements

L'outil génère systématiquement un fichier Excel structuré en trois feuilles distinctes, permettant de conserver à la fois les paramètres de simulation et les résultats synthétiques. La première feuille regroupe les informations d'entrée : territoire étudié, nombre de colis, localisation précise de l'entrepôt (nom, latitude et longitude). La seconde est dédiée à la flotte de véhicules utilisée, détaillant pour chaque type de véhicule sa capacité, son facteur d'émission en grammes de CO₂ par kilomètre, et le pourcentage de colis qu'il prend en charge sur l'ensemble des livraisons. Enfin, la troisième feuille rassemble les résultats de simulation, avec pour chaque itération la distance totale parcourue (en kilomètres), les émissions totales de CO₂ (en kilogrammes) et le nombre de tournées générées.

La visualisation cartographique est également intégrée dans l'interface, mais uniquement pour les simulations rapides. Elle permet de vérifier la cohérence de la sectorisation et des trajets avant de lancer une simulation poussée sur 100 tirages. Les indicateurs clés — distance totale, durée cumulée et émissions — sont affichés directement dans l'interface pour faciliter une première lecture sans devoir ouvrir le fichier Excel. Cette double approche, combinant export complet et restitution immédiate, permet à la fois une analyse rapide et une exploitation approfondie des résultats.

Résultat simulation simple



Résultat simulation poussée

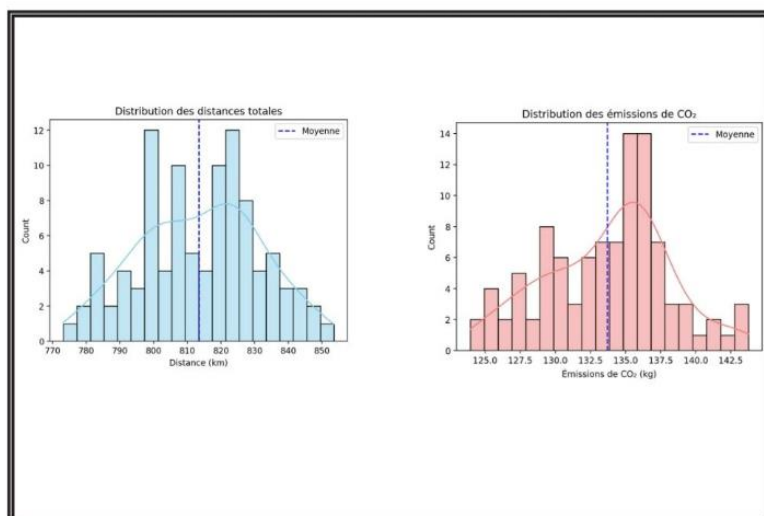


Figure 7 : Résultats des simulations dans Sogaflow — Source : Jules Pade (2025)

Enfin, des ajustements successifs ont été réalisés pour améliorer l'accessibilité et la stabilité de l'outil. Ces modifications ont notamment porté sur l'organisation des exports, la clarté des cartes et la fiabilité des calculs, afin de rendre les simulations plus fluides et les résultats plus exploitables. Cette logique d'exports détaillés, de visualisation cartographique et de contrôle des paramètres en amont prépare le terrain pour les tests grandeur nature. C'est dans ce cadre qu'un premier jeu de simulations a été mené, afin d'illustrer concrètement le fonctionnement de l'outil et d'évaluer ses résultats dans des conditions proches d'un cas réel.

3.3. Premier test et cas d'usage

Pour donner un aperçu concret de ce que produit Sogaflow avant d'entrer dans l'étude de cas, une simulation poussée a été réalisée sur 100 tirages d'adresses, sans comparaison d'entrepôts ni de flottes. Le territoire, le nombre de colis et la composition de flotte ont été fixés, puis chaque tirage a été traité indépendamment, permettant de lisser les effets liés à la répartition aléatoire des points de livraison. Ce mode fournit ainsi des ordres de grandeur plus stables et représentatifs des conditions moyennes simulées.

Les résultats sont restitués sous trois formats complémentaires. D'abord dans l'interface Streamlit, où s'affichent les indicateurs clés — distance totale parcourue, durée cumulée des tournées et émissions totales de CO₂ — pour une lecture immédiate. *(Capture d'écran à insérer ici : affichage des indicateurs dans Streamlit)*

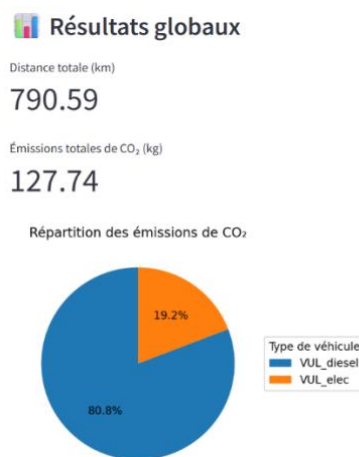


Figure 8 : Résultats globaux d'une simulation Sogaflow — Source : Jules Pade (2025)

Ensuite, via la visualisation cartographique disponible dans le mode rapide, utilisée en amont pour valider la cohérence de la sectorisation et des trajets avant le passage en simulation poussée.

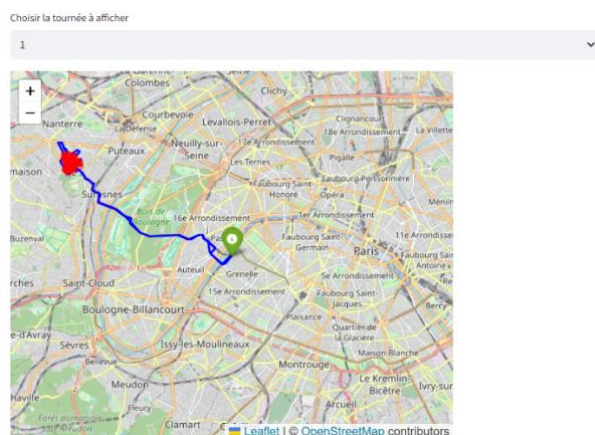


Figure 9 : Visualisation cartographique d'une tournée — Source : Jules Pade (2025)

Enfin, dans l'export Excel généré automatiquement, qui regroupe l'ensemble des paramètres d'entrée, les caractéristiques détaillées de la flotte et les résultats consolidés sur l'ensemble des tirages. Ce fichier permet d'explorer finement les résultats et de conserver une trace exploitable des simulations.

Cet exemple, bien qu'entièrement arbitraire, sert de référence pour illustrer le type de sortie que l'on peut attendre de Sogaflow. Les mêmes mécanismes seront mobilisés dans l'étude de cas, cette fois appliqués à une situation réelle et confrontés aux données observées sur le terrain.

4. Simulation sur le cas de Beaugrenelle : retour d'expérience complet

4.1. Choix du cas d'étude et intégration des données

Le site de Beaugrenelle a été retenu comme terrain de validation. Ce choix se justifie par l'existence d'une étude approfondie menée par le bureau d'études Jonction, reconnue pour la précision de ses hypothèses et la robustesse de ses résultats. Cette base de travail offrait un cadre idéal pour mettre à l'épreuve Sogaflow avec un jeu de données documenté.

Les paramètres intégrés dans Sogaflow reprennent fidèlement les hypothèses retenues dans le rapport Jonction :

- **Zone desservie** : ensemble des secteurs de distribution associés au site de Beaugrenelle.
- **Nombre de livraisons** : estimation annuelle équivalente à l'activité observée dans l'étude.
- **Flotte** : exclusivement composée de véhicules utilitaires légers (VUL).
- **Capacité par VUL** : 70 colis/jour.
- **Nombre de jours d'activité** : 302 jours par an.
- **Facteur d'émission** : valeur représentative des ordres de grandeur couramment utilisés en analyse de cycle de vie (gCO₂/km).

4.2. Résultats de simulation

En appliquant ces paramètres dans une simulation poussée (100 tirages d'adresses), Sogaflow a produit des résultats moyens de 270 860 km parcourus par an et 67 715 kg de CO₂ émis. Ces chiffres sont issus d'une chaîne complète de traitement comprenant la sectorisation en k-means contraint, l'optimisation des trajets avec VROOM et l'estimation environnementale par type de véhicule.

4.3. Comparaison avec l'étude et discussion

L'étude Jonction indique 203 646 km et 46 095 kg de CO₂ pour le même périmètre et les mêmes hypothèses de base. L'écart observé atteint +33 % sur les distances et +47 % sur les émissions. Malgré cette différence, on reste dans le bon ordre de grandeur, ce qui valide la pertinence globale de l'outil pour une utilisation comparative ou prospective.

Valeur annuelle	Distance (km)	Emission (KgCo2)
Sogaflow	270860	67715
Etude Jonction	203646	46095
Écart absolu	67214	21620
Écart relatif	33%	47%

Tableau 1 : Comparaison annuelle distances et émissions entre Sogaflow et l'étude Jonction — Source : Jules Pade (2025)

Ces écarts tiennent en grande partie à la rigidité de la sectorisation et des trajets simulés. Chaque secteur étant assigné à un seul véhicule, certaines configurations incluent des détours qui ne seraient pas réalisés dans la réalité, et qui allongent mécaniquement la distance totale. De plus, l'absence de réallocation inter-secteurs et la prise en compte stricte des zones définies peuvent limiter l'efficacité globale des tournées.

Toutefois, dans le cadre du module de comparaison de Sogaflow, ces imperfections d'optimisation ont un impact réduit : le fait de comparer deux scénarios sur un même jeu d'adresses et avec les mêmes contraintes permet de neutraliser les biais liés à la répartition spatiale ou aux détours spécifiques à un tirage. Les écarts observés sont donc davantage liés aux paramètres testés (localisation, composition de flotte) qu'aux imperfections du moteur d'optimisation.



Figure 10 : Exemple de détour parasite — Source : Jules Pade (2025)

4.4. Enseignements pour le calibrage du modèle

Ce test grandeur nature confirme la capacité de Sogaflow à reproduire l'ordre de grandeur d'une situation réelle à partir de données ouvertes et de traitements standardisés. La comparaison avec l'étude Jonction montre que, même avec des logiques d'optimisation perfectibles, l'outil fournit des résultats suffisamment robustes pour alimenter une réflexion stratégique. La validation par un cas réel permet d'identifier les points d'amélioration prioritaires, comme l'assouplissement de la sectorisation, l'introduction de mécanismes de réaffectation inter-secteurs et l'harmonisation des facteurs d'émission avec les référentiels utilisés par les bureaux d'études.

Une seconde validation avait été envisagée, notamment sur le site de La Chapelle, mais le refus de transmission des données opérationnelles par l'exploitant n'a pas permis de mener à bien ces projets. Cette étude de cas reste donc, à ce stade, la seule référence terrain pour évaluer le comportement du modèle et guider ses évolutions futures.

5. Positionnement et perspectives

5.1. Apports de Sogaflow dans les démarches territoriales

Sogaflow se positionne comme un outil de simulation capable de traiter rapidement des scénarios logistiques à l'échelle locale, tout en intégrant des paramètres opérationnels et environnementaux. Sa force réside dans la possibilité de générer, en quelques minutes, des ordres de grandeur exploitables par des décideurs, que ce soit pour comparer deux localisations d'entrepôt, tester différentes compositions de flotte ou évaluer l'effet d'un changement d'organisation.

Pour les collectivités, il constitue un moyen concret de rendre visibles des flux qui restent souvent invisibles dans les documents de planification. En produisant des indicateurs tels que les distances totales parcourues, les durées de tournée ou les émissions de CO₂, il apporte des éléments tangibles pour appuyer des choix d'aménagement ou de régulation, par exemple dans le cadre d'un plan climat, d'un schéma directeur logistique ou d'un projet de réaménagement urbain.

Sogaflow ne prétend pas remplacer les études de terrain détaillées, mais plutôt fournir une base chiffrée et neutre pour initier ou orienter la discussion entre acteurs. La souplesse de l'outil, capable de s'adapter à différents périmètres et à diverses hypothèses, permet de tester rapidement des scénarios multiples sans mobiliser de moyens lourds. Associé au module de comparaison, il offre également une lecture plus robuste des écarts entre deux configurations, en limitant l'influence des aléas de tirage grâce à un protocole identique pour chaque scénario.

En ce sens, Sogaflow s'inscrit pleinement dans la logique d'outils d'aide à la décision territoriale : il ne se limite pas à décrire une situation existante, mais accompagne la construction de politiques logistiques plus efficaces et plus durables, en rapprochant les préoccupations d'urbanisme, de mobilité et d'environnement.

5.2. Complémentarité avec les outils existants

Sogaflow, en l'état, fonctionne de manière autonome. Néanmoins, sa conception ouvre la voie à des liens avec d'autres outils en développement ou en expérimentation. Parmi eux figure **O+**, un projet expérimental porté par ma tutrice dans le cadre de sa thèse, qui vise à estimer les flux de marchandises à l'échelle d'un projet donné. L'idée explorée est d'utiliser directement ces estimations comme entrées pour Sogaflow, afin de chaîner les deux démarches : d'abord quantifier les flux liés à un aménagement ou à une opération urbaine, puis simuler leur distribution finale dans l'espace et le temps.

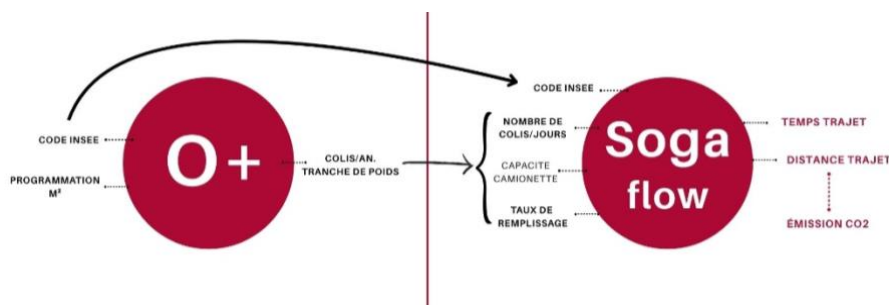


Figure 11 : Schéma du processus O+ → Sogaflow — Source : Jules Pade (2025)

Cette articulation n’a pas encore été mise en œuvre, mais elle constituerait un prolongement naturel pour enrichir les scénarios testés dans Sogaflow et les relier à des situations réelles.

En parallèle, des pistes de réflexion existent pour connecter l’outil à des données issues d’acteurs comme Wintics ou Altaroad, qui pourraient fournir des informations fines sur les usages de voirie ou sur le trafic poids lourds. Là encore, il s’agit d’une perspective à moyen terme, qui demanderait un travail de cadrage méthodologique et technique avant toute intégration effective. Une telle connexion permettrait d’ancrer les simulations dans des données de terrain actualisées, rendant les scénarios plus réalistes et mieux adaptés aux spécificités locales.

En somme, si Sogaflow couvre déjà la partie simulation du dernier kilomètre, son interfaçage avec d’autres outils pourrait élargir considérablement son champ d’application. C’est précisément cette ouverture vers de nouvelles fonctionnalités et de nouveaux cas d’usage qui alimente aujourd’hui la réflexion autour d’une éventuelle version V3 de l’outil.

5.3. Vers une version V3 ?

Plusieurs pistes d’évolution se dessinent pour une prochaine version de Sogaflow, afin d’enrichir ses usages et de renforcer sa pertinence opérationnelle.

La première concerne la mise sur serveur et l’ouverture multi-utilisateurs, qui permettrait de rendre l’outil accessible en ligne à différents profils (collectivités, opérateurs, chercheurs), tout en assurant une gestion centralisée des calculs et des données. Ce déploiement impliquerait toutefois de résoudre certaines contraintes techniques, notamment liées à la stabilité, à la sécurité et au suivi des performances.

Une autre évolution majeure porterait sur l’intégration de la cyclologistique, explorée avec la coopérative Les Cargonautes. L’enjeu serait de mieux orienter les vélos-cargos vers les zones les plus pertinentes, par exemple en tenant compte de la congestion ou de la proximité des points de livraison, afin de maximiser leur efficacité dans la répartition des flux.

Le développement futur pourrait également inclure la gestion multi-entrepôts et les fenêtres horaires, pour modéliser des scénarios plus proches des contraintes réelles d’exploitation. Ces ajouts viendraient enrichir la capacité de l’outil à traiter des cas complexes, mais augmenteraient aussi la charge de calcul.

L'analyse des performances du moteur VROOM montre que le nombre de colis est le facteur ayant l'impact le plus fort sur le temps de traitement, avec une croissance exponentielle au-delà d'un certain seuil (Figure 12). À l'inverse, le nombre de véhicules a un effet marginal sur la complexité. C'est précisément pour maîtriser cette variable que la sectorisation joue un rôle clé : elle ne sert pas uniquement à affecter un véhicule par secteur, mais aussi à limiter le nombre de colis traités simultanément par l'algorithme, ce qui évite des temps de calcul prohibitifs.

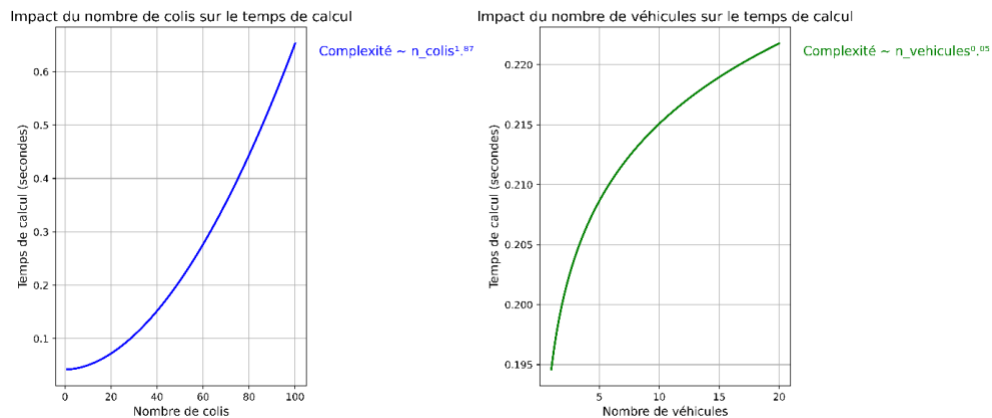


Figure 12 : Impact du nombre de colis et de véhicules sur le temps de calcul dans VROOM — Source : Jules Pade (2025)

Cette contrainte explique pourquoi un déploiement à grande échelle — par exemple à l'échelle départementale, question déjà posée lors de présentations — reste difficilement envisageable dans l'état actuel. La montée en charge ferait exploser les temps de calcul et nuirait à la réactivité de l'outil, sauf à repenser profondément son architecture ou à introduire des méthodes d'approximation.

Les représentations tridimensionnelles (Figure 13) confirment cette tendance : la complexité dépend avant tout du nombre de colis à traiter, bien plus que du nombre de véhicules. Ces éléments rappellent que toute évolution fonctionnelle devra conserver un équilibre entre précision des résultats et temps de réponse, pour garantir que l'outil reste exploitable dans un cadre opérationnel.

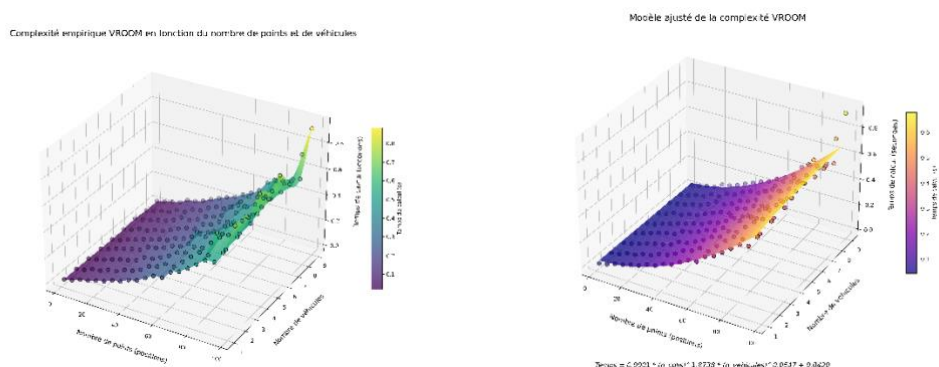


Figure 13 : Complexité VROOM en fonction du nombre de points et de véhicules — Source : Jules Pade (2025)

Conclusion

Ce stage a été bien plus qu'un travail de développement technique sur un outil de simulation. Il m'a permis de prendre la mesure de ce que peut représenter la donnée lorsqu'elle est bien pensée, bien structurée et partagée. La donnée n'est pas seulement une suite de chiffres ou de coordonnées : c'est un langage commun, un socle sur lequel des acteurs aux logiques parfois très différentes peuvent se comprendre et agir ensemble.

En travaillant sur Sogaflow, j'ai vu comment un outil, lorsqu'il est robuste et documenté, peut dépasser son rôle initial pour devenir un lieu de rencontre. Un lieu où les préoccupations opérationnelles des exploitants, les objectifs stratégiques des décideurs publics et les approches méthodiques des chercheurs peuvent se rejoindre. Cette mise en dialogue n'efface pas les divergences, mais elle permet de les aborder sur des bases solides, chiffrées et transparentes.

Ce que je retiens surtout, c'est qu'au-delà de la technique, un outil comme Sogaflow devient une passerelle. Il relie des univers qui se parlent peu, il traduit des besoins en indicateurs, des intuitions en scénarios mesurables, et il ouvre des perspectives d'action partagées. Dans un contexte urbain où les décisions doivent être prises de plus en plus rapidement et sur des enjeux complexes, cette capacité à fédérer autour d'éléments tangibles est précieuse.

En tant qu'élève fonctionnaire, sur le point de m'engager pleinement dans l'action publique, je sais que cette compétence pourra s'appliquer bien au-delà de la logistique urbaine. La capacité à créer, interpréter et mettre en partage la donnée pourra servir dans de nombreux domaines : aménagement, mobilité, environnement, ou encore gestion des services urbains.

Ce stage a renforcé ma conviction que la donnée, lorsqu'elle est produite avec rigueur et pensée pour être comprise par tous, peut devenir un véritable outil de dialogue et de décision collective. C'est cette logique, celle d'un rassemblement autour de faits partagés et d'objectifs communs, que je souhaite continuer à développer tout au long de ma carrière.

Bibliographie/Webographie-Sitographie

ADEME. (2022). *Données et études sur la logistique urbaine et les émissions de CO₂*. Agence de la transition écologique. <https://librairie.ademe.fr>

Altaroad. (2024). *Données de voirie pour la logistique urbaine*. <https://altaroad.com>

Cardenas, I., Beckers, J., & Vanelslender, T. (2017). E-commerce last-mile in Belgium: Developing an external cost delivery index. *Research in Transportation Business & Management*, 24, 123-129. <https://doi.org/10.1016/j.rtbm.2017.07.001>

Chaire Logistics City. (2021). *Logistique urbaine et foncier à Paris : état des lieux et enjeux*. Université Gustave Eiffel.

Dablanc, L., Morganti, E., Arvidsson, N., Browne, M., Saidi, N., & Salanova, J. (2017). The rise of on-demand “Instant Deliveries” in European cities. *Supply Chain Forum: An International Journal*, 18(4), 203-217. <https://doi.org/10.1080/16258312.2017.1377511>

Granger, T. (2025, janvier). Livraison du dernier kilomètre : un défi logistique majeur en 2025. *Transports Granger – Blog Logistique et Transport*. <https://transportsgranger.com/blog/logistique-et-transport/livraison-dernier-kilometre/>

Jonction. (2023). *Rapport d'évaluation logistique Beaugrenelle*. Étude commandée par Sogaris.

Lopez, D. (2017). Last-mile delivery costs and logistics challenges in retail supply chains. *Journal of Business Logistics*, 38(4), 254-270. <https://doi.org/10.1111/jbl.12163>

O+. (2024). *Outil de modélisation des flux logistiques à l'échelle d'un projet*. Expérimentation, Direction RSE & Innovation, Sogaris.

Sogaris. (2024). *Présentation institutionnelle et données internes*. Direction RSE & Innovation, Sogaris.

Université Gustave Eiffel. (2024). *Thèse CIFRE – Modélisation et simulation des flux logistiques urbains*. Laboratoire de recherche en logistique urbaine.

Wang, X., Zhan, J., & Hu, H. (2016). Cost structures of last-mile delivery in e-commerce. *International Journal of Physical Distribution & Logistics Management*, 46(6-7), 672-691. <https://doi.org/10.1108/IJPDLM-09-2015-0224>

Wintics. (2024). *Solutions de mesure et d'analyse du trafic urbain*. <https://wintics.com>

Yang, X., Lin, P., & Li, S. (2023). Sustainable last-mile delivery: Operational strategies and environmental implications. *Transportation Research Part E: Logistics and Transportation Review*, 165, 102903. <https://doi.org/10.1016/j.tre.2022.102903>

Annexe I :



Document Technique – Déploiement de Sogaflow V2

1. Architecture logicielle

Sogaflow V2 est un outil de simulation logistique urbaine. Il combine une interface Streamlit (Python), un moteur d'optimisation de tournées (VROOM) et un moteur de calcul d'itinéraires (OSRM).

L'interface Streamlit est exécutée localement sous Windows (via Anaconda), tandis que les services OSRM et VROOM sont lancés sous Ubuntu (via WSL) dans des conteneurs Docker. La communication entre composants s'effectue via HTTP (localhost).

2. Pipeline complet

1. L'utilisateur configure la simulation via l'interface Streamlit (choix de la zone, entrepôt, flotte...)
2. Les adresses sont filtrées via le module ``zone_desserte``
3. La sectorisation est réalisée via ``k-means-constrained``
4. Les livraisons sont regroupées en tournées via ``reallocation_colis``
5. Chaque tournée est envoyée à OSRM + VROOM (appel HTTP)
6. Les trajets optimisés sont affichés, et les résultats sont exportables

3. Fichiers / modules principaux

- ``app_flotte_mixte_multiple_custom_input.py`` : interface Streamlit principale, logique utilisateur et affichage
- ``config.py`` : chargement des données, configuration des chemins
- ``zone_desserte.py`` : chargement et filtrage des adresses
- ``sectorisation.py`` : sectorisation équilibrée par k-means contraint
- ``simulation_sectorisation_oronnancement.py`` : tirage aléatoire, regroupement, appel VROOM/OSRM, génération des résultats
- ``flotte.py`` : gestion des véhicules et affectation par tournée
- ``start_sogaflow.sh`` : script bash lançant OSRM et VROOM-Express dans des conteneurs Docker

4. Dépendances Python (triées)

- Interface : streamlit, streamlit_folium, folium, ipyleaflet
- Clustering : k-means-constrained
- Données géo : utm, pyproj, shapely, geopandas
- Machine learning : scikit-learn, joblib
- Traitement Excel : openpyxl, XlsxWriter
- Visualisation : matplotlib, seaborn, plotly
- Divers : pandas, numpy, requests, editdistance, polyline, pyyaml

5. Versions installées (sogaflow_env)

- Python : 3.10.17
- Streamlit : 1.45.0
- pandas : 2.2.3
- numpy : 1.26.4
- scikit-learn : 1.6.1
- matplotlib : 3.10.0
- seaborn : 0.13.2
- folium : 0.19.5
- utm : 0.7.0
- openpyxl : 3.1.5
- XlsxWriter : 3.2.3
- editdistance : 0.6.2
- polyline : 2.0.2

6. Configuration système

- Système principal : Windows 10/11
- Environnement Python : Anaconda (`sogaflow\_env`)
- Lancement Streamlit : via `streamlit run app\_flotte\_mixte\_multiple\_custom\_input.py`
- Services OSRM & VROOM : exécutés via WSL (Ubuntu) et Docker

Annexe I.1 : Document technique Sogaflow – Source : PADE Jules

Sogaflow (Simulation logistique)

Mode de simulation ⓘ

- ☐ Étude rapide (1 simulation)
- ☒ Étude poussée (100 simulations)
- ☐ Comparaison d'entrepôts – rapide (1 simulation)
- ☐ Comparaison d'entrepôts – poussée (100 simulations)
- ☐ Comparaison de flottes – rapide (1 simulation)
- ☐ Comparaison de flottes – poussée (100 simulations)

Zone de desserte à simuler (plusieurs codes séparés par une virgule) ⓘ

92000

Nombre de colis à livrer par jour

2000

Choix de la flotte principale (Flotte A)

Mode de sélection de la flotte ⓘ

- ☒ Flotte type
- ☐ Flotte personnalisée

Choix d'une flotte type

VUL mixte – 50% diesel / 50% électrique (France)

☒ Flotte type sélectionnée : VUL mixte – 50% diesel / 50% électrique (France)

Flotte actuelle (Flotte A) :

	nom	Capacité (nb. colis)	Emissions (gCO ₂ /km)	% Flotte totale
0	VUL_diesel	50	270	
1	VUL_elec	50	00	

Sélection de l'entrepôt

Mode de sélection de l'entrepôt

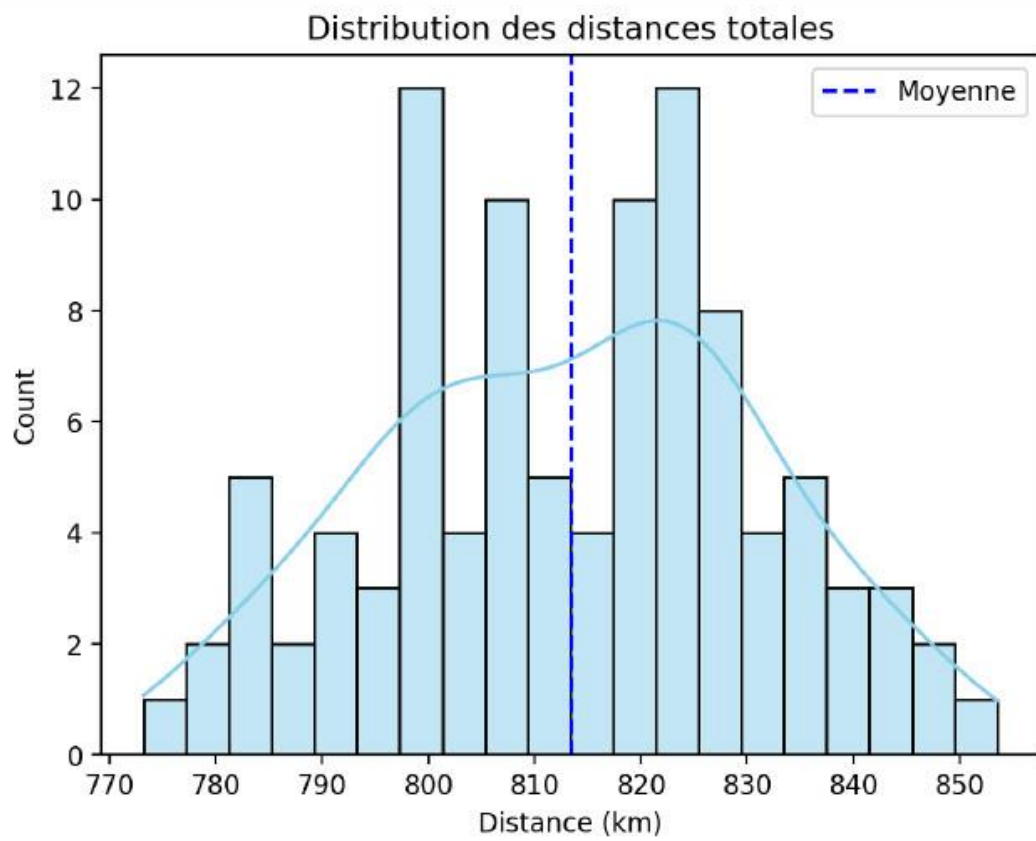
- ☒ Entrepôt Sogaris
- ☐ Base manuelle

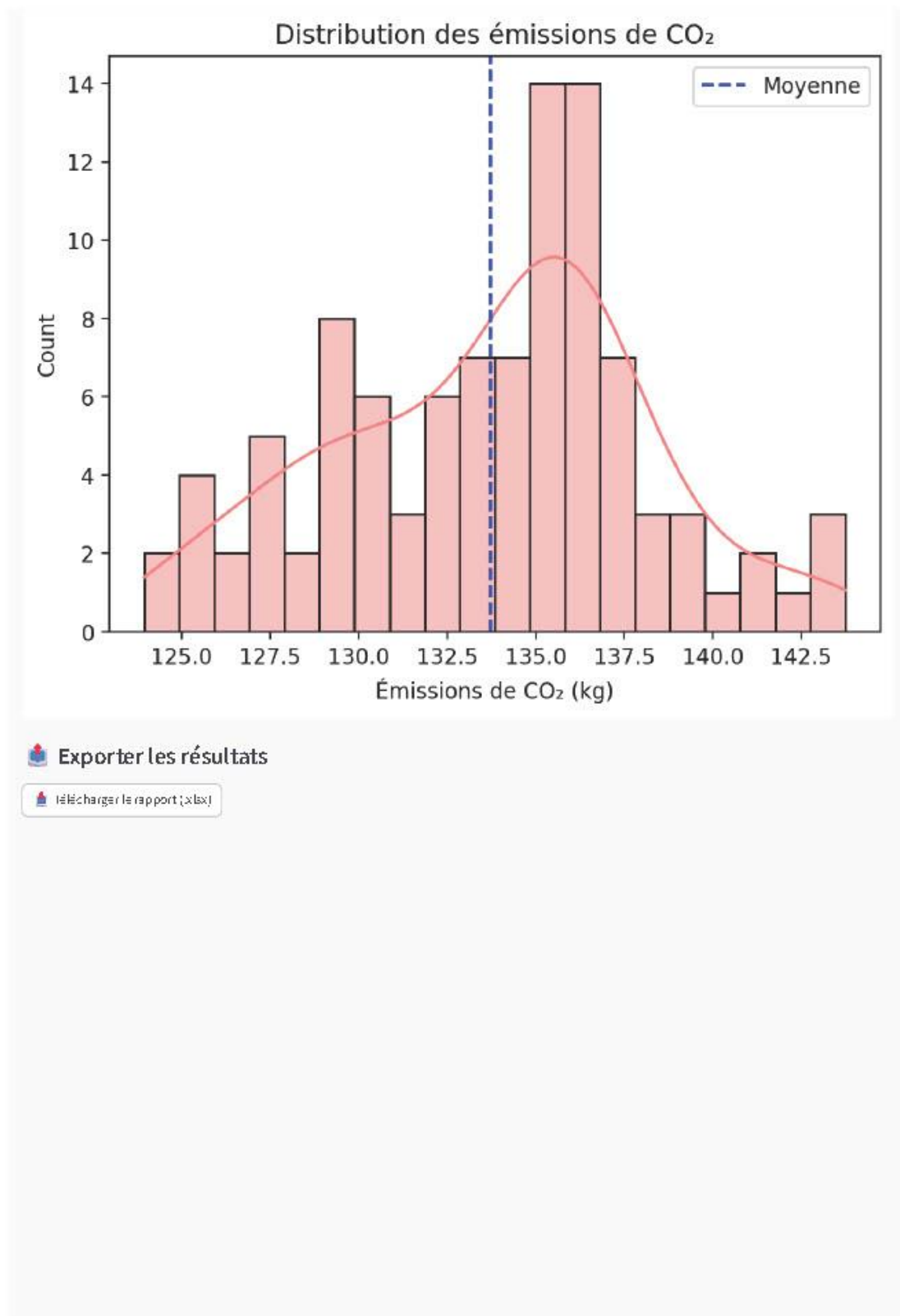
Choix d'un entrepôt Sogaris ⓘ

Beaugrenelle (Espace urbain de distribution)

 Lancer la simulation

Distributions des résultats des 100 simulations





Annexe I.2 : Interface utilisateur avec résultats – Source : PADE Jules

Annexe II :

```
#!/usr/bin/env bash                # Lance le script avec bash.

set -euo pipefail                  # Arrêt sur erreur (-e), variables non définies interdites (-u), pipe
propage les erreurs (-o pipefail).

# =====

# 0) PARAMÈTRES GÉNÉRAUX

# =====

OSRM_IMAGE="osrm/osrm-backend:latest"    # Image Docker officielle d'OSRM.
VROOM_IMAGE="vroomvrp/vroom-docker:latest" # Image Docker officielle de VROOM-Express.
DATA_DIR="$HOME/osrm-data"               # Dossier où se trouvent les fichiers .osm.pbf / .osrm.
OSRM_PBF="ile-de-france-latest.osm.pbf"  # Fichier OSM source (exemple).
OSRM_BASE="ile-de-france-latest.osrm"    # Préfixe attendu par OSRM après
extraction/partition/customize.
OSRM_ALGO="mld"                         # Algorithme d'OSRM (mld recommandé pour grandes zones).
NETWORK_NAME="sogaflow_net"              # Réseau Docker pour interconnecter services.
SUBNET_CIDR="172.23.0.0/16"              # Sous-réseau Docker (optionnel, utile pour IP fixes).
OSRM_NAME="osrm"                         # Nom du conteneur OSRM.
VROOM_NAME="vroom"                       # Nom du conteneur VROOM-Express.
OSRM_PORT=5000                           # Port OSRM exposé.
VROOM_PORT=3000                           # Port VROOM exposé.

# =====

# 1) PRÉREQUIS & RÉSEAU

# =====

command -v docker >/dev/null || { echo "❌ Docker introuvable."; exit 1; } # Vérifie que Docker est
installé.

mkdir -p "$DATA_DIR"                  # Crée le dossier data si besoin.

# Crée un réseau Docker dédié si absent (isole et simplifie la découverte des services).
if ! docker network ls --format '{{.Name}}' | grep -q "^${NETWORK_NAME}$"; then
    docker network create --subnet "$SUBNET_CIDR" "$NETWORK_NAME"    # Réseau avec sous-
réseau fixe (utile mais optionnel).
fi
```

```

# =====

# 2) PRÉPARATION DES DONNÉES OSRM

# =====

# Si tu n'as pas encore généré les fichiers .osrm, décommente les 3 lignes suivantes :

# docker run --rm -v "$DATA_DIR:/data" "$OSRM_IMAGE" osrm-extract -p /opt/car.lua
"/data/$OSRM_PBF" # Extraction

# docker run --rm -v "$DATA_DIR:/data" "$OSRM_IMAGE" osrm-partition "/data/$OSRM_BASE"
# Partition (MLD)

# docker run --rm -v "$DATA_DIR:/data" "$OSRM_IMAGE" osrm-customize "/data/$OSRM_BASE"
# Customize (MLD)


# Vérifie la présence du fichier binaire principal OSRM.
if [ ! -f "$DATA_DIR/$OSRM_BASE" ]; then
    echo "❌ Données OSRM absentes dans $DATA_DIR. Génère-les avant de lancer."; exit 1;
fi

# =====

# 3) (RE)LANCEMENT PROPRES

# =====

# Stop/Remove d'éventuels conteneurs existants pour éviter conflits de ports.
docker rm -f "$OSRM_NAME" >/dev/null 2>&1 || true
docker rm -f "$VROOM_NAME" >/dev/null 2>&1 || true

# =====

# 4) LANCEMENT D'OSRM

# =====

docker run -d --name "$OSRM_NAME" --network "$NETWORK_NAME" -p ${OSRM_PORT}:5000 -v
"$DATA_DIR:/data" "$OSRM_IMAGE" osrm-routed --algorithm="$OSRM_ALGO"
"/data/$OSRM_BASE" # Démarre le serveur OSRM.


# Attente active : ping l'API OSRM jusqu'à réponse 200.
echo "⌚ Attente d'OSRM sur http://localhost:${OSRM_PORT}/health ..."
for i in {1..30}; do
    if curl -s "http://localhost:${OSRM_PORT}/health" | grep -q "OK"; then
        echo "✅ OSRM opérationnel."; break
    fi

```

```

sleep 1

if [ "$i" -eq 30 ]; then echo "❌ OSRM ne répond pas."; docker logs "$OSRM_NAME"; exit 1; fi

done

# =====

# 5) LANCEMENT DE VROOM-EXPRESS

# =====

# VROOM-Express a besoin de connaître l'URL OSRM ; la variable OSRM_HOST est l'endpoint interne
(nom du conteneur).

docker run -d --name "$VROOM_NAME" --network "$NETWORK_NAME" -p ${VROOM_PORT}:3000
-e OSRM_HOST="http://${OSRM_NAME}:${OSRM_PORT}" "$VROOM_IMAGE"
# Démarre VROOM-Express (serveur API).

# Attente active : ping l'API VROOM jusqu'à réponse.

echo "⌚ Attente de VROOM sur http://localhost:${VROOM_PORT} ..."

for i in {1..30}; do

    if curl -s "http://localhost:${VROOM_PORT}" | grep -qi "vroom"; then

        echo "✅ VROOM opérationnel."; break

    fi

    sleep 1

    if [ "$i" -eq 30 ]; then echo "❌ VROOM ne répond pas."; docker logs "$VROOM_NAME"; exit 1; fi

done

# =====

# 6) RÉCAPITULATIF

# =====

echo "-----"

echo "OSRM   : http://localhost:${OSRM_PORT}"

echo "VROOM  : http://localhost:${VROOM_PORT}"

echo "Réseau : ${NETWORK_NAME}"

echo "Données: ${DATA_DIR}"

echo "-----"

echo "🚀 Stack prête (OSRM + VROOM)."
```

Annexe II.1 : Script start_sogaflow.sh avec commentaires - Source : PADE Jules

```

{
  "code": 0,
  "summary": {
    "cost": 4213,
    "routes": 1,
    "unassigned": 0,
    "duration": 3120,
    "distance": 12650
  },
  "routes": [
    {
      "vehicle": 1,
      "cost": 4213,
      "delivery": [45],
      "duration": 3120,
      "distance": 12650,
      "steps": [
        {"type": "start", "location": [2.277, 48.855], "arrival": 0},
        {"type": "job", "id": 102, "location": [2.295, 48.842], "arrival": 780, "service": 300},
        {"type": "job", "id": 103, "location": [2.305, 48.851], "arrival": 1500, "service": 300},
        {"type": "job", "id": 101, "location": [2.289, 48.858], "arrival": 2280, "service": 300},
        {"type": "end", "location": [2.277, 48.855], "arrival": 3120}
      ],
      "geometry": "u`miH_qpM...<polyline_raccourcie>...p`giH}hpM"
    }
  ],
  "unassigned": []
}

```

Annexe II.2 : Exemple de réponse JSON simplifiée renvoyée par l'API VROOM – Source :
PADE Jules

=====

```

# CONFIGURATION SOGAFLOW

# =====

# Chemins vers les données
# -----

# Dossier contenant les shapefiles et bases d'adresses
DATA_DIR = "C:/Users/MonNom/SogaflowEnv/data" # Adapter en fonction de l'installation
SHAPEFILES_DIR = f"{DATA_DIR}/shapefiles"
ADRESSES_CSV = f"{DATA_DIR}/adresses/adresses_geolocalisees.csv"

# Paramètres par défaut pour le chargement
# -----

# Liste des départements à charger (exemple : 92 = Hauts-de-Seine)
DPTS_TO_LOAD = [92]

# Types de fichiers attendus
# -----

# Shape des communes, shape du réseau routier, shape des zones logistiques
COMMUNES_SHP = f"{SHAPEFILES_DIR}/communes.shp"
ROUTES_SHP = f"{SHAPEFILES_DIR}/routes.shp"
ZONES_SHP = f"{SHAPEFILES_DIR}/zones_logistiques.shp"

# Paramètres de simulation
# -----

# Taux de remplissage moyen des véhicules
TAUX_REMPLISSAGE = 0.85 # 85% par défaut

# Profils de véhicules disponibles (alignés avec OSRM)
VEHICLE_PROFILES = {
    "vul": "car",      # Véhicule utilitaire léger
    "velo": "bike",    # Vélo cargo
    "pl": "truck"      # Poids lourd
}

```

```
# API externes (ports et adresses)
```

```
# -----
```

```
OSRM_HOST = "http://localhost:5000" # Service OSRM
```

```
VROOM_HOST = "http://localhost:3000" # Service VROOM-Express
```

Annexe II.3 – Exemple de configuration config.py pour un déploiement local de Sogaflow