

UNIVERSITÉ DU QUÉBEC À MONTRÉAL

DÉVELOPPEMENT D'UN MODÈLE D'OPTIMISATION POUR LES MICROCORRIDORS DE  
TRANSPORT : LE CAS DU PARC JEAN-DRAPEAU

MÉMOIRE

PRÉSENTÉ

COMME EXIGENCE PARTIELLE

MAÎTRISE ÈS SCIENCES (TECHNOLOGIES DE L'INFORMATION)

PAR

HUGO RENARD-BOUCHARD

NOVEMBRE 2025

UNIVERSITÉ DU QUÉBEC À MONTRÉAL  
Service des bibliothèques

Avertissement

La diffusion de ce mémoire se fait dans le respect des droits de son auteur, qui a signé le formulaire *Autorisation de reproduire et de diffuser un travail de recherche de cycles supérieurs* (SDU-522 – Rév.12-2023). Cette autorisation stipule que «conformément à l'article 11 du Règlement no 8 des études de cycles supérieurs, [l'auteur] concède à l'Université du Québec à Montréal une licence non exclusive d'utilisation et de publication de la totalité ou d'une partie importante de [son] travail de recherche pour des fins pédagogiques et non commerciales. Plus précisément, [l'auteur] autorise l'Université du Québec à Montréal à reproduire, diffuser, prêter, distribuer ou vendre des copies de [son] travail de recherche à des fins non commerciales sur quelque support que ce soit, y compris l'Internet. Cette licence et cette autorisation n'entraînent pas une renonciation de [la] part [de l'auteur] à [ses] droits moraux ni à [ses] droits de propriété intellectuelle. Sauf entente contraire, [l'auteur] conserve la liberté de diffuser et de commercialiser ou non ce travail dont [il] possède un exemplaire.»

## REMERCIEMENTS

Je tiens à exprimer ma profonde gratitude à toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce mémoire.

En premier lieu, je remercie chaleureusement mes directeurs de recherche, Franklin Djeumou Fomeni et Walter Rei, pour leur accompagnement rigoureux, leur disponibilité, leurs conseils éclairés et leur bienveillance tout au long de ce projet. Leur expertise et leur confiance m'ont été d'un soutien précieux à chaque étape de cette recherche.

Je remercie également les membres de l'équipe de la Société du parc Jean-Drapeau ainsi que ceux du Campus de la transition écologique, qui ont toujours été disponibles pour répondre à mes questions et ont collaboré activement à la réalisation du projet.

Enfin, je souhaite adresser mes remerciements les plus sincères à ma famille et à mes proches pour leur patience, leur écoute et leur soutien constant tout au long de cette aventure.

## TABLE DES MATIÈRES

|                                                                                              |     |
|----------------------------------------------------------------------------------------------|-----|
| REMERCIEMENTS .....                                                                          | ii  |
| LISTE DES FIGURES .....                                                                      | v   |
| LISTE DES TABLEAUX .....                                                                     | vii |
| LISTE DES ABRÉVIATIONS, DES SIGLES ET DES ACRONYMES .....                                    | ix  |
| RÉSUMÉ.....                                                                                  | x   |
| ABSTRACT .....                                                                               | xi  |
| INTRODUCTION .....                                                                           | 1   |
| CHAPITRE 1 PROBLÉMATIQUE, QUESTION ET OBJECTIF DE RECHERCHE.....                             | 4   |
| 1.1 La problématique spécifique au terrain .....                                             | 4   |
| 1.2 Problème générique de la recherche .....                                                 | 8   |
| 1.3 La question de recherche .....                                                           | 8   |
| 1.4 L'objectif de recherche .....                                                            | 9   |
| 1.4.1 Contribution de la recherche.....                                                      | 9   |
| CHAPITRE 2 REVUE DE LITTÉRATURE .....                                                        | 10  |
| 2.1 Logistique en milieu urbain .....                                                        | 10  |
| 2.1.1 Motifs du développement de la logistique en milieu urbain .....                        | 11  |
| 2.1.2 Barrières au changement pour la logistique en milieu urbain .....                      | 12  |
| 2.1.3 Externalités négatives de la logistique urbaine.....                                   | 13  |
| 2.1.4 Promotion de la logistique en milieu urbain .....                                      | 14  |
| 2.1.5 Méthodes de transport alternatives .....                                               | 15  |
| 2.1.6 Principe de la consolidation.....                                                      | 15  |
| 2.2 Outils d'aide à la décision et recherche opérationnelle pour la logistique urbaine ..... | 16  |
| 2.3 Le Design Science comme cadre de conception.....                                         | 21  |
| CHAPITRE 3 CADRE MÉTHODOLOGIQUE .....                                                        | 23  |
| 3.1 Présentation de la recherche en science du design .....                                  | 23  |
| 3.2 Résumé de la démarche de recherche .....                                                 | 34  |
| CHAPITRE 4 DESIGN ET DÉVELOPPEMENT.....                                                      | 36  |
| 4.1 Formulation du problème.....                                                             | 36  |
| 4.2 Élaboration du modèle .....                                                              | 43  |
| 4.3 Collecte de donnée et production d'instances.....                                        | 49  |

|                                                                                        |     |
|----------------------------------------------------------------------------------------|-----|
| 4.4 Conception et développement de l’outil d’aide à la décision .....                  | 53  |
| CHAPITRE 5 TEST, DÉMONSTRATION ET ÉVALUATION .....                                     | 69  |
| 5.1 Test et évaluation des performances du modèle d’optimisation .....                 | 69  |
| 5.2 Test et évaluation de l’OAD selon les cas d’application au Parc Jean-Drapeau ..... | 80  |
| 5.3 Discussion et généralisation de la théorie .....                                   | 104 |
| CONCLUSION .....                                                                       | 108 |
| ANNEXE A LISTE DES CAUSES .....                                                        | 110 |
| ANNEXE B DIAGRAMME D’ISHIKAWA.....                                                     | 112 |
| ANNEXE C QQQQCCP .....                                                                 | 114 |
| ANNEXE D DESCRIPTION DE L’OUTIL DE COLLECTE DE DONNÉES .....                           | 118 |
| ANNEXE E HISTOIRES UTILISATEUR.....                                                    | 123 |
| BIBLIOGRAPHIE.....                                                                     | 126 |

## LISTE DES FIGURES

|                                                                                                                                                                                                    |    |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Figure 1-1 Carte des deux îles composant le Parc Jean-Drapeau .....                                                                                                                                | 5  |
| Figure 1-2 Représentation graphique du site du parc Jean-Drapeau .....                                                                                                                             | 6  |
| Figure 3-1 RÉSUMÉ DES ÉTAPES DU CADRE MÉTHODOLOGIQUE SUIVIE.....                                                                                                                                   | 35 |
| Figure 4-1 Position des points de livraisons potentiels (Hubs) sur la carte du Parc Jean-Drapeau.....                                                                                              | 37 |
| Figure 4-2 Représentation schématique des points de livraison du parc Jean-Drapeau .....                                                                                                           | 38 |
| Figure 4-3 Graphe cyclique bidirectionnel.....                                                                                                                                                     | 41 |
| Figure 4-4 Trajet de cardinalité 2 .....                                                                                                                                                           | 42 |
| Figure 4-5 Trajet de cardinalité supérieure à 2.....                                                                                                                                               | 43 |
| Figure 4-6 Structure de l'OAD .....                                                                                                                                                                | 54 |
| Figure 4-7 Exemple de fichier appelé item4_demo.csv regroupant le volet item d'une instance de démonstration .....                                                                                 | 55 |
| Figure 4-8 Exemple de fichier appelé bin3_demo.csv regroupant le volet bin d'une l'instance de démonstration.....                                                                                  | 56 |
| Figure 4-9 Arborescence des fichiers et dossiers de l'OAD .....                                                                                                                                    | 57 |
| Figure 4-10 Commande d'aide et arguments du script scriptOAD.py .....                                                                                                                              | 59 |
| Figure 4-11 Exemple de valeurs pour le fichier param.cfg .....                                                                                                                                     | 60 |
| Figure 4-12 Fonction utilisé par défaut pour calculer le coût d'assignation d'un item à une bin .....                                                                                              | 61 |
| Figure 4-13 Aide-mémoire pour la création de fonctions de coût personnalisé dans le fichier cost_functions.py. ....                                                                                | 62 |
| Figure 4-14 Exemple de fonction personnalisée pouvant être utilisé pour calculer le coût d'assignation d'un item à une bin.....                                                                    | 64 |
| Figure 4-15 Commande employée pour exécuter le programme avec deux fichiers de test, en appliquant les fonctions de coût par défaut, et aperçu des résultats.....                                  | 66 |
| Figure 4-16 Commande utilisée pour exécuter le programme avec deux fichiers de test, en appliquant les fonctions de coût personnalisées, ainsi que les résultats obtenus lors de l'exécution ..... | 68 |
| Figure 5-1 Arborescence des fichiers et dossiers pour la version aléatoire de l'OAD .....                                                                                                          | 72 |
| Figure 5-2 Gap moyen (%) par type d'instance et nombre d'item pour le premier ensemble .....                                                                                                       | 76 |

|                                                                                                   |     |
|---------------------------------------------------------------------------------------------------|-----|
| Figure 5-3 Gap (%) par type d'instance et nombre d'item pour le deuxième ensemble .....           | 78  |
| Figure 5-4 Temps CPU (s) par ensemble d'instance .....                                            | 79  |
| Figure 5-5 Gap (%) par ensemble d'instance .....                                                  | 79  |
| Figure 5-6 Position des bureaux (hub) entre lesquels les employés doivent se déplacer .....       | 82  |
| Figure 5-7 Représentation sous forme de graphe du problème de transport de personnel .....        | 83  |
| Figure 5-8 Temps CPU (s) par ensemble d'instance .....                                            | 89  |
| Figure 5-9 Gap (%) par ensemble d'instance .....                                                  | 90  |
| Figure 5-10 Position des points de livraisons pour le problème de transport de marchandises ..... | 94  |
| Figure 5-11 Représentation sous forme de graphe du problème de transport de marchandise .....     | 95  |
| Figure 5-12 Fonctions de coût personnalisés dans le fichier cost_function.py .....                | 98  |
| Figure 5-13 Temps CPU moyen (s) par ensemble d'instance .....                                     | 102 |
| Figure 5-14 Gap moyen (%) par ensemble d'instance .....                                           | 103 |

## LISTE DES TABLEAUX

|                                                                                                                                                       |    |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Tableau 3-1 Classification des artefacts et cadres méthodologiques associés .....                                                                     | 25 |
| Tableau 3-2 Correspondance des étapes méthodologiques entre les cadres méthodologiques d'O'Keefe (2014) et de Norbert, Ouellet et Parent (2009) ..... | 25 |
| Tableau 4-1 Niveau d'activité selon la période .....                                                                                                  | 39 |
| Tableau 4-2 Déplacements possibles selon la cardinalité du trajet .....                                                                               | 42 |
| Tableau 4-3 Ensembles, fonctions, paramètres du modèle .....                                                                                          | 44 |
| Tableau 4-4 Participante de la première entrevue non-dirigée .....                                                                                    | 50 |
| Tableau 4-5 Participants à la deuxième entrevue non-dirigée.....                                                                                      | 52 |
| Tableau 4-6 Participants à la troisième entrevue non dirigé.....                                                                                      | 53 |
| Tableau 4-7 Description des champs des Items dans une instance de transport .....                                                                     | 54 |
| Tableau 4-8 Description des champs des Bins dans une instance de transport.....                                                                       | 55 |
| Tableau 4-9 Arguments attendues par le programme scriptOAD.py .....                                                                                   | 58 |
| Tableau 4-10 Description des paramètres du programme modifiables contenues dans le fichier param.cfg .....                                            | 59 |
| Tableau 4-11 Composantes du modèle utilisables pour les fonctions de coût personnalisées.....                                                         | 63 |
| Tableau 4-12 Principaux types de coût, Valeurs par défaut et valeurs personnalisées .....                                                             | 64 |
| Tableau 5-1 Types d'instances générés aléatoirement .....                                                                                             | 73 |
| Tableau 5-2 Résultats des tests de performances pour les instances du premier ensemble .....                                                          | 74 |
| Tableau 5-3 Résultats des tests de performances pour les instances du deuxième ensemble .....                                                         | 77 |
| Tableau 5-4 Modélisation du problème de transport de personnel dans un microcorridor .....                                                            | 83 |
| Tableau 5-5 Principaux types de coût, Valeurs par défaut et valeurs personnalisées .....                                                              | 84 |
| Tableau 5-6 Principaux types de coût, Valeurs par défaut et valeurs personnalisées .....                                                              | 85 |
| Tableau 5-7 Résultats des tests de performances pour les instances du premier ensemble .....                                                          | 86 |
| Tableau 5-8 Résultats des tests de performances pour les instances du deuxième ensemble .....                                                         | 87 |
| Tableau 5-9 Résultats des tests de performances pour les instances du troisième ensemble.....                                                         | 88 |



|                                                                                                                               |     |
|-------------------------------------------------------------------------------------------------------------------------------|-----|
| Tableau 5-10 Volume, écart-type et taux d’occupation par item moyen des bins par ensemble .....                               | 91  |
| Tableau 5-11 Principaux types de coût, Valeurs par défaut et valeurs personnalisées .....                                     | 95  |
| Tableau 5-12 Modélisation du problème de transport de marchandises pour l’exécution de tâches de maintenance récurrentes..... | 96  |
| Tableau 5-13 Principaux types de coût, Valeurs par défaut et valeurs personnalisées .....                                     | 97  |
| Tableau 5-14 Résultats des tests de performances pour les instances du premier ensemble .....                                 | 99  |
| Tableau 5-15 Résultats des tests de performances pour les instances du deuxième ensemble .....                                | 100 |
| Tableau 5-16 Résultats des tests de performances pour les instances du troisième ensemble .....                               | 101 |

## LISTE DES ABRÉVIATIONS, DES SIGLES ET DES ACRONYMES

|                       |                                                  |
|-----------------------|--------------------------------------------------|
| <b>BPP</b>            | Bin-Packing Problem                              |
| <b>CO<sub>2</sub></b> | Dioxyde de carbone                               |
| <b>OAD</b>            | Outil d'aide à la décision                       |
| <b>RO</b>             | Recherche opérationnelle                         |
| <b>RSD</b>            | Recherche en science du design                   |
| <b>SPJD</b>           | Société du Parc Jean-Drapeau                     |
| <b>STI</b>            | Système de transport intelligent                 |
| <b>TI</b>             | Technologie de l'information                     |
| <b>TIC</b>            | Technologie de l'information de la communication |
| <b>VRP</b>            | Véhicule Routing Problem                         |

## RÉSUMÉ

Le Parc Jean-Drapeau est un espace naturel et culturel emblématique de Montréal, conçu pour offrir aux citoyens un lieu de détente, de loisirs et de découvertes. Toutefois, les activités régulières et événementielles qu'il accueille génèrent une logistique interne complexe, marquée par de nombreux déplacements de marchandises. Ce mémoire s'inscrit dans une démarche d'optimisation de cette logistique urbaine, en introduisant un concept novateur : celui de microcorridor de transport.

Inspiré des principes de la logistique urbaine durable, le microcorridor désigne une portion restreinte et bien définie d'un territoire urbain, à l'intérieur de laquelle on cherche à optimiser les flux de transport à petite échelle, en tenant compte de contraintes spécifiques au contexte local. Dans ce cadre, un outil d'aide à la décision intégrant un modèle d'optimisation de type *bin-packing* a été conçu. Ce modèle permet de regrouper les livraisons dans des véhicules de manière à minimiser les trajets, tout en respectant les contraintes de capacité, de temps et de localisation.

Le réseau logistique du parc a été modélisé sous forme de graphe, ce qui a facilité la représentation des lieux de chargement et de livraison, ainsi que des chemins qui les relient. Afin d'évaluer le modèle de manière réaliste, des histoires utilisateur ont été élaborées à partir d'entretiens et d'observations terrain. Celles-ci ont également permis de mieux comprendre et de cerner avec précision les problèmes rencontrés par le Parc sur le terrain.

Celles-ci ont permis de construire des instances de test représentatives, ancrées dans les besoins concrets des parties prenantes. L'outil développé a ainsi été expérimenté à travers des scénarios plausibles, illustrant son potentiel décisionnel et sa pertinence pour la planification logistique en contexte urbain restreint. Ce travail constitue une première exploration prometteuse de l'application du *bin-packing* à la gestion logistique dans des microcorridors de transport.

**Mots clés :** logistique urbaine, microcorridor, optimisation, bin-packing, aide à la décision, transport durable, Parc Jean-Drapeau.

## ABSTRACT

Parc Jean-Drapeau is a natural and cultural landmark in Montréal, designed to provide citizens with a space for relaxation, leisure, and discovery. However, the regular and event-driven activities it hosts generate a complex internal logistics network, characterized by frequent movements of goods. This thesis is part of an effort to optimize this urban logistics system by introducing an innovative concept: the transport microcorridor.

Inspired by the principles of sustainable urban logistics, a microcorridor refers to a small, well-defined area within an urban territory where transport flows are optimized at a local scale, taking into account specific contextual constraints. Within this framework, a decision-support tool incorporating a *bin-packing* optimization model was developed. This model allows for grouping deliveries into vehicles in a way that minimizes trips, while respecting constraints related to capacity, time, and location.

The park's logistics network was modeled as a graph, which facilitated the representation of pick-up and delivery locations as well as the paths connecting them. To realistically evaluate the model, user stories were developed based on interviews and field observations. These stories made it possible to construct representative test instances, grounded in the real needs of stakeholders. The tool was tested through plausible scenarios, demonstrating its decision-making potential and its relevance for logistics planning in a constrained urban environment. This work represents a promising first exploration of applying bin-packing to logistics management in transport microcorridors.

**Keywords :** urban logistics, microcorridor, optimization, bin-packing, decision-support tool, sustainable transport, Parc Jean-Drapeau

## INTRODUCTION

Le Parc Jean-Drapeau est un parc récréatif situé à Montréal. Il s'étend sur deux îles du fleuve Saint-Laurent : l'île Notre-Dame et l'île Sainte-Hélène. Il a été créé dans les années 1960 dans le cadre des préparatifs de l'Exposition internationale et universelle de 1967, également connue sous le nom d'Expo 67. Le parc Jean-Drapeau constitue une destination prisée tant pour les touristes que pour les habitants de la région. Il abrite plusieurs attractions, dont la Biosphère de Montréal, un musée de l'environnement, le Casino de Montréal, un établissement d'envergure nationale, ainsi que le Circuit Gilles-Villeneuve, une piste de course automobile. Le parc offre également de nombreuses installations récréatives extérieures telles que des plages, des pistes cyclables et des infrastructures sportives (Wikipédia, 2022).

Les partenaires insulaires, tels que La Ronde et le Casino de Montréal, ainsi que les nombreux événements saisonniers organisés sur le site du parc Jean-Drapeau, engendrent d'importants flux de transport, de marchandises comme de personnes, qui sollicitent fortement les infrastructures routières et les stratégies de transport actuellement en place. Dans le contexte de la mise en œuvre d'un nouveau plan directeur axé sur la résilience écologique, les responsables de la gestion du site souhaitent explorer de nouvelles stratégies de transport à la fois écologiques et économiques, s'inspirant des principes de la logistique urbaine.

Un corridor de transport est défini comme un « faisceau d'infrastructures de transport, généralement de natures diverses, servant à relier des lieux stratégiques sur un axe donné, et qui est destiné à faciliter le déplacement de marchandises et de personnes » (Office québécois de la langue française, 2023). Traditionnellement, un corridor de transport est conçu pour couvrir de longues distances, comme dans le cas de traversées intercontinentales. Nous proposons ici d'adapter ce concept au contexte de la logistique urbaine afin d'organiser le transport de marchandises et de personnes sur un axe local. La particularité des corridors de transport en milieu urbain réside dans la courte distance entre les points stratégiques. Nous avons donc choisi d'appeler ce type d'aménagement un « microcorridor de transport ».

Un microcorridor de transport peut ainsi être défini comme un axe de transport aux points d'entrée et de sortie limités, caractérisé par une forte densité de nœuds. Selon cette définition, le chemin reliant les différents sites du parc Jean-Drapeau peut être considéré comme un microcorridor de transport, ce qui ouvre de nouvelles possibilités de modélisation du problème. Le concept de microcorridor de transport est absent de la littérature scientifique, et il n'existe à notre connaissance aucun outil permettant de soutenir une équipe de gestion dans le processus d'acheminement de marchandises à l'intérieur de tels corridors.

La question de recherche qui guide ce mémoire est la suivante : Quel modèle d'optimisation, en tant que composante centrale d'un outil d'aide à la décision, peut soutenir les équipes de la SPJD dans l'optimisation du processus d'acheminement de marchandises dans des microcorridors de transport en contexte urbain ? Comment le concevoir, le développer et le mettre à l'épreuve ?

Pour y répondre, nous proposons de concevoir un modèle d'optimisation de type Bin Packing Problem (BPP), permettant de résoudre des problématiques d'acheminement de marchandises dans des microcorridors urbains. Ce modèle s'inspire des travaux de Crainic *et al.* (2021) et est adapté à la réalité logistique du parc Jean-Drapeau. Il constituera le cœur d'un outil d'aide à la décision (OAD) destiné à permettre aux équipes de gestion de simuler et d'analyser différents scénarios logistiques entre les divers sites du parc.

Selon Leavitt et Whisler (1953), l'application de méthodes d'optimisation à des problèmes de décision s'inscrit dans le champ des technologies de l'information (TI), tel qu'ils le conceptualisaient à l'époque. Les outils d'aide à la décision relèvent ainsi du domaine de l'aide à la décision en TI.

Un outil d'aide à la décision est un artéfact (March et Smith, 1995). En conséquence, sa conception, son développement et sa mise à l'épreuve peuvent s'appuyer sur la recherche en science du design (RSD), à la fois comme approche de recherche et comme cadre méthodologique (Hevner *et al.*, 2004 ; March et Smith, 1995 ; Peffers *et al.*, 2008).

Ce document est structuré comme suit. Le chapitre I traite de la problématique, de la question et de l'objectif de recherche. Le chapitre II présente une revue de littérature portant sur la recherche opérationnelle (RO) en logistique urbaine. Le chapitre III aborde la démarche méthodologique suivie. Le

chapitre IV traite de la conception et du développement des artéfacts. Enfin, le chapitre V présente la phase de test, de démonstration et d'évaluation des artéfacts proposés.

## **CHAPITRE 1**

### **PROBLÉMATIQUE, QUESTION ET OBJECTIF DE RECHERCHE**

Dans ce chapitre, nous présentons la problématique générale et spécifique de la recherche ainsi que la question et l'objectif poursuivis. Nous définissons d'abord les particularités du site du parc Jean-Drapeau, qui constitue le terrain d'expérimentation du projet. Ensuite, nous formalisons le problème générique de la recherche, en introduisant le concept de microcorridor de transport. Finalement, nous exposons la question de recherche, l'objectif poursuivi et la contribution attendue.

#### **1.1 La problématique spécifique au terrain**

Le Parc Jean-Drapeau est un parc montréalais situé au milieu du fleuve Saint-Laurent. Ancien site militaire, il devient accessible au grand public avec l'inauguration du pont Jacques-Cartier en 1930, avant d'être officiellement transformé en parc dans les années 1960. Le site connaît des transformations majeures à l'occasion de l'Exposition universelle de 1967 (Expo 67), notamment par l'agrandissement de l'île Notre-Dame, puis participe à l'accueil des Jeux olympiques d'été de 1976. Aujourd'hui, il regroupe des infrastructures majeures telles que le Casino de Montréal, La Ronde, ainsi qu'un ensemble d'événements culturels et sportifs d'envergure. Ces activités génèrent une logistique complexe, sollicitant fortement les infrastructures de transport existantes.

Le Parc Jean-Drapeau est composé de deux îles principales, l'île Notre-Dame et l'île Sainte-Hélène. La figure 1-1 présente la géographie du site, où l'on distingue quatre pôles majeurs : le pont de la Concorde, le pont Jacques-Cartier, le Casino de Montréal et le site principal du parc Jean-Drapeau. La figure 1-2 illustre les connexions entre ces pôles. Il convient de noter que le pont Jacques-Cartier et le pont de la Concorde constituent les deux seules portes d'accès terrestres au site, en excluant la voie maritime.





Figure 1-1 Carte des deux îles composant le Parc Jean-Drapeau

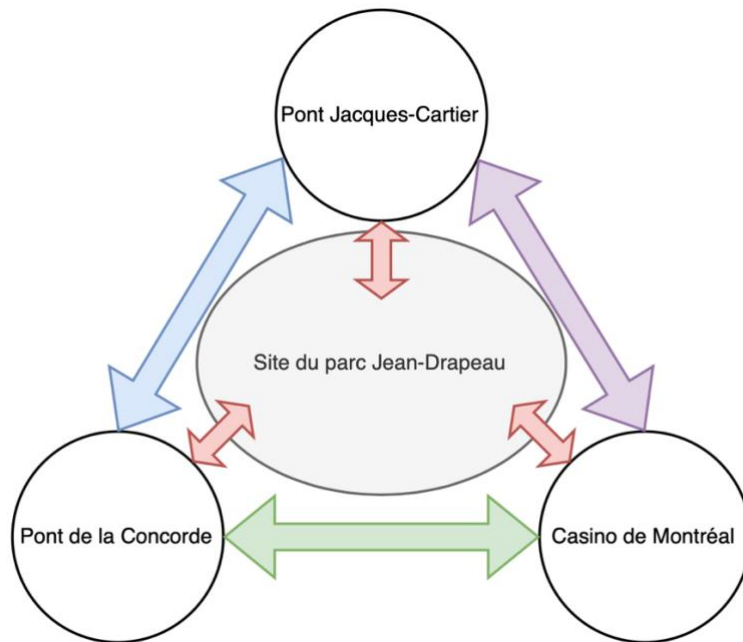


Figure 1-2 Représentation graphique du site du parc Jean-Drapeau

Actuellement, plusieurs acteurs majeurs tels que La Ronde et le Casino de Montréal assurent de manière indépendante la gestion de leur logistique respective. Cette autonomie limite les opportunités de consolidation entre les différentes opérations présentes sur le site. Les mouvements de marchandises qui en résultent peuvent être regroupés en trois catégories : (1) de l'extérieur vers le parc, lorsque des marchandises sont livrées d'un point extérieur vers un point situé à l'intérieur du parc ; (2) du parc vers l'extérieur, lorsque des marchandises sont expédiées depuis un point situé dans le parc vers une destination extérieure ; et (3) à l'intérieur du parc, lorsque des marchandises sont déplacées d'un site à un autre à l'intérieur même du parc. Compte tenu du nombre restreint de points d'accès et du réseau routier limité, les stratégies logistiques en place se révèlent souvent inefficaces. Cette problématique est exacerbée par la mise en œuvre d'un nouveau plan directeur par les autorités du parc, plan qui vise à rendre le site plus écologique et plus attrayant pour les visiteurs. Ce plan introduit notamment l'intégration des principes de logistique urbaine durable dans les stratégies de transport.

Cette problématique s'inscrit dans une dynamique plus large observée dans les grandes villes contemporaines, où l'aménagement des espaces publics vise à réduire la dépendance aux véhicules motorisés lourds, à favoriser le transport actif et à intégrer des stratégies de consolider les flux urbains.

Afin de mieux cerner les enjeux logistiques spécifiques au terrain, une analyse approfondie des causes a été réalisée à partir des données issues du plan directeur (voir Annexe A). Cette analyse repose sur la méthode de questionnement QQQQCCP (*Qui, Quoi, Où, Quand, Comment, Combien, Pourquoi*) (Annexe C), ainsi que sur le diagramme d'Ishikawa (Annexe B). Elle nous a permis d'effectuer les deux constats majeurs suivants:

Premièrement, la qualité et la structure du réseau routier du parc posent problème. Bien que ce réseau relie les principaux pôles, il présente plusieurs lacunes : redondance de certains segments, complexité inutile des trajets et signalisation déficiente. De plus, la modernisation prévue par le plan directeur n'inclut pas de nouvelles routes ; au contraire, elle vise une réduction des surfaces minéralisées en faveur du transport actif (piétons et cyclistes). À terme, l'accès au pont Jacques-Cartier à partir du pont de la Concorde ne sera plus possible sans transiter par le Casino de Montréal, réduisant ainsi à trois les principaux nœuds logistiques du réseau.

Deuxièmement, la gestion des périodes de fort achalandage est problématique. Bien que ces périodes (fins de semaine, saison estivale) soient prévisibles, elles génèrent des défis logistiques nécessitant des mesures spécifiques.

Le Parc Jean-Drapeau accueille chaque année plusieurs dizaines d'événements d'envergure, générant en haute saison des flux logistiques correspondant à plusieurs centaines de livraisons hebdomadaires. Cette intensité d'activité, concentrée sur un réseau routier de quelques kilomètres seulement, accentue les risques de congestion, de saturation et de perturbation des opérations.

Au-delà des enjeux opérationnels, une inefficacité logistique pourrait également entraîner des externalités négatives : augmentation des émissions de gaz à effet de serre, congestion accrue aux abords des accès terrestres, détérioration de l'expérience visiteur, et exacerbation des tensions entre les différents usagers du site. La recherche d'une logistique optimisée s'inscrit ainsi non seulement dans une logique de performance, mais aussi dans une perspective environnementale et sociale plus large.

En résumé, le Parc Jean-Drapeau présente un défi logistique lié à la coexistence d'acteurs autonomes, à un réseau d'accès à la fois dense et limité, à une planification fragmentée ainsi qu'à des objectifs écologiques contraignants. Les dimensions physiques et temporelles de cette configuration correspondent

aux caractéristiques d'un microcorridor de transport, concept que nous mobiliserons et formaliserons dans ce mémoire afin de modéliser et d'optimiser la logistique du site.

## 1.2 Problème générique de la recherche

Le concept de microcorridor de transport est actuellement absent de la littérature scientifique. Alors que celui de corridor de transport est bien documenté dans les contextes interurbains ou internationaux, peu d'études, à notre connaissance, se sont intéressées à son adaptation à l'échelle intra-urbaine. Pourtant, plusieurs contextes logistiques, notamment dans les parcs événementiels ou les quartiers à accès restreint, présentent des caractéristiques similaires aux corridors de transport traditionnels, mais à plus petite échelle. Ces situations se traduisent principalement par des corridors de courte distance comportant une forte densité de points de livraison et une temporalité plus contraignante. Ce sont ces conditions que nous désignons par le terme microcorridor de transport.

À notre connaissance, il n'existe actuellement aucun outil d'aide à la décision spécifiquement conçu pour soutenir une équipe de gestion dans la planification logistique de tels microcorridors. Les approches existantes en recherche opérationnelle (RO) portent principalement sur des réseaux de grande échelle ou sur des problématiques de type livraison du dernier kilomètre (*last-mile delivery*), sans proposer de modèle dédié à la structuration logistique dans des réseaux courts, denses et contraints.

Dans ce contexte, l'objectif est de développer une solution capable d'optimiser les déplacements de marchandises à l'intérieur de microcorridors de transport, à l'aide d'un modèle d'optimisation formel, afin de faciliter la prise de décision des gestionnaires et d'améliorer l'efficacité logistique en milieu urbain restreint.

## 1.3 La question de recherche

Quel modèle d'optimisation, comme composante centrale d'un OAD, peut soutenir les équipes de la Société du Parc Jean-Drapeau (SPJD) pour optimiser le processus d'acheminement de marchandises dans des microcorridors de transport situés en contexte urbain ? Comment le concevoir, le développer et le mettre à l'épreuve ?

#### 1.4 L'objectif de recherche

L'objectif de cette recherche est de développer un outil d'aide à la décision et l'une de ses composantes, un modèle d'optimisation, afin d'optimiser le transport de marchandises dans un contexte de logistique urbaine. Cet outil permettra aux équipes de la SPJD d'analyser différents scénarios pour l'organisation du transport de marchandises à travers le site.

##### 1.4.1 Contribution de la recherche

La littérature scientifique comporte déjà plusieurs références à des modèles de type Bin Packing appliqués à des contextes logistiques et de transport. Le modèle d'optimisation au cœur de l'outil d'aide à la décision s'inscrit dans la continuité de ces travaux. En effet, le modèle développé dans ce mémoire constitue une généralisation de celui présenté par Crainic *et al.* (2021). Plus précisément, l'apport principal réside dans l'extension du modèle de Bin Packing proposé par ces auteurs, qui permettait de planifier la capacité de transport de manière temporelle sur un corridor unique, vers un modèle capable d'effectuer cette planification sur un ensemble de corridors distincts. Il s'agit d'étendre le modèle de Bin Packing de Crainic *et al.* au cas des microcorridors de transport.

Le site du Parc Jean-Drapeau représente un excellent cas d'application pour tester l'applicabilité de cette méthode dans un problème réel à la complexité raisonnable, grâce à son nombre limité de nœuds et à la possibilité de réaliser des observations sur le terrain.

En complément, la méthodologie mobilisée pour répondre à cette problématique comporte également une innovation importante : l'utilisation d'histoires utilisateur pour construire les scénarios d'évaluation de l'outil d'aide à la décision. Cette approche, relativement inédite dans un contexte RSD appliqué à la logistique urbaine, a permis d'explorer la pertinence des histoires utilisateur comme levier méthodologique pour tester, valider et raffiner un artéfact dans un environnement opérationnel simulé.

Ainsi, au-delà du cas spécifique du Parc Jean-Drapeau, la contribution de ce mémoire est double : elle propose d'une part un modèle d'optimisation adapté aux microcorridors de transport, et d'autre part une méthodologie novatrice pour évaluer des outils d'aide à la décision dans le cadre de recherches orientées design. Ces contributions pourront être transposées à d'autres contextes logistiques urbains complexes, notamment les parcs événementiels, les campus fermés ou les quartiers à circulation restreinte.

## CHAPITRE 2

### REVUE DE LITTÉRATURE

Dans ce chapitre, nous présentons le cadre conceptuel et théorique qui soutient la conception du modèle et de l'outil d'aide à la décision développé dans le cadre de ce mémoire. La revue de littérature est structurée autour de trois axes principaux : d'abord, la logistique en milieu urbain et les défis spécifiques à ce contexte ; ensuite, les outils d'aide à la décision basés sur la RO, et plus particulièrement les modèles de type bin packing ; enfin, la RSD, qui encadre la démarche de conception du système. L'objectif est de situer la recherche dans les courants scientifiques existants et de justifier les choix méthodologiques et techniques qui sous-tendent le développement de l'artéfact.

#### 2.1 Logistique en milieu urbain

Savelsbergh et Van Woensel (2016) définissent la logistique urbaine comme étant la recherche de moyens efficaces pour transporter des marchandises dans les zones urbaines tout en tenant compte des effets négatifs sur la congestion, la sécurité et l'environnement. Taniguchi *et al.* (2001) définissent la logistique urbaine comme « le processus d'optimisation totale des activités de logistique et de transport par des entreprises privées avec le soutien de systèmes d'information avancés dans les zones urbaines, en tenant compte de l'environnement du trafic, de la congestion du trafic, de la sécurité du trafic et des économies d'énergie dans le cadre d'une économie de marché. »

Ces deux définitions mettent en évidence l'impact environnemental, la congestion, la sécurité et l'efficacité des services liés au transport de marchandises dans un contexte de logistique urbaine. La deuxième définition vient souligner l'optimisation totale des activités logistiques des entreprises privées plutôt que l'optimisation locale (Taniguchi, 2014), à l'instar de la première qui se veut plus généraliste quant à la portée des opérations.

Dans le contexte de la logistique urbaine, il y a quatre parties prenantes : les expéditeurs, les transporteurs, les administrateurs et les résidents. Ils ont tous des points de vue et des objectifs différents, mais doivent collaborer pour obtenir des villes plus durables et vivables (Taniguchi, 2014).

### 2.1.1 Motifs du développement de la logistique en milieu urbain

Plusieurs tendances viennent encourager le développement des activités de transport urbain. Selon une enquête des Nations unies, près de la moitié de la population mondiale, soit environ 3,5 milliards de personnes, vivait dans des zones urbaines en 2010, et ce chiffre devrait dépasser 60 % d'ici 2030 (Taniguchi, 2014). Sous ces conditions démographiques, la gestion du transport de marchandises en milieu urbain devient un enjeu de première importance. Il s'agit de soutenir le développement économique sans pour autant nuire à la qualité de l'environnement urbain, tout en reconnaissant que cette croissance s'accompagne nécessairement d'une augmentation significative du volume de marchandises transportées, ce qui entraîne d'importants défis quant à l'organisation de ces activités dans une perspective de durabilité.

L'augmentation du commerce en ligne est une autre tendance qui complexifie la gestion de la logistique urbaine. En 2021, l'Europe a connu une augmentation des ventes en ligne de 13 %, passant de 633G à 718G d'euros (Laine-Devroede, 2022), contre seulement 143G en 2009 (AFP, 2010). La livraison des produits directement chez le consommateur plutôt que dans des points de vente augmente considérablement le nombre de mouvements de marchandises, ce qui cause de nombreux défis logistiques (Savelsbergh et Van Woensel, 2016).

Le désir de se faire livrer toujours plus rapidement est un autre facteur de transformation pour la logistique urbaine. Afin de compétitionner avec la disponibilité immédiate des produits dans les commerces physiques, les plateformes de commerce en ligne proposent des livraisons en 1 jour (voire même en 1–2 heures dans le cas d'Amazon Prime dans certaines villes des États-Unis). La gestion de ce type de livraison demande des systèmes hautement dynamiques. Une approche utilisée par les groupes ayant à la fois des magasins physiques et une plateforme en ligne est l'utilisation de ces magasins comme mini centres de distribution (Savelsbergh et Van Woensel, 2016).

Le développement de l'économie de partage engendre également des défis logistiques. Acquier *et al.* (2019) définissent l'économie du partage comme « un ensemble d'initiatives qui augmentent la disponibilité et l'efficacité des ressources sous-utilisées dans la société en organisant des échanges entre pairs ou en favorisant l'accès plutôt que la propriété, ou les deux. » Deux variantes de l'économie de partage sont mentionnées. La consommation collaborative désigne une catégorie d'arrangements économiques dans lesquels les participants partagent l'accès à des produits ou à des services. Le commerce collaboratif implique le partage d'infrastructures et de services logistiques avec des

concurrents. Acquier *et al.* (2017) définissent plutôt l'économie de partage selon trois piliers : l'économie d'accès, l'économie de plateforme et l'économie communautaire.

Enfin, les changements climatiques génèrent un intérêt accru pour l'importance de l'impact qu'exercent les transports urbains sur la santé des résidents et le climat mondial. Les objectifs écologiques ou les conditions environnementales marginales peuvent induire des solutions techniques novatrices, et peuvent en outre s'avérer judicieux sur le plan économique en raison d'une plus grande efficacité des ressources (Working group of the Scientific Advisory Board of German Logistics Association (BVL) *et al.*, 2010).

Ces dynamiques de croissance mettent en évidence la nécessité d'adapter les pratiques logistiques. Toutefois, leur adoption se heurte à de nombreuses barrières, que nous examinons dans la section suivante.

#### 2.1.2 Barrières au changement pour la logistique en milieu urbain

Afin de pouvoir implémenter des changements de manière efficace, il est important de connaître et comprendre les facteurs pouvant y faire obstacle, ainsi que la manière dont ces barrières interagissent entre elles.

Kervall et Pålsson (2022) ont identifié onze catégories d'obstacles au changement : la technologie, les infrastructures, l'économie, les règlements, la politique, les objectifs, l'organisation, la connaissance, la coopération, les facteurs sociétaux et le désavantage du premier arrivé. Ces catégories peuvent être classées selon trois groupes correspondant aux activités du processus de gestion du changement.

Afin de faciliter le processus de changement, il faut premièrement s'occuper des barrières stratégiques. Définir une stratégie cohérente permet d'assurer une bonne organisation et collaboration entre les individus impliqués dans le processus de changement. Les barrières stratégiques peuvent être politiques, organisationnelles ou coopératives.

Il faut ensuite s'occuper des barrières instrumentales. L'instrumentalisation permet de formaliser la stratégie en prévision de l'implémentation. Les barrières instrumentales peuvent être au niveau des objectifs, des connaissances ou des règlements.



Retirer les barrières de ces deux étapes devrait supporter le retrait des barrières de l'étape d'implémentation. L'implémentation permet la réalisation des changements définis lors de l'élaboration de la stratégie. Les barrières à l'implémentation sont au niveau des infrastructures, de l'économie et de la technologie.

Les facteurs sociétaux et le désavantage du premier arrivé sont liés à l'environnement dans lequel évolue le système. Ils affectent l'ensemble des trois étapes du processus précédemment catégorisé et doivent être adressés en priorité.

### 2.1.3 Externalités négatives de la logistique urbaine

Lorsqu'on examine les externalités liées au transport de marchandises, celles généralement considérées incluent les émissions de dioxyde de carbone (CO<sub>2</sub>), le bruit, la congestion, les accidents, la pollution de l'eau et l'utilisation des sols. Cependant, il existe d'autres externalités telles que les effets dus à la production de véhicules et d'infrastructures de transport. Il s'agit notamment de la production d'énergie, de biens et de services, de l'entretien et de l'élimination des véhicules, ainsi que de la construction des infrastructures (Demir *et al.*, 2015 ; Taniguchi, 2014).

Lorsque l'on examine des cas spécifiques à la logistique urbaine, plus particulièrement dans un contexte d'économie de partage, on peut également observer des externalités d'ordre social. En étudiant les paradoxes de la durabilité dans les initiatives d'économie de partage, Moncef et Monnet Dupuy (2021) permettent l'identification d'externalités spécifiques au contexte de la logistique urbaine.

Le premier paradoxe pertinent est le suivant : mettre en relation les acteurs de l'offre et de la demande et optimiser la logistique. La pression sur les coûts liés aux ressources logistiques nécessite une optimisation et des compromis pour relier l'offre et la demande. Cela conduit à l'utilisation des coursiers comme variable d'ajustement dans le processus de planification et d'optimisation logistique, ce qui peut entraîner des conséquences sociales négatives pour ces derniers. Les conséquences identifiées par les auteurs sont les suivantes : variation des horaires des coursiers, annulations tardives des courses prévues et incertitudes de paiement. Ainsi, l'optimalité du système logistique peut avoir comme externalité une dégradation des conditions de travail des employés.

Une autre paire de propositions paradoxales pertinente est la suivante : la réduction des émissions de gaz et l'adaptation de la capacité de livraison. Un objectif commun à plusieurs acteurs de la logistique urbaine est la réduction de l'impact environnemental. Pourtant, l'incertitude liée au volume de commandes à transporter encourage l'utilisation de camions de livraison surdimensionnés. Ceux-ci ne sont pas toujours remplis à capacité, ce qui nuit aux objectifs de réduction de CO<sub>2</sub> et contribue aux externalités environnementales négatives. Il est d'ailleurs pertinent de noter que cette situation problématique est fortement liée au concept de la consolidation, qui sera discuté un peu plus loin.

Selon le contexte, cette situation peut également engendrer des conséquences sociales. Si le transport de marchandises s'effectue dans le cadre d'une entreprise de livraison à vélo, un volume de livraison imprévisible peut générer de mauvaises conditions de travail (vélos surchargés, risques d'accident) ou un non-paiement des coursiers (en cas d'annulation des livraisons).

#### 2.1.4 Promotion de la logistique en milieu urbain

Les responsables de la logistique sont des acteurs clés des opérations de transport de marchandises en ville qui, en reprenant le processus de changement identifié dans la section 2.1.2, ont un grand impact sur le plan stratégique et l'aspect sociétal. Afin de faciliter le changement, il existe des certifications permettant de sensibiliser les employés aux actions visant à développer des systèmes logistiques plus écologiques, telles que la certification ISO 9001 (gestion de la qualité) et la certification ISO 14001 (gestion de l'environnement) (Taniguchi, 2014).

Au niveau technologique, l'application des technologies innovantes des technologies de l'information et de la communication (TIC) et des systèmes de transport intelligents (STI) dans le transport urbain de marchandises permet de collecter des données précises sur les mouvements des camions de ramassage-livraison sur les réseaux routiers urbains, à moindre coût. Ce type d'optimisation des opérations des véhicules peut contribuer à réduire les coûts logistiques, à diminuer les émissions de CO<sub>2</sub>, d'oxydes d'azote (NO<sub>x</sub>) et de particules en suspension, ainsi qu'à réduire les embouteillages (Taniguchi, 2014).

Les partenariats public-privé permettent à toutes les parties prenantes de participer à l'élaboration des plans de transport de marchandises en ville dès la phase initiale, au lieu de se baser uniquement sur leurs propres données. Le partage d'information entre les entreprises privées et le secteur public est très utile pour comprendre la situation réelle de la distribution des marchandises et les problèmes connexes. Il

permet ainsi la mise en place de mesures dont l'impact est mesuré pour tous dès leur déploiement (Taniguchi, 2014).

#### 2.1.5 Méthodes de transport alternatives

Plusieurs nouveaux modes et méthodes de transport ont vu le jour pour soutenir les activités logistiques en milieu urbain tout en limitant leur impact sur la population et l'environnement. Par exemple, les véhicules à carburant alternatif prennent de plus en plus d'importance dans les systèmes de livraison. Ces derniers fonctionnent généralement à l'électricité, à l'hydrogène ou au gaz naturel, et offrent une alternative qui se veut plus écologique et durable aux carburants à base de pétrole (Savelsbergh et Van Woensel, 2016).

Concernant les véhicules électriques, le développement des infrastructures favorisant leur utilisation soutient leur démocratisation (Hughes-Cromwick et Cregger, 2013). Cependant, leur coût d'acquisition reste généralement plus élevé que celui des véhicules à essence, ce qui peut compliquer l'élaboration d'un modèle économique viable. Une intervention des gouvernements, par le biais d'incitatifs, pourrait contribuer positivement à leur développement (Quak *et al.*, 2016).

L'intégration de véhicules plus petits et plus maniables aux flottes des transporteurs en ville devient aussi de plus en plus répandue, et pourrait permettre de réduire les émissions de CO<sub>2</sub> en diminuant le nombre de véhicules polluants sur les routes. L'utilisation de vélos cargos en est un bon exemple. Cependant, il est important d'examiner les avantages et les limites de chaque type de véhicule, afin d'adapter la flotte aux besoins spécifiques. Le vélo cargo, par exemple, est écologique et parfaitement adapté à la circulation urbaine. Toutefois, il ne permet que le transport de colis de petite taille. Sa capacité de chargement réduite implique qu'un plus grand nombre de véhicules (et de conducteurs) est nécessaire pour livrer le même volume de colis dans un délai raisonnable. Il est ainsi mieux adapté à la livraison locale et dynamique à partir de microdépôts (Llorca et Moeckel, 2021).

#### 2.1.6 Principe de la consolidation

Chaque déplacement effectué par un véhicule génère un coût, qu'il soit économique ou écologique. La consolidation des marchandises est l'une des meilleures pratiques lorsqu'il s'agit de réduire ces coûts et d'augmenter la synergie entre les flottes en utilisant au maximum les ressources disponibles.

Les avantages pour les camionneurs et les expéditeurs vont de l'augmentation des économies d'échelle à la réduction des coûts de collaboration, en passant par une offre de transport plus variée, une évaluation plus facile des offres, une diminution des risques associés au transport, une réduction des émissions de gaz à effet de serre et un renforcement de la confiance entre partenaires (Montecinos *et al.*, 2021). C'est un concept clé qui est à la base de plusieurs initiatives de logistique urbaine.

Une application intéressante du principe de consolidation consiste à intégrer les réseaux de transport public et de marchandises afin d'améliorer les opérations, car les mêmes besoins de transport peuvent être satisfaits avec moins de véhicules et de conducteurs (Savelsbergh et Van Woensel, 2016). Cela permettrait entre autres d'assurer la faisabilité économique des systèmes de transport collectif, tout en réduisant les externalités négatives liées au transport urbain de marchandises. En pratique, il est par exemple possible d'intégrer des systèmes de casiers faisant office de mini-points de dépôt à un réseau de transport en commun pour que ce dernier serve à transporter la marchandise (Oliveira *et al.*, 2022).

Ce principe de consolidation pose ainsi les bases pour des modèles de planification tels que le bin packing, abordés dans la section suivante.

## 2.2 Outils d'aide à la décision et recherche opérationnelle pour la logistique urbaine

Les systèmes de transport de marchandises en milieu urbain font appel à des outils d'optimisation pour planifier les opérations de manière efficiente, particulièrement dans un contexte où les contraintes sont nombreuses (temps, espace, coût, ressources). Parmi ces outils, la RO permet de modéliser les systèmes complexes, tandis que certains modèles spécialisés comme le bin packing permettent de structurer les flux à l'intérieur même des réseaux. Ces approches constituent des composantes fondamentales des OAD dans les environnements logistiques urbains.

La RO constitue un cadre analytique central dans la résolution de problèmes logistiques. Elle vise à modéliser et optimiser les processus complexes en s'appuyant sur des méthodes mathématiques, algorithmiques et de simulation.

Monks (2015) propose de la considérer comme une science de l'implémentation, soulignant l'importance de concevoir des modèles qui ne soient pas uniquement optimaux en théorie, mais également utilisables

en pratique, capables d'influencer concrètement la planification logistique dans les environnements urbains.

Dans leur revue des systèmes de transport intelligents, Crainic *et al.* (2009) soulignent que le développement technologique a souvent devancé les outils d'analyse, notamment dans le domaine de la logistique urbaine. Ils proposent d'intégrer davantage les méthodes de RO dans la conception de systèmes décisionnels intelligents, capables d'analyser et de réagir à des données dynamiques en temps réel, en particulier dans les Advanced Fleet Management Systems.

Marrekchi *et al.* (2021) proposent une taxonomie des problèmes de routage verts (Green Routing Problems), une extension des *Vehicle Routing Problems* (VRP), qui inclut des objectifs environnementaux tels que la réduction des émissions ou du bruit. Ils distinguent les VRP centrés sur les nœuds (points de livraison) et les *Arc Routing Problem* (ARP) centrés sur les arcs (segments routiers), tous deux applicables à la distribution urbaine. Le problème de tournées de véhicules (VRP), introduit par Dantzig et Ramser en 1959 sous le nom de *truck dispatching problem*, vise à déterminer l'itinéraire optimal permettant de visiter l'ensemble des points d'un réseau. Il s'agit de satisfaire la demande d'un ensemble de clients dispersés géographiquement à partir d'un dépôt central, en utilisant une flotte homogène de véhicules. L'objectif principal est de minimiser le coût économique total, généralement mesuré par la distance parcourue ou le temps total passé sur les routes (Marrekchi *et al.*, 2021).

Si ces modèles de routage constituent un pilier de la RO appliquée à la logistique urbaine, ils présentent certaines limites. En effet, ils se concentrent principalement sur la planification des itinéraires et intègrent de manière simplifiée les contraintes de chargement et de capacité. Or, la performance logistique dépend aussi de l'optimisation de l'espace de transport, c'est-à-dire de la manière dont les marchandises sont consolidées et disposées à l'intérieur des véhicules.

Cette problématique, moins centrée sur le parcours que sur l'utilisation optimale des ressources de transport, est précisément abordée par les modèles de *bin packing*, qui prolongent la logique de la RO vers la gestion fine des capacités.

Afin de diminuer les coûts d'opération et de distribution lors du déplacement des biens à travers un réseau, il est possible d'utiliser des outils d'aide à la décision basés sur des modèles de *bin packing*. Le problème de base de *Bin Packing* (BPP) consiste à charger tous les items dans le plus petit nombre de bacs de

transport possible afin d'optimiser la répartition et de réduire les coûts (Crainic *et al.*, 2021). Contrairement au VRP, le BPP ne prend pas de décision sur les itinéraires à emprunter, mais se concentre exclusivement sur le chargement et la consolidation des marchandises.

Dans les modèles de VRP, l'organisation des tournées de véhicules repose sur la prise en compte explicite de plusieurs contraintes définissant la faisabilité des solutions. Parmi celles-ci, les contraintes de capacité jouent un rôle central : elles limitent la quantité ou le volume de marchandises qu'un véhicule peut transporter et, par conséquent, la consolidation possible des livraisons au sein d'une même tournée. Les modèles de BPP apportent une contribution complémentaire à cet égard, en proposant une formulation plus détaillée des contraintes de consolidation à l'intérieur d'un véhicule ou d'un conteneur. Ils permettent notamment de représenter avec précision le volume total des articles à charger, leur poids, ainsi que leur disposition spatiale à l'intérieur des compartiments de transport, offrant ainsi une modélisation plus fine des capacités logistiques.

Dans le cas étudié, bien que la problématique présente plusieurs caractéristiques propres au problème de VRP avec ramassage et livraison (*pickup and delivery*), l'approche par BPP s'avère plus pertinente. En effet, contrairement au VRP classique, il n'est pas nécessaire ici de construire explicitement les itinéraires : ceux-ci peuvent être énumérés préalablement et considérés comme des constantes. Le modèle de BPP permet alors d'assigner chaque bin à l'un de ces trajets prédéfinis, éliminant ainsi la dimension combinatoire liée à la génération des routes. Cette simplification offre la possibilité de se concentrer sur la consolidation optimale des véhicules, tout en maintenant une représentation fidèle des contraintes opérationnelles.

Depuis plusieurs années, les problèmes de type bin packing ont été largement mobilisés dans le domaine de la logistique urbaine, notamment pour modéliser la planification des ressources dans des contextes de livraison du dernier kilomètre. Ces modèles permettent de minimiser les coûts de transport ou de maximiser l'utilisation des ressources tout en respectant diverses contraintes opérationnelles.

Crainic *et al.* (2021) proposent une généralisation multi-période du BPP. Leur modèle prend en compte des contraintes temporelles, les coûts d'affectation item-bin et la disponibilité variable des ressources au fil du temps. Il permet également d'intégrer des transporteurs alternatifs, comme ceux provenant du *spot market*, ce qui enrichit les possibilités de planification. Leur contribution repose sur l'introduction d'un modèle appelé *Multi-period Variable Cost and Size Bin Packing Problem with Assignment Costs*, dans lequel les coûts d'affectation des items aux bins dépendent non seulement des dimensions physiques et des

coûts fixes, mais également des caractéristiques temporelles des items et des bacs (ex. : date de disponibilité, fenêtre de livraison, durée du transport). Chaque bin peut être sélectionné à une période donnée selon sa fenêtre de disponibilité, et les items peuvent être affectés à ces bins en tenant compte de leur fenêtre de livraison ainsi que des coûts induits (pénalités pour livraison anticipée ou retardée). Le modèle permet aussi d'avoir recours à des bins *spot-market*, plus coûteuses, lorsque les ressources régulières sont insuffisantes.

Par ailleurs, Crainic *et al.* (2011) développent un modèle de bin packing avec coûts et tailles variables (*Variable Cost and Size Bin Packing Problem*), dans lequel les bins de transport sont hétérogènes, tant en termes de volume que de coût d'utilisation. Cette approche prend en compte la réalité économique des opérations logistiques urbaines, où la sélection des ressources doit répondre à des compromis complexes entre capacité et coût. Leurs heuristiques, appuyées par des bornes inférieures efficaces, permettent d'obtenir rapidement des solutions de qualité même sur de très grands ensembles d'items, renforçant ainsi l'intérêt de ce modèle pour la planification urbaine multi-scénarios

Dans un registre complémentaire, Baldi *et al.* (2019) proposent une extension appelée *Generalized Bin Packing Problem with bin-dependent item profits*. Cette version enrichit le modèle de base en introduisant une fonction objectif bi-critère : elle prend en compte à la fois les coûts de transport et les profits associés à chaque item-bin, selon le profil de chaque transporteur.

De leur côté, Perboli *et al.* (2021) introduisent une autre variante, le *Variable Cost and Size Bin Packing Problem with Time Dependency*, qui modélise un système de livraison à deux niveaux impliquant des centres satellites. Cette formulation tient compte non seulement de la taille variable des conteneurs, mais aussi de la dépendance temporelle des coûts et des ressources.

Farahani *et al.* (2024) introduisent le *Dynamic Stochastic Temporal Bin Packing Problem*, une extension du *bin packing* conçue pour la planification de capacité dans des corridors logistiques soumis à incertitude. Ce modèle intègre la stochasticité des arrivées de conteneurs et la dimension temporelle continue. Les auteurs comparent une programmation stochastique à deux étapes, reposant sur la génération de scénarios probabilistes, à une approche d'apprentissage par renforcement profond (*Deep Reinforcement Learning*, DRL) utilisant l'algorithme *Proximal Policy Optimization* (PPO), qui assure une mise à jour stable et progressive de la politique d'apprentissage du modèle. Trois heuristiques stochastiques servent également de référence : *First Fit*, *Postpone–First Fit* (décision différée pour limiter l'incertitude) et une

heuristique aléatoire. Leurs résultats montrent que, si la programmation stochastique produit des solutions légèrement meilleures, le DRL s'avère plus efficace et réaliste en pratique grâce à sa capacité d'adaptation rapide, à sa robustesse computationnelle et à sa meilleure exploitabilité industrielle. Ce travail ouvre ainsi une voie prometteuse pour l'application de l'intelligence artificielle dans la planification de capacité logistique en temps réel.

Bruni, Crainic et Perboli (2024) proposent pour leur part une synthèse approfondie des méthodologies de *bin packing* appliquées à la planification de capacité en transport et logistique. Les auteurs mettent en lumière les limites des formulations classiques de capacité, souvent trop simplificatrices, et montrent en quoi le recours au *bin packing* permet de représenter de manière plus fine l'affectation des flux de marchandises aux unités de transport et de stockage. Les auteurs identifient plusieurs extensions pertinentes : prise en compte de contraintes multiples (poids, volume, conditions spécifiques comme la réfrigération), intégration de la dimension temporelle dans les décisions de capacité, et modélisation de contextes multiacteurs, où la coopération et la compétition entre expéditeurs, transporteurs et intermédiaires logistiques influencent directement la planification. Enfin, le chapitre ouvre des perspectives de recherche en insistant sur la nécessité d'intégrer les méthodologies de *bin packing* dans les modèles de conception de réseaux logistiques et d'enrichir les formulations avec des dimensions liées à la durabilité et à l'efficacité énergétique.

Ces travaux témoignent d'un élargissement progressif du cadre *bin packing* vers des modèles capables de mieux représenter la complexité des opérations logistiques en milieu urbain, intégrant simultanément des dimensions économiques, temporelles et organisationnelles. Une revue récente de Boysen *et al.* (2021) propose une synthèse des concepts de livraison du dernier kilomètre selon une perspective de RO. L'accent y est mis sur la diversité croissante des configurations logistiques (véhicules autonomes, drones, vélos cargo, micro-dépôts, etc.) et sur les défis associés à leur opérationnalisation. Cette étude met en évidence le rôle central des outils d'aide à la décision pour structurer et comparer les chaînes logistiques alternatives selon des critères économiques, temporels et environnementaux. Les auteurs soulignent notamment la nécessité de développer des modèles capables de s'adapter à des environnements dynamiques et fragmentés, ce qui rejoint les préoccupations soulevées par l'intégration du *bin packing* dans la logistique urbaine.



Par ailleurs, ces modèles mathématiques ne sont pas conçus dans une logique purement théorique : ils ont vocation à être intégrés dans des OAD. Une fois opérationnalisés dans des interfaces interactives, ces modèles permettent aux décideurs de simuler différents scénarios, d'évaluer les conséquences de leurs choix et de sélectionner les configurations les plus efficaces. Bien que cette finalité ne soit pas toujours explicitement mentionnée dans la littérature sur les BPP, elle en constitue une implication naturelle. Ces modèles forment ainsi le noyau computationnel d'outils exploitables sur le terrain, notamment dans les contextes de logistique urbaine dense ou fragmentée.

En ce sens, ces approches participent également à une démarche de RSD, dans laquelle l'élaboration de l'artéfact repose sur une réponse structurée à un problème réel. Le lien entre modélisation formelle et contexte d'usage est au cœur de cette logique : les solutions proposées ne visent pas uniquement l'optimalité théorique, mais aussi l'utilisabilité, l'adaptabilité et la valeur stratégique dans l'action.

### 2.3 Le Design Science comme cadre de conception

La démarche adoptée dans le cadre de cette recherche s'inscrit dans une logique de conception d'un artéfact visant à répondre à un problème observé dans un contexte réel, soit celui de la planification logistique dans des microcorridors urbains. Ce type d'approche est associé au paradigme de la RSD, une perspective de recherche orientée vers la construction de solutions formelles à des problèmes pratiques (Hevner *et al.*, 2004 ; March et Smith, 1995 ; Peffers *et al.*, 2008).

Selon ce cadre, les artéfacts développés peuvent prendre différentes formes (modèles, systèmes, méthodes) et doivent être à la fois fonctionnels et fondés sur des principes de recherche rigoureux. Dans le cas présent, il s'agit d'un modèle de bin packing multi-période intégré à un outil d'aide à la décision, adapté à des contextes urbains restreints.

Le processus suivi correspond aux six étapes décrites par Peffers *et al.* (2008) : formulation du problème, définition des objectifs, conception de l'artéfact, démonstration, évaluation et diffusion. Cette structuration permet de lier directement le besoin identifié à une solution développée de manière itérative et justifiable.

Sur le plan épistémologique, cette démarche s'inscrit dans une logique pragmatique. La valeur du modèle repose moins sur son caractère novateur que sur sa capacité à produire un changement observable dans

un contexte défini (Goldkuhl, 2012). Ce principe permet de concentrer l'attention sur les effets du modèle, autant que sur sa structure formelle.

L'artéfact développé dans ce mémoire peut être situé, selon la typologie de Hevner et Gregor (2013), au niveau d'un modèle opérationnel. Il ne s'agit ni d'une simple application locale, ni d'une théorie généralisée, mais d'un modèle transposable à d'autres configurations logistiques présentant des contraintes similaires.

L'évaluation du système s'appuie sur les principes du cadre FEDS proposé par Venable *et al.* (2016). Cette approche insiste sur la nécessité d'évaluer à la fois la pertinence de l'artéfact pour les utilisateurs finaux et la rigueur de sa conception. Le modèle est donc évalué en fonction de sa capacité à améliorer la prise de décision logistique dans des environnements fragmentés, tout en respectant les exigences méthodologiques propres à la RO.

Cette perspective rejoint par ailleurs les travaux de O'Keefe (2014) et Monks (2015), qui proposent de considérer la RO comme une science de l'implémentation. Les modèles sont alors conçus non seulement pour être justes, mais aussi pour être applicables, utilisables, et capables de soutenir une transformation des pratiques existantes.

## CHAPITRE 3

### CADRE MÉTHODOLOGIQUE

Dans ce chapitre, nous présentons le cadre méthodologique de la recherche. Celui-ci s'ancre dans la RSD en RO orientée design (O'Keefe, 2014) et opère une synthèse avec le cadre méthodologique de la RO (Norbert, Ouellet et Parent, 2009). Deux artéfacts ont été conçus, développés et validés soit un outil d'aide à la décision (OAD) et un modèle d'optimisation (MO). L'outil d'aide à la décision est une réponse spécifique aux défis logistiques du Parc Jean-Drapeau, mais il pourra être généralisé à d'autres situations logistiques présentant des problèmes similaires liés à des microcorridors de transport. Dans une première section, nous présentons la RSD pour la RO orientée design et le cadre méthodologique de la RO. Par la suite, nous détaillons chacune des étapes qui ont été suivies en présentant les méthodes et outils mobilisés pour concevoir, développer et valider l'artéfact.

#### 3.1 Présentation de la recherche en science du design

La RSD s'ancre dans les sciences de l'artificiel (Simon, 2004) qui cherchent à établir les fondations d'une nouvelle perspective épistémologique pour traiter les problèmes de design. La RSD se donne pour objectif, en partant de la définition d'un problème, d'y répondre en proposant un ou plusieurs artéfacts qui peuvent prendre la forme de construits, de modèles, de modèles et d'instanciation. Les artéfacts instanciés sont à titre d'exemple des produits, processus, services, technologies ou idées. À côté de ces artéfacts, la RSD produit de nouvelles connaissances prescriptives (Hevner *et al.*, 2004 ; Hevner et Gregor, 2013 ; Peffers *et al.*, 2008) qui portent tout autant sur les processus de conception et développement des artéfacts que sur leur utilisation en situation.

Dans cette recherche, nous associons le cadre méthodologique de la RSD pour la RO orientée design (*A Design Science methodology for design oriented OR*) d'O'Keefe (2014) avec la méthode de RO proposée par Norbert, Ouellet et Parent (2009).

O'Keefe (2014) suggère qu'un cadre méthodologique fondé sur la RSD en RO est plus approprié que les méthodes de RO habituelles pour proposer une solution qui soit utilisable en situation. O'Keefe (2014) justifie ce point de vue en avançant l'idée qu'habituellement la RO se concentre davantage sur l'optimisation théorique des systèmes en utilisant des modèles mathématiques abstraits, souvent déconnectés des contextes spécifiques d'application et des besoins opérationnels des utilisateurs finaux.

Cela peut limiter la pertinence des résultats dans des environnements complexes où les contraintes pratiques, les retours des parties prenantes et l'adaptabilité du système sont essentiels.

À l'inverse, pour O'Keefe (2014), le cadre méthodologique de la RSD permet d'intégrer ces dimensions dès les premières étapes du processus, en mettant l'accent sur la création d'artéfacts fonctionnels et pertinents, adaptés à des contextes réels qui sont des solutions appropriées aux problèmes rencontrés par les parties prenantes. Ce cadre méthodologique favorise une évaluation itérative et contextuelle, garantissant non seulement la validité du modèle de RO, mais aussi son utilité pratique et son acceptation par les utilisateurs finaux.

O'Keefe (2014) reprend le modèle du processus de RSD tel que proposé par Peffers et *al.* (2008) qu'il adapte à la RO orientée design. Six étapes principales sont définies : (1) l'identification du problème et des motivations ; (2) la définition des objectifs pour une solution ; (3) le design et développement ; (4) les tests et démonstrations ; (5) l'évaluation ; (6) la généralisation de la théorie. Ces étapes ont comme extrant final un artéfact qui est l'outil d'aide à la décision, ci-après OAD.

De plus, nous avons également intégré au cadre méthodologique d'O'Keefe (2014) le cadre méthodologique de la RO comme proposé par Norbert, Ouellet et Parent (2009). Ce cadre nous a permis de préciser les étapes nécessaires au développement du modèle d'optimisation au cœur de l'OAD. Ce cadre méthodologique comporte sept étapes, soit (1) la détection d'un problème, (2) la formulation d'un problème, (3) l'élaboration d'un modèle, (4) la collecte de données, (5) la résolution du modèle, (6) la validation du modèle, et (7) la mise en œuvre de la solution. Ces étapes ont comme extrant final un artéfact qui est le modèle d'optimisation, ci-après MO.

En résumé, deux artéfacts ont été produits dans la recherche, soit l'OAD qui prend la forme d'un programme en Python (artéfact 1), et au cœur de ce dernier, un MO de RO (artéfact 2).

Le tableau 3.1 ci-dessous permet de préciser les deux artéfacts, leur nature et le cadre méthodologique qui leur correspond. Il est important de noter que le cadre méthodologique d'O'Keefe (2014) chapeaute l'ensemble de la recherche qui vise à produire l'artéfact 1 soit l'OAD, tandis que le cadre méthodologique de Norbert, Ouellet et Parent (2009) permet de produire l'artéfact 2, soit le modèle d'optimisation de RO qui est une composante centrale de l'OAD.

Tableau 3-1 Classification des artéfacts et cadres méthodologiques associés

| Artéfact         | Nature de l'artéfact                           | Cadre méthodologique              |
|------------------|------------------------------------------------|-----------------------------------|
| Artéfact 1 : OAD | OAD incorporant le modèle d'optimisation de RO | O'Keefe (2014)                    |
| Artéfact 2 : MO  | Modèle d'optimisation de RO                    | Norbert, Ouellet et Parent (2009) |

Ci-dessous, le Tableau 3.2 présente notre cadre méthodologique agrégé, qui combine le cadre méthodologique d'O'Keefe (2014) à celui de Norbert, Ouellet et Parent (2009). Le tableau présente toutes les étapes qui ont été suivies. À droite du tableau, nous avons précisé les méthodes et outils qui ont été employés tout au long de la recherche, quelles que soient les étapes des deux cadres méthodologiques mis en œuvre dans la recherche.

Tableau 3-2 Correspondance des étapes méthodologiques entre les cadres méthodologiques d'O'Keefe (2014) et de Norbert, Ouellet et Parent (2009)

| Pour produire l’OAD (artéfact 1)                                          |                                                             | Pour produire le modèle d’optimisation de RO au cœur de l’OAD (artéfact 2) |                                                                                                    | Méthodes et outils employés  |
|---------------------------------------------------------------------------|-------------------------------------------------------------|----------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|------------------------------|
| Cadre méthodologique de la RSD pour la RO orientée design: O’Keefe (2014) |                                                             | Cadre méthodologique pour la RO : Norbert, Ouellet et Parent (2009)        |                                                                                                    |                              |
| Étapes                                                                    | Objectif                                                    | Étapes                                                                     | Objectif                                                                                           |                              |
| 1. Identification du problème                                             | Définir le problème à résoudre dans un contexte spécifique. | 1. Détection d’un problème                                                 | Identifier la présence d'un problème ou d'une opportunité qui nécessite une attention particulière | Diagramme d'Ishikawa, QQQCCP |
| 2. Définition des objectifs pour une solution                             | Formuler des objectifs clairs pour orienter le              |                                                                            |                                                                                                    |                              |

|                                      |                                                                                                                                                                                      |                            |                                                                                                                            |                                                                                                                                                                                    |
|--------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------|----------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                      | développement de la solution.                                                                                                                                                        |                            |                                                                                                                            |                                                                                                                                                                                    |
| 3. Design et développement           | Concevoir et créer des artefacts (modèles, processus, systèmes) pour répondre au problème.                                                                                           | 2. Formulation du problème | Définir clairement la problématique en termes précis et mesurables                                                         | Représentation du problème sous forme de graphe                                                                                                                                    |
|                                      |                                                                                                                                                                                      | 3. Élaboration d'un modèle | Concevoir un modèle théorique ou opérationnel permettant de représenter le problème                                        | Représentation mathématique du modèle. Algorithme de résolution exacte: Cplex. Langage de programmation : Python et bibliothèque DocPlex                                           |
|                                      |                                                                                                                                                                                      | 4. Collecte de données     | Réunir les informations nécessaires pour alimenter et valider le modèle                                                    | Entrevues non dirigées, canevas de donnée, histoires utilisateur                                                                                                                   |
|                                      |                                                                                                                                                                                      |                            |                                                                                                                            | Développement de l'interface utilisateur: Python et Excel                                                                                                                          |
| 4. Test, démonstration et évaluation | Valider l'efficacité de l'artefact en le testant dans un environnement pertinent et évaluer l'utilité de l'artefact en analysant ses performances en fonction des objectifs définis. | 5. Résolution du modèle    | Valider la capacité du modèle à résoudre des problèmes et à produire des solutions                                         | Méthode d'évaluation des risques techniques et de l'efficacité ( <i>Technical Risk &amp; Efficacy Evaluation</i> ): Résolution d'instances produites avec des données synthétiques |
|                                      |                                                                                                                                                                                      | 6. Validation du modèle    | Vérifier que les solutions proposées sont cohérentes avec les objectifs initiaux et réalistes dans le contexte du problème | Méthode d'évaluation des risques humains et de l'efficacité ( <i>Human Risk &amp; Effectiveness Evaluation</i> ): Résolution d'instances                                           |

|                                 |                                                                                |                                 |                                                                                                            |                                                         |
|---------------------------------|--------------------------------------------------------------------------------|---------------------------------|------------------------------------------------------------------------------------------------------------|---------------------------------------------------------|
|                                 |                                                                                | 7. Mise en œuvre de la solution | Appliquer la solution choisie et en évaluer les impacts pour assurer qu'elle répond aux besoins identifiés | correspondant au cas des histoires utilisateur          |
| 5. Généralisation de la théorie | Étendre les résultats et les conclusions pour contribuer au domaine théorique. |                                 |                                                                                                            | Analyse de la complexité et des limites de notre modèle |

### Identification du problème et motivation

L'étape d'identification du problème et motivation consiste à analyser le problème dans son environnement ainsi que d'identifier les parties prenantes concernées. Lors de cette étape, une ou plusieurs théories sont choisies pour ultérieurement être mises à l'épreuve pour soutenir la conception et le développement de l'artéfact. De plus, il est important d'intégrer une évaluation des coûts liés à la collecte des données nécessaires à la réalisation des différentes étapes de recherche, afin de vérifier leur précision et leur validité, ainsi que d'anticiper les changements à apporter au processus actuel dans lequel se manifeste le problème. Cette étape inclut également l'identification des utilisateurs finaux de la solution et les caractéristiques de l'artéfact développé pour répondre au problème. Cette première étape est similaire à celle de la première étape du cadre méthodologique de RO proposé par Nobert, Ouellet et Parent (2009), appelée détection du problème. Cette étape vise également à reconnaître l'existence d'un problème ou d'une opportunité nécessitant une attention particulière. En conséquence, il n'a pas été nécessaire d'ajouter une sous-activité spécifique tirée du cadre méthodologique de RO proposé par Nobert, Ouellet et Parent (2009).

### Définition des objectifs

L'étape de définition des objectifs pour une solution consiste à identifier clairement les bénéfices que l'artéfact 1 doit apporter pour répondre au problème initial. Cela inclut aussi l'identification des futurs utilisateurs et des changements à apporter au processus actuel qui seront nécessaires pour mettre en œuvre la solution. Enfin, il est essentiel de spécifier les limites de la solution envisagée ainsi que les

situations où ces limites pourraient être dépassées, afin de mieux cadrer les attentes et les usages potentiels.

Dans le cadre de cette recherche, les deux premières étapes, à savoir (1) l'identification du problème et motivation et (2) la définition des objectifs pour une solution, ont été réalisées à partir d'une analyse approfondie des effets de la mise en œuvre du plan directeur du Parc Jean-Drapeau sur la mobilité. Les résultats sont présentés dans le chapitre 1 qui détaille la problématique. Ces étapes ont permis de définir clairement le problème à résoudre, ainsi que d'établir des objectifs pour guider le développement de la solution. Pour structurer la résolution de problème, nous avons utilisé la méthode du QQOCQCCP et le Diagramme d'Ishikawa comme préconisés par Johannesson et Perjons (2014).

La méthode du QQOCQCCP propose un questionnement structuré autour de 7 grandes questions afin de « faire parler » le terrain et d'identifier un certain nombre de problèmes. Par la suite, le Diagramme d'Ishikawa (ou diagramme de cause et effet) a permis d'identifier les causes sous-jacentes aux problèmes identifiés. Cette analyse a révélé un manque d'efficacité dans les stratégies de transport de marchandises au Parc Jean-Drapeau, lié à un accès limité au site et à un réseau routier restreint, réduisant les options de déplacement. Par ailleurs, la volonté de la SPJD de réduire les surfaces minéralisées a guidé la définition d'une solution centrée sur la consolidation des flux logistiques dans un réseau limité.

Pour concevoir l'artéfact 2 (le modèle d'optimisation au cœur de l'OAD), nous avons étudié les théories antérieures applicables et choisi de nous inspirer du modèle de BPP tel que défini par Crainic et *al.* (2021), au contexte de la logistique urbaine du parc. Cette démarche vise à optimiser le transport tout en répondant aux besoins des futurs utilisateurs, principalement la SPJD et les parties prenantes impliquées dans les opérations logistiques.

Nous avons également convenu que l'objectif de l'OAD (artéfact 1) est d'optimiser les opérations logistiques du Parc Jean-Drapeau en réduisant les coûts et en améliorant l'efficacité, tout en répondant aux contraintes environnementales et structurelles, notamment la réduction des surfaces minéralisées. L'OAD est conçu pour soutenir les gestionnaires dans la simulation et l'évaluation de différentes stratégies logistiques, leur permettant de choisir celles qui répondent le mieux aux objectifs opérationnels. Les utilisateurs ciblés incluent la SPJD et les équipes opérationnelles responsables des flux logistiques et des opérations de maintenance. L'intégration de l'OAD nécessitera une formation adaptée pour garantir une



utilisation optimale et des ajustements des processus logistiques, tels que la consolidation des flux dans des hubs spécifiques.

## **Design et développement**

La phase de design et développement vise à concevoir et à développer (1) l'OAD et (2) le MO sous-jacent, et à spécifier les processus et données associés. Pour concevoir et développer ces deux, quatre sous-étapes ont été réalisées de manière itérative.

Les trois premières sous-étapes suivent le cadre méthodologique de RO de Norbert, Ouellet et Parent (2009) et concernent le design et développement du MO (artéfact 2).

La première sous-étape est la formulation du problème, qui consiste à définir clairement la problématique à l'aide de termes précis et mesurables. Pour cette étape, nous avons choisi de représenter notre problème de transport sous forme de graphe, une structure mathématique composée de nœuds et d'arcs. Dans ce contexte, les nœuds représentent les points clés du réseau, tels que les lieux de départ, les destinations ou les hubs logistiques, tandis que les arcs symbolisent les connexions entre ces points, comme les itinéraires possibles pour le transport. Cette représentation permet de visualiser le problème, de formaliser les relations entre les différentes composantes, et de poser les bases de la modélisation mathématique qui suivra.

La deuxième sous-étape est l'élaboration du MO qui est conçu en fonction des besoins identifiés. À partir de la représentation du problème sous forme de graphe réalisé à l'étape précédente, nous avons défini mathématiquement les composantes principales du modèle : les variables de décision, la fonction objectif et les contraintes. Les variables de décision représentent les éléments sur lesquels portent les choix à effectuer pour résoudre le problème, tels que l'affectation des véhicules aux marchandises ou la sélection des itinéraires à emprunter. La fonction objectif, calculée en fonction des variables de décision, permet d'assigner un coût à chaque solution possible, guidant ainsi la recherche de la meilleure alternative. Dans notre cas, elle vise à minimiser les coûts logistiques, notamment les distances parcourues, les temps de transport et les émissions de carbone, tout en respectant les objectifs opérationnels et environnementaux. Les contraintes imposent des limites ou des conditions pour garantir la faisabilité des solutions.

La troisième sous-étape est la collecte des données, qui implique l'acquisition des données nécessaires à la paramétrisation et à la validation du modèle, en veillant à leur précision et leur fiabilité. Pour résoudre notre MO et valider les performances de notre OAD, il est essentiel de produire des instances de problème à résoudre, permettant à la fois de tester les limites du modèle et de résoudre des problèmes concrets. Une instance correspond à un ensemble de données associant une valeur à chacune des variables de décision du modèle afin de produire une solution.

Initialement, nous avons défini le format de données nécessaire pour générer les instances de problème et transmis ces spécifications aux services informatiques du parc. L'objectif était d'accéder aux données pertinentes déjà collectées, tout en minimisant les données manquantes ou superflues. Pour faciliter le transfert d'information et guider les futures collectes de données nécessaires à l'utilisation de l'outil, nous avons conçu un classeur Excel structuré selon la logique d'une base relationnelle, représentant les différents champs de données pertinents par entités.

Cependant, il s'est avéré impossible de collecter des données réelles pour notre problématique initiale. En conséquence, nous avons opté pour la génération de données synthétiques afin de tester la robustesse technique du modèle. Ces données ont été conçues pour refléter la réalité du problème, en s'appuyant sur des critères tels que la capacité des véhicules, les volumes de marchandises et les itinéraires possibles. L'outil de collecte de données a été particulièrement utile pour encadrer le format des données générées, en assurant leur conformité avec les besoins du modèle.

Comme l'a mentionné O'Keefe (2014), les données synthétiques, bien qu'utiles dans un premier temps, sont souvent insuffisantes dans le cadre la RSD en RO. En effet, elles ne permettent pas de démontrer pleinement l'utilité et la pertinence de l'artéfact dans un environnement d'usage où les interactions avec les utilisateurs et les spécificités contextuelles jouent un rôle déterminant. Sans cette contextualisation, le modèle risque de rester une abstraction théorique, dépourvue d'applicabilité pratique. Cela nous a conduits à effectuer une seconde collecte de données, cette fois axée sur des scénarios réalistes et orientée vers la validation pratique.

Par ailleurs, trois entrevues non dirigées (Mayer *et al.*, 2000) ont été menées sur le site avec quatre membres du personnel de la SPJD pour identifier les informations pouvant orienter et affiner la génération de ces données synthétiques. Ces trois entrevues non dirigées (Mayer *et al.*, 2000) ont permis de mieux

comprendre la topologie du site et de préciser les dynamiques de déplacement entre les différentes prenantes qui utilisent les corridors de transport.

Enfin, la quatrième sous-étape, propre à la RSD, consiste à concevoir l'artéfact 1, soit l'OAD. Dans notre cas, cette étape implique la définition précise des fonctionnalités de l'OAD, ses composantes, ainsi que l'interface utilisateur. Cette interface, pensée pour être intuitive et accessible, permettra aux utilisateurs d'interagir efficacement avec l'outil, de saisir les données nécessaires et d'exploiter les résultats générés par le modèle d'optimisation pour soutenir la prise de décision. Une approche par prototypage (Braunstein *et al.*, 1991) a été suivie pour cette quatrième sous-étape. Cette approche a nécessité plusieurs itérations qui ont résulté par la réalisation de tests et démonstrations auprès des utilisateurs de l'OAD.

### **Test, démonstration et évaluation**

La phase de test, démonstration et évaluation a pour objectif de valider la composante modèle d'optimisation de l'OAD (artéfact 2) ainsi que l'artéfact principal, l'OAD (artéfact 1) dans son ensemble. Elle vise à garantir leur bon fonctionnement, tout en démontrant leur efficacité et leur pertinence auprès des utilisateurs, propriétaires du problème et autres parties prenantes. Les étapes de conception et développement, de test et démonstration, ainsi que d'évaluation ont été réalisées de manière itérative, permettant une amélioration progressive de l'artéfact pour qu'il réponde efficacement à la problématique identifiée.

Pour tester et démontrer la composante MO de l'OAD, nous avons suivi les deux étapes définies par Norbert, Ouellet et Parent (2009). La première étape, de résolution du modèle d'optimisation, vise à tester la capacité et l'efficacité du modèle à produire des solutions cohérentes.

La deuxième étape est la validation du modèle d'optimisation, réalisée à partir de la seconde collecte de données, permettant d'élaborer des cas d'utilisation réalistes. Cette étape poursuit deux objectifs principaux : démontrer le système aux utilisateurs, propriétaires du problème et autres parties prenantes et valider l'efficacité globale du système dans un contexte opérationnel. En testant le modèle dans des environnements proches de la réalité, cette approche assure que la composante MO de l'OAD n'est pas seulement techniquement valide, mais qu'elle est également pertinente, utilisable et efficace dans des contextes opérationnels complexes.

La première étape, soit la résolution du MO a été réalisée au cours de la première itération, qui avait pour objectif principal de développer et de tester le modèle d'optimisation au cœur de l'OAD. Durant cette itération, nous avons soumis la composante modèle, élaborée lors de la phase de design et développement, à des tests visant à évaluer son efficacité technique et sa capacité à produire des solutions cohérentes. Pour ce faire, nous avons mobilisé la méthode d'évaluation des risques techniques et de l'efficacité (*Technical Risk & Efficacy Evaluation*) proposée par Venable *et al.*, (2016). Cette méthode est particulièrement adaptée pour tester la robustesse technique et la capacité fonctionnelle d'un artéfact dans un environnement contrôlé. En utilisant des données synthétiques, elle permet d'identifier d'éventuelles limitations techniques, telles que des incohérences dans les résultats ou des inefficacités dans l'algorithme, tout en offrant une base solide pour affiner le modèle. En pratique, nous avons généré des instances réalistes de problèmes de transport à résoudre, afin d'évaluer à la fois la qualité des solutions proposées par le modèle et sa vitesse de résolution. Par ailleurs, nous avons également exploré les limites du modèle en testant son comportement face à des problèmes correspondant aux valeurs extrêmes des paramètres, notamment en ce qui concerne la quantité de marchandises et leur taille.

Cependant, comme l'a mentionné O'Keefe (2014), les données synthétiques, bien qu'utiles dans un premier temps, sont souvent insuffisantes dans le cadre la RSD en RO. En effet, elles ne permettent pas de démontrer pleinement l'utilité et la pertinence de l'artéfact dans un environnement d'usage où les interactions avec les utilisateurs et les spécificités contextuelles jouent un rôle déterminant. Sans cette contextualisation, le modèle risque de rester une abstraction théorique, dépourvue d'applicabilité pratique.

La deuxième étape, la validation du modèle d'optimisation, repose sur l'utilisation de données issues de la seconde phase de collecte, nous permettant de concevoir des cas d'utilisation réalistes. Cette étape a pour but de démontrer l'OAD aux parties prenantes et de valider l'efficacité du MO dans des environnements proches de la réalité. Dans le cadre de notre travail, la validation de la composante MO de l'OAD constitue un sous-produit de l'évaluation globale de l'OAD, présentée plus bas. Cette validation a été effectuée lors de la seconde itération, axée sur le développement et l'évaluation de l'OAD dans son ensemble.

Pour évaluer l'OAD dans son ensemble, nous avons appliqué la méthode d'évaluation des risques humains et de l'efficacité (*Human Risk & Effectiveness Evaluation*) proposée par Venable *et al.* (2016). Cette

méthode se concentre sur l'interaction entre les utilisateurs et l'artéfact, en évaluant sa facilité d'utilisation, son efficacité et son potentiel d'adoption. En mobilisant des scénarios concrets et des retours d'utilisateurs dans un environnement semi-contrôlé, elle garantit que l'OAD satisfait aux exigences pratiques tout en facilitant l'engagement des parties prenantes. Les retours obtenus permettent d'identifier les points de friction et d'ajuster l'OAD pour en améliorer l'utilité et l'acceptabilité.

Dans ce cadre, nous avons élaboré deux cas d'utilisation concrets, en collaboration avec les parties prenantes, pour représenter des problématiques réelles de transport de marchandises dans des microcorridors. Ces cas d'utilisation visaient à tester les capacités du modèle à résoudre des problèmes de transport directement liés aux besoins actuels des opérations, mieux documentés que la problématique initiale du transport entre hubs. Afin de concevoir ces cas d'utilisation, nous avons mobilisé trois histoires utilisateur, illustrant l'usage potentiel de notre OAD selon trois scénarios différents.

Les histoires utilisateur sont des descriptions courtes et centrées sur les besoins des utilisateurs, conçues pour capturer de manière concise leurs objectifs et attentes. Elles se distinguent par plusieurs caractéristiques essentielles, qui en font un outil privilégié dans des méthodologies orientées utilisateur, telles que la méthode d'évaluation des risques humains et de l'efficacité. Une histoire utilisateur est formulée du point de vue de l'utilisateur final, en mettant en avant ce qu'il souhaite accomplir et pourquoi (Cohn, 2004). Cette simplicité favorise une compréhension commune entre les équipes de conception et les parties prenantes, tout en restant suffisamment ouverte pour permettre des discussions collaboratives visant à affiner les besoins et les solutions proposées (Cohn, 2004). De plus, les histoires utilisateur sont accompagnées de critères d'acceptation, définissant les conditions nécessaires pour qu'elles soient considérées comme accomplies, garantissant ainsi leur clarté et leur vérifiabilité. Les histoires utilisateur ont également l'avantage de contrer le biais du statu quo, une difficulté souvent rencontrée lorsqu'il s'agit de faire accepter une solution innovante. En fournissant des scénarios réalistes et compréhensibles, elles facilitent les échanges avec les parties prenantes et démontrent de manière tangible la pertinence de l'artéfact (Muratovski et Vogel, 2019). Ces caractéristiques expliquent pourquoi elles sont particulièrement pertinentes dans le cadre de la méthode d'évaluation des risques humains et de l'efficacité. Cette démarche visait dans un premier temps à renforcer leur compréhension de l'outil pour faciliter le dialogue et identifier les problématiques pouvant être résolues avec l'OAD. Elle a abouti à l'identification et à la documentation de deux cas d'utilisation concrets, qui ont servi à évaluer les performances de notre modèle dans la résolution de problèmes spécifiques liés au transport. Le premier

cas porte sur le transport de personnel entre les différents bureaux utilisés par la SPJD et le deuxième porte sur le transport de marchandises pour l'exécution de tâches de maintenance récurrentes au sein du parc.

En combinant les deux méthodes d'évaluation, cette approche itérative garantit que l'artéfact est non seulement techniquement robuste, mais également pertinent et utilisable dans des contextes opérationnels réels. Si des ajustements sont nécessaires, ces méthodes permettent de revenir aux étapes de conception et de développement pour affiner davantage l'outil, assurant la production d'un artéfact capable de répondre efficacement à la problématique, tout en maximisant son utilité et son acceptabilité.

### **Généralisation de la théorie**

Enfin, la généralisation de la théorie prescriptive vise à évaluer si l'artéfact développé peut être transféré à d'autres contextes similaires, permettant ainsi d'étendre son utilité au-delà du cadre spécifique du Parc Jean-Drapeau. Cette étape est essentielle pour renforcer la valeur scientifique de l'outil en le positionnant comme une solution applicable à des problèmes logistiques rencontrés dans d'autres environnements présentant des problèmes de transport dans des microcorridors de transport. En identifiant les éléments modulables et les principes fondamentaux du modèle, il devient possible d'adapter l'OAD à des scénarios variés, tout en maintenant son efficacité et sa pertinence. Cette démarche contribue également à produire de nouvelles connaissances prescriptives, enrichissant le domaine de la science du design et de la RO. Cependant, cette dernière étape mériterait d'autres développements qui vont au-delà de la présente recherche.

### **3.2 Résumé de la démarche de recherche**

La figure 3.1 illustre les différentes étapes de l'approche méthodologique adoptée. Les deux premières étapes (1) l'identification du problème et des motivations, et (2) la définition des objectifs pour une solution ont été présentées au chapitre I, puis résumées dans le présent chapitre. Les résultats de la troisième étape, soit le design et le développement, ainsi que ses différentes sous-activités (3.1), la formulation du problème, (3.2) l'élaboration du modèle, (3.3) la collecte de données, et (3.4) la conception de l'artéfact sont présentées au chapitre IV.

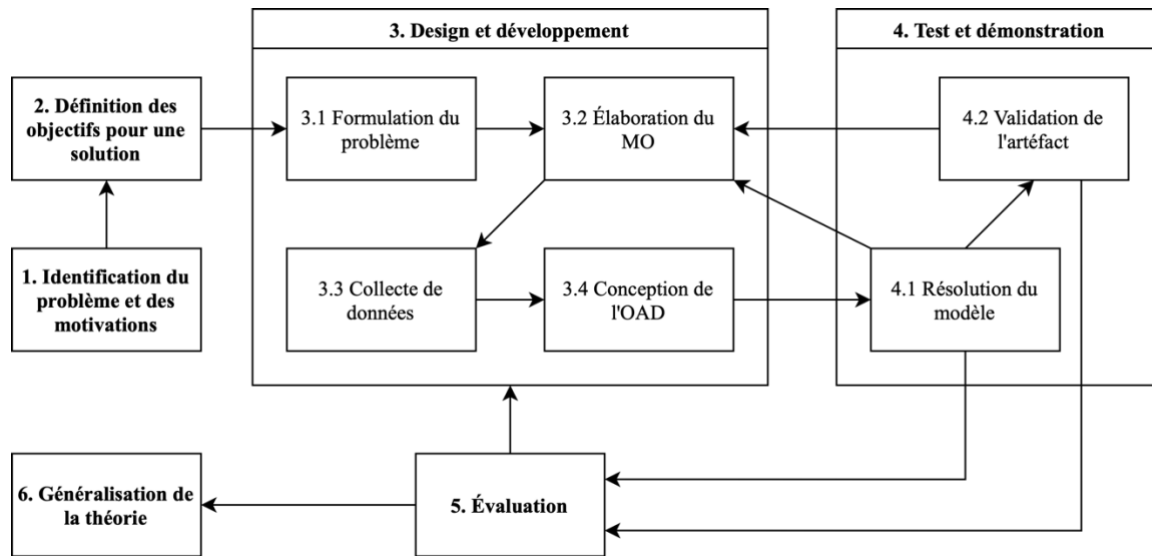


Figure 3-1 Résumé des étapes du cadre méthodologique suivie

Les résultats de la quatrième étape, soit les tests et démonstrations, ainsi que ses sous-activités (4.1), la résolution du modèle et (4.2) la validation de l'artéfact sont détaillées au chapitre V. La cinquième étape, l'évaluation, réalisée en parallèle de la quatrième, est également abordée dans ce même chapitre. Enfin, la sixième étape, soit la généralisation de la théorie, est traitée dans la discussion qui conclut le chapitre V.

## **CHAPITRE 4**

### **DESIGN ET DÉVELOPPEMENT**

Dans ce chapitre, nous présentons les résultats des étapes de design et développement réalisées pour les deux artéfacts : le MO et l'OAD. Ces étapes correspondent à la troisième phase du cadre méthodologique adopté, combinant le cadre de RSD (O'Keefe, 2014) et celui de la RO (Norbert, Ouellet et Parent, 2009).

Le design et développement du MO est détaillé dans un premier temps à travers trois parties correspondant aux étapes de formulation du problème (4.1), élaboration du modèle (4.2), et collecte des données (4.3). Ce modèle, qui constitue le cœur de l'OAD, a été conçu pour répondre aux besoins logistiques propres aux microcorridors de transport, en s'appuyant sur une formalisation mathématique robuste et adaptée aux spécificités du contexte du Parc Jean-Drapeau.

La quatrième section, la conception de l'artéfact (4.4), présente le design et développement de l'OAD. Cette étape comprend la définition des fonctionnalités principales de l'OAD, le développement de son interface utilisateur, et la préparation des outils nécessaires à sa mise en œuvre. Une approche par prototypage a été adoptée pour garantir que l'OAD répond efficacement aux attentes des utilisateurs finaux et qu'il s'intègre harmonieusement dans leur environnement opérationnel.

Ainsi, ce chapitre détaille les processus de conception de chacun des artéfacts, en montrant comment ils s'articulent pour répondre aux objectifs fixés dans le cadre méthodologique global.

#### **4.1 Formulation du problème**

Pour chacun des problèmes de transport du parc étudié, nous pouvons représenter le réseau de transport sous la forme d'un graphe. C'est à partir de la modélisation sous forme de graphe que nous serons en mesure de formaliser les différentes composantes de notre modèle. La figure 4-1 représente les lieux de chargement et de déchargement, ainsi que les déplacements possibles au sein de notre réseau de transport, dans le cadre du problème de transport de marchandises entre les trois hubs logistiques, tels qu'ils apparaissent sur la carte du site. La figure 4-2 présente cette situation sous la forme d'un graphe, qui servira à modéliser le problème. Les nœuds correspondent aux différents lieux de chargement et de déchargement, et les arcs, aux différents chemins les reliant.



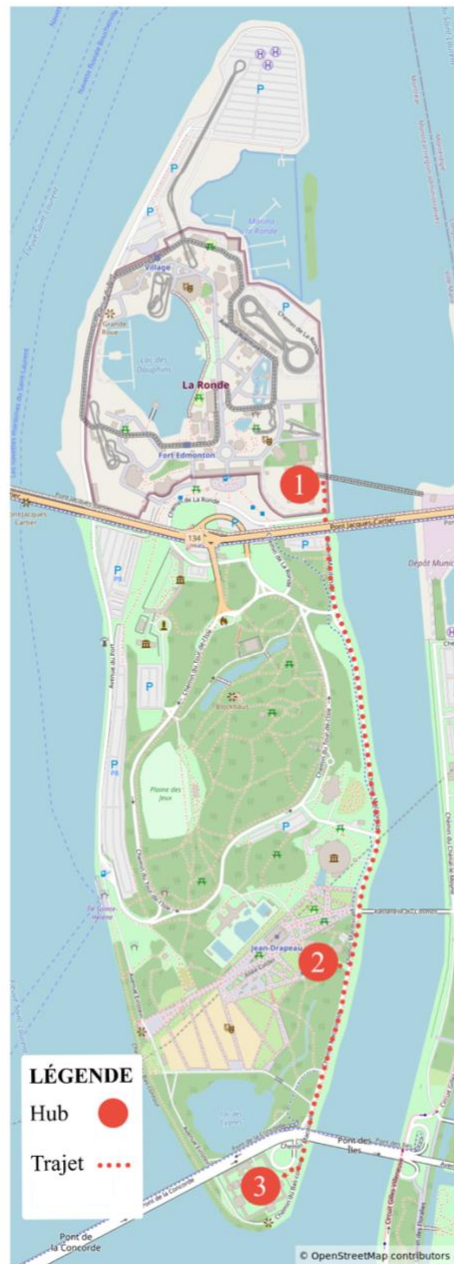


Figure 4-1 Position des points de livraisons potentiels (Hubs) sur la carte du Parc Jean-Drapeau

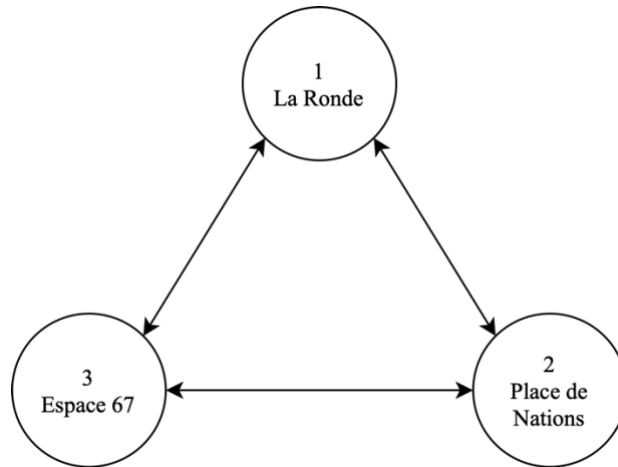


Figure 4-2 Représentation schématique des points de livraison du parc Jean-Drapeau

L'objectif de l'OAD est de produire un plan d'opération indiquant, pour chaque période, quelle marchandise (*item*) charger dans quel véhicule (*bin*). Les véhicules (*bin*) permettent de déplacer les marchandises (*item*) d'un lieu à l'autre du parc Jean-drapeau. Les *items* sont caractérisés par leur taille, identifiée par le volume et le poids, ainsi que par des exigences particulières en matière de transport (dans notre cas, réfrigération et fragilité). Les *items* sont également caractérisés par des attributs temporels, soit (1) la disponibilité, lorsque l'article est disponible pour l'expédition ; (2) l'instant limite de livraison, indiquant le moment où l'envoi devrait être à destination ; (3) la livraison au plus tôt, lorsqu'il est possible de livrer plus tôt, éventuellement moyennant un coût de pénalité ; et (4) la livraison au plus tard, le moment au-delà duquel la livraison est devenue inacceptable, même encourant une pénalité. Les *items* possèdent également un attribut économique, soit le coût associé à la livraison de l'item au moyen d'une ressource externe. Finalement les *items* sont caractérisés par des attributs indiquant leur position initiale dans le réseau (représenté par les nœuds 1, 2 et 3 sur la figure 4-2), ainsi que la position finale (le lieu de livraison).

Les *bins* sont caractérisés par leurs capacités, identifiées par leurs volumes et leurs poids maximaux, ainsi que par des compétences particulières en matière de transport (dans notre cas, être en mesure de transporter des *items* réfrigérés ou fragiles). Les *bins* sont également caractérisés par un attribut correspondant à leur position initiale dans le réseau. Dans notre version du problème, nous avons posé comme hypothèse que chaque véhicule retourne à son nœud de départ à chaque nouvelle période.

Finalement, les *bins* sont représentés par des attributs économiques, soit (1) le coût fixe d'utilisation, indiquant le coût associé à la sélection d'un véhicule pour un trajet; et (2) le coût variable d'utilisation indiquant le coût d'utilisation du véhicule selon la distance parcourue. Le problème se déroule sur plusieurs périodes de durée égales. Toutefois, il importe de souligner que la notion de coût, dans ce contexte, dépasse les seuls aspects économiques : elle peut également modéliser des éléments supplémentaires tels que les délais de livraison, les émissions de GES générées par certains véhicules, ou encore leur niveau de sécurité pour assurer des transports en milieu urbain. Le problème se déroule sur plusieurs périodes de durée égales, dont la somme correspond à la durée totale à optimiser.

Pour le présent cas, l'artéfact se présente comme un OAD conçu pour optimiser le transport en limitant le nombre de véhicules circulant sur le site. Cet outil repose sur un modèle d'optimisation de type « Bin packing » implémenté à l'aide de CPLEX, un algorithme de résolution exact. Le fonctionnement sera le suivant : on produit une instance d'un problème de livraison de marchandise pour un horizon temporel donné et on utilise l'OAD pour obtenir une solution optimale. Une instance est composée des éléments suivants : la liste des *items* à livrer et la liste des *bins* disponibles. Les *items* et les *bins* sont caractérisés par un lot d'attribut, tels que leur poids et leur volume, qui devront être préalablement définis de manière à correspondre à la situation sur le terrain. Une solution définit le chargement/déchargement optimal de chaque bin pour chaque nœud qu'il empruntera sur son trajet pour la période étudiée.

Tableau 4-1 Niveau d'activité selon la période

| Période d'intérêt   | Niveau d'activité |
|---------------------|-------------------|
| Hivernale           | Faible            |
| Estivale            | Moyen-Élevé       |
| Estivale/Festival   | Élevé             |
| Estivale/Grand-Prix | Très élevé        |

Nous avons également déterminé que le volume et la nature des marchandises transportées au sein du parc variaient grandement selon la période et le problème de transport particulier qui est considéré. Présentement, la période hivernale représente une période creuse, alors que la période estivale engendre beaucoup de déplacement de marchandises. Le volume et la nature des marchandises à transporter durant la période estivale sont d'ailleurs fortement affectés par les événements ayant lieu au parc, comme le Grand-Prix de Montréal et les festivals musicaux. Il sera ainsi important d'être en mesure de générer des instances représentatives pour chacune de ces périodes d'intérêts. Les périodes d'intérêts déjà identifiés sont représentées dans le tableau 4-1.

Avant de schématiser le problème, nous allons formaliser la définition d'un microcorridor de transport. Nous définissons un microcorridor de transport comme étant un système ayant des points d'entrée et de sorties limitées et comportant une densité de nœuds importante (correspondant aux différents points dont il faut gérer les livraisons). Un exemple simple de microcorridor de transport pourrait être une rue commerciale qui n'est accessible pour les camions de livraison que par les extrémités. Les différents commerces forment les nœuds du système et les voies d'accès et de sortie sont limitées. Le concept de microcorridor n'est pas encore décrit formellement dans la littérature scientifique, mais on peut en retrouver des traces dans la littérature grise, principalement sur les sites des plateformes offrant des services logistiques pour ce type d'environnement comme *Envoie Montréal* (<https://www.cqcd.org/envoi-montreal/>).

Chaque problème de transport possède un horizon temporel, correspondant à la perspective de temps durant laquelle le problème peut être résolu. Cet horizon est divisé en un certain nombre de périodes de durée égale et la somme des périodes équivaut à l'horizon temporel. Chaque véhicule est assigné à un trajet. Ce trajet correspond à la séquence de nœud que le véhicule va parcourir et se déroule à l'intérieur d'une période. Chaque trajet correspond à un microcorridor de transport potentiel. Une fois qu'un véhicule (bin) est affecté à un trajet pour une période donnée, la décision ne consiste pas à déterminer chaque segment de route indépendamment, mais plutôt à choisir un trajet complet (c'est-à-dire la séquence de nœuds à parcourir) et la période où ce trajet sera effectué. Dans ce contexte, on décide également, pour chaque nœud visité (départ, intermédiaires et destination), quels items sont chargés ou déchargés afin de respecter les contraintes de capacité. Autrement dit, la décision d'affecter un bin à un trajet englobe à la fois le choix du chemin et la gestion des flux de marchandises tout au long de ce chemin, pour une période précise.

Nous pouvons observer que les trois nœuds sont disposés en cycle. Pour chaque nœud, il suffit de suivre le cycle pour visiter tous les nœuds et revenir au point de départ. Notre réseau de transport est représenté sous la forme d'un graphe cyclique bidirectionnel  $G(N,A)$  où  $N$  comprend l'ensemble des nœuds et  $A$  l'ensemble des arcs (voir figure 4-3).

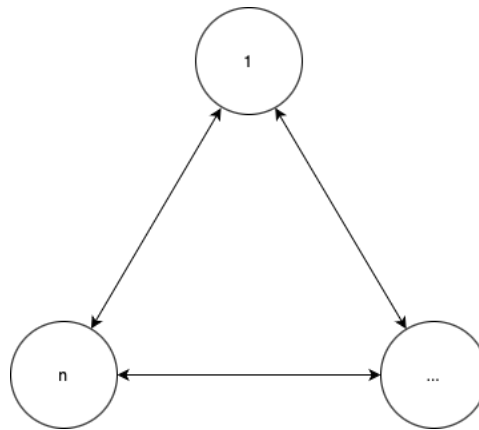


Figure 4-3 Graphe cyclique bidirectionnel

Pour un graphe cyclique bidirectionnel comportant  $n$  nœuds, nous avons  $2n^2$  trajets possibles. Le trajet le plus long considéré correspond à un cycle complet du graphe. Pour un tel trajet, l'ensemble des nœuds du graphe est parcouru de façon à terminer le trajet au nœud de départ (le nœud de départ est égal au nœud de destination). Dans un graphe comportant  $n$  nœuds, cela correspond à un trajet de longueur  $n$ . Pour chaque longueur de trajet de 1 à  $n$ , nous avons  $2n$  trajets différents. Le graphe étant bidirectionnel, chaque nœud permet de générer 2 trajets, soit un trajet par direction.  $n$  longueurs de trajets générant chacun  $2n$  trajets différents nous donne un total de  $2n \times n$  trajets possibles, soit  $2n^2$ . Ainsi, pour un réseau comportant trois nœuds, nous avons  $18 (2 \cdot 3^2)$  trajets possibles.

Les trajets peuvent être divisés en deux types : les trajets de cardinalité 2, qui comportent uniquement un nœud de départ et un nœud de destination et les trajets de cardinalité supérieurs à 2, qui comporte un nœud de départ, un nœud de destination et au moins un nœud intermédiaire. Un véhicule peut charger des marchandises au nœud de départ et au(x) nœud(s) intermédiaire(s) et décharger des marchandises

au nœud de destination et au(x) nœud(s) intermédiaire(s). Les déplacements possibles sont résumés dans le tableau 4-2

Tableau 4-2 Déplacements possibles selon la cardinalité du trajet

| Cardinalité du trajet | Actions au nœud de départ | Actions au(x) nœud(s) intermédiaire(s) | Actions au nœud de destination |
|-----------------------|---------------------------|----------------------------------------|--------------------------------|
| $card(p) = 2$         | Chargement                |                                        | Déchargement                   |
| $card(p) \geq 3$      | Chargement                | Chargement,<br>Déchargement            | Déchargement                   |

Chaque type de trajet peut être représenté comme un graphe linéaire. La figure 4-4 représente les trajets de cardinalité 2 et la figure 4-5 représente les trajets de cardinalité supérieurs à 2. Pour un trajet  $p$ , le nœud de départ est donné par la fonction  $P_{deb}(p)$  et le nœud de destination est donné par la fonction  $P_{fin}(p)$ .

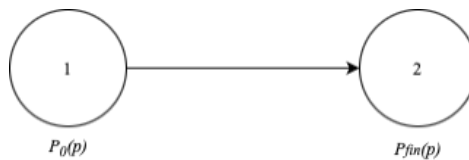


Figure 4-4 Trajet de cardinalité 2

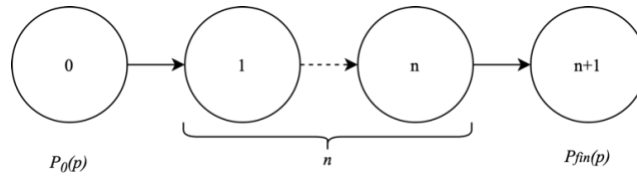


Figure 4-5 Trajet de cardinalité supérieure à 2

Avant d’élaborer notre modèle, nous avons dû poser les hypothèses suivantes : (1) un véhicule ne peut être assigné qu’à un seul trajet, toutes périodes confondues, (2) la position du véhicule et des marchandises dans le réseau ainsi que les différents trajets possibles sont connues (ce sont des paramètres et non des variables) et (3) chaque trajet peut être effectué, en incluant les opérations de chargement et déchargement, à l’intérieur d’une période.

Le problème de transport de marchandises au parc Jean-Drapeau possède les caractéristiques suivantes : (1) un réseau de transport pouvant être divisé en microcorridor, (2) un horaire de livraison qui peut être décortiqué en plusieurs périodes et (3) un besoin de déplacer des *Items* (des marchandises et des humains) dans un réseau en les transportant dans des *Bin* (différents types de véhicules). Ces caractéristiques peuvent s’apparenter à celles du problème de type *Multi-period bin packing*. Dans le contexte de la logistique urbaine, le *Bin-Packing Problem* (BPP) peut être intéressant pour résoudre les problèmes de transport de marchandises dans des microcorridors de transport, plus particulièrement lorsqu’on utilise une variante prenant en compte la dimension temporelle ainsi que la variabilité des attributs des items et des bins, comme celle proposée par Crainic *et al.*, (2021).

## 4.2 Élaboration du modèle

Nous avons décidé d'adapter le modèle de BPP, tel que défini par Crainic *et al.*, au contexte de la logistique urbaine du Parc Jean-Drapeau. Ce modèle permet de résoudre un problème d'optimisation au sein d'un corridor de transport en sélectionnant les *bins* à une certaine période selon leur disponibilité. La différence principale entre le modèle de BPP dans Crainic *et al.* et le nôtre est que l’assignation des *bins* ne se fait pas uniquement selon une période, mais également selon un trajet. Pour les trajets de cardinalité supérieurs à 2, il est possible de charger et de décharger des marchandises aux nœuds intermédiaires. Nous nous

retrouvons ainsi également avec un problème de flot, car il faut s'assurer que la capacité des véhicules n'est jamais excédée pour tous les nœuds du trajet.

Le modèle est défini sur les graphes correspondant aux deux types de trajets possibles (voir figures 4-4 et 4-5). Le problème comporte trois types de variables binaires pour déterminer : (1) quelle *bin* a été assigné à quel trajet à quelle période, (2) quel *item* a été assigné à quelle *bin* à quelle période et (3) quel *item* n'a pas été livré à quelle période. Chaque véhicule doit être affecté à un trajet pour une période donnée et chaque marchandise doit être assignée à un véhicule pour un trajet et une période spécifiques. Notre objectif est de minimiser le coût total, qui comprend à la fois le coût d'assignation d'un véhicule à un trajet à une période donnée, le coût d'assignation d'un item à un véhicule et le coût de non-livraison lorsque celle-ci est inévitable.

Le tableau 4-3 résume les ensembles, fonctions et paramètres du modèle.

Tableau 4-3 Ensembles, fonctions, paramètres du modèle

| Ensembles           | Définition                                                                                                                                           |
|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| $I$                 | Ensemble des items $i$                                                                                                                               |
| $J$                 | Ensemble des bins $j$                                                                                                                                |
| $\mathcal{T}$       | Ensemble des périodes $t$                                                                                                                            |
| $\mathcal{T}_{ijp}$ | Ensemble des périodes $t$ où l'item $i$ peut être assigné à la bin $j$ en effectuant le trajet $p$ , tel que $\mathcal{T}_{ijp} \subset \mathcal{T}$ |
| $T_i$               | Ensemble des périodes où l'item $i$ est disponible pour être livré, tel que $T_i \subset \mathcal{T}$                                                |
| $\Gamma_{jp}$       | Ensemble des périodes $t$ où la bin $j$ peut être assigné à un trajet $p$ , tel que $\Gamma_{jp} \subset \mathcal{T}$                                |



|                   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| $\mathcal{N}$     | Ensemble des nœuds $n$ où $\mathcal{N} = \{1,2,3\}$                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| $\mathcal{P}$     | <p>Ensemble de tous les trajets possibles <math>p</math> reliant au moins deux nœuds du réseau étudié. Chaque trajet <math>p \in \mathcal{P}</math> correspond à une séquence ordonnée de nœuds distincts ou répétés, représentant un chemin réalisable entre un point de départ et un point d'arrivée</p> <p>Dans le problème étudié,</p> $\mathcal{P} = \{(1,2)(2,1)(1,3)(3,1)(2,3)(3,2)(1,2,3)(1,3,2)(2,3,1)(2,1,3)(3,2,1)(3,1,2)(1,3,2,1)(1,2,3,1)(2,3,1,2)(2,1,3,1)(3,2,1,3)(3,1,2,3)\}$ |
| <b>Fonctions</b>  | <b>Définition</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| $P_{deb}(p)$      | Donne le nœud de départ du trajet $p \in \mathcal{P}$                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| $P_{fin}(p)$      | Donne le nœud de destination du trajet $p \in \mathcal{P}$                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| $card(p)$         | Donne la cardinalité du trajet $p \in \mathcal{P}$                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <b>Paramètres</b> | <b>Définition</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| $v_i$             | Volume de l'item $i \in I$                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| $\underline{t}_i$ | Moment de disponibilité le plus tôt de l'item $i \in I$ , tel que $T_i = [\underline{t}_i, \bar{t}_i]$                                                                                                                                                                                                                                                                                                                                                                                        |
| $\bar{t}_i$       | Moment de disponibilité maximal de l'item $i \in I$                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| $\gamma_i$        | Moment de disponibilité le plus tôt pour la livraison de l'item $i \in I$                                                                                                                                                                                                                                                                                                                                                                                                                     |
| $\delta_i$        | Moment de livraison dû de l'item $i \in I$ , tel que $\underline{t}_i \leq \gamma_i \leq \delta_i \leq \bar{t}_i$                                                                                                                                                                                                                                                                                                                                                                             |

|                                                     |                                                                                                                                     |
|-----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| $\omega_i^t$                                        | Coût de non-livraison de l'item $i \in I$ à la période $t \in T_i$                                                                  |
| $\lambda_{it}$                                      | Position de l'item $i \in \mathcal{I}$ dans le système à la période $t \in \mathcal{T}_{ijp}$                                       |
| $o_i$                                               | Nœud de départ de l'item $i \in I$                                                                                                  |
| $d_i$                                               | Nœud de destination de l'item $i \in \mathcal{I}$                                                                                   |
| $f_j$                                               | Coût fixe d'utilisation de la bin $j \in \mathcal{J}$                                                                               |
| $\alpha_{jp}^t$                                     | Coût variable d'utilisation de la bin $j \in \mathcal{J}$ effectuant le trajet $p \in \mathcal{P}$ à la période $t \in \Gamma_{jp}$ |
| $\Gamma_j = [\underline{\Gamma}_j, \bar{\Gamma}_j]$ | Fenêtre de disponibilité de la bin $j \in \mathcal{J}$                                                                              |
| $V_j$                                               | Volume maximal (capacité) de la bin $j \in \mathcal{J}$                                                                             |
| $\Lambda_j$                                         | Position initiale de la bin $j \in \mathcal{J}$ dans le système                                                                     |
| $a_{ij}^t$                                          | Le coût d'assignation de l'item $i \in \mathcal{I}$ à la bin $j \in \mathcal{J}$ à la période $t \in \mathcal{T}_{ij}$              |

Les variables de décision sont les suivantes :

$$y_{jp}^t = \begin{cases} 1 & \text{si la bin } j \in \mathcal{J} \text{ est sélectionnée pour le trajet } p \in \mathcal{P} \\ & \text{à la période } t \in \Gamma_{jp}, \\ 0 & \text{sinon;} \end{cases}$$

$$x_{ijp}^t = \begin{cases} 1 & \text{si l'item } i \in I \text{ est assignée à la bin } j \in \mathcal{J} \text{ pour le trajet } p \in \mathcal{P} \\ & \text{est sélectionnée à la période } t \in \mathcal{T}_{ijp}, \\ 0 & \text{sinon;} \end{cases}$$

$$u_i^t = \begin{cases} 1 & \text{si l'item } i \in I \text{ n'est pas transporté à la période } t \in T_i, \\ 0 & \text{sinon.} \end{cases}$$

La fonction objectif et les contraintes sont les suivantes :

$$\text{Min}_{y, x, u} \left( \sum_{j \in J} \sum_{p \in \mathcal{P}} \sum_{t \in \Gamma_{jp}} (f_j + \alpha_{jp}^t) y_{jp}^t + \sum_{i \in I} \sum_{j \in J} \sum_{p \in \mathcal{P}} \sum_{t \in \mathcal{T}_{ijp}} a_{ij}^t x_{ijp}^t + \sum_{i \in I} \sum_{t \in T_i} \omega_i^t u_i^t \right) \quad (1)$$

$$\sum_{i \in I: o_i = P_0(p): d_i \in p} v_i x_{ijp}^t \leq V_j y_{jp}^t, \quad \forall j \in J, t \in \Gamma_j, p \in \mathcal{P}: \text{card}(p) = 2 \quad (2)$$

$$\begin{aligned} & \sum_{i \in I_{n'}: d_i \geq n: n' \leq n} v_i x_{ijp}^t + \sum_{i \in I_n: d_i \geq n} v_i x_{ijp}^t \\ & - \sum_{i \in I_{n'}: d_i = n: n' \leq n} v_i x_{ijp}^t \leq V_j y_{jp}^t, \end{aligned} \quad \begin{aligned} & \forall j \in J, t \in T, \\ & p \in \mathcal{P}: \text{card}(p) \geq 3, \\ & n \in p, n' \in p, n' \leq n, \\ & n \neq P_0(p) \wedge n \neq P_{fin}(p) \end{aligned} \quad (3)$$

$$\sum_{p \in \mathcal{P}} \sum_{j \in J} \sum_{t \in \mathcal{T}_{ijp}} x_{ijp}^t + \sum_{t \in T_i} u_i^t = 1, \quad \forall i \in I \quad (4)$$

$$\sum_{t \in \Gamma_j} \sum_{p \in \mathcal{P}} y_{jp}^t \leq 1, \quad \forall j \in J \quad (5)$$

$$y_{jp}^t \in \{0,1\}, \quad \forall j \in J, t \in \Gamma_j, p \in \mathcal{P} \quad (6)$$

$$x_{ijp}^t \in \{0,1\}, \quad \forall i \in I, j \in J, t \in \mathcal{T}_{ijp}, p \in \mathcal{P} \quad (7)$$

$$u_i^t \in \{0,1\}, \quad \forall i \in I, j \in J, t \in T_i \quad (8)$$

La fonction objectif de ce modèle vise à minimiser les coûts totaux associés aux trois décisions possibles. Elle est composée de trois volets, chacun correspondant à une des composantes de coût. Le premier volet,  $\sum_{j \in J} \sum_{p \in \mathcal{P}} \sum_{t \in \Gamma_{jp}} (f_j + \alpha_{jp}^t) y_{jp}^t$ , représente le coût d'assignation d'une bin à un trajet pour une période donnée. Il inclut le coût fixe  $f_j$  lié à l'utilisation du véhicule (bin) ainsi que le coût variable  $\alpha_{jp}^t$  associé à

l'affectation de la bin  $j$  au trajet  $p$  à la période  $t$ . La variable binaire  $y_{jp}^t$  indique si la bin  $j$  est affectée à ce trajet au cours de la période  $t$ .

Le deuxième volet,  $\sum_{i \in I} \sum_{j \in J} \sum_{p \in \mathcal{P}} \sum_{t \in T_{ijp}} a_{ij}^t x_{ijp}^t$ , correspond au coût d'assignation d'un item à une bin sur un trajet et une période donnés. Le paramètre  $a_{ij}^t$  reflète le coût propre au chargement de l'item  $i$  dans la bin  $j$ , sur le trajet  $p$  à la période  $t$ . La variable binaire  $x_{ijp}^t$  indique si l'item  $i$  est chargé dans la bin  $j$  au trajet  $p$  pendant la période  $t$ .

Le troisième volet,  $\sum_{i \in I} \sum_{t \in T_i} \omega_i^t u_i^t$ , exprime le coût de non-livraison d'un item pour l'ensemble des périodes données.

Les contraintes (2) et (3) permettent de s'assurer que la quantité d'items chargés dans chaque bin n'excède pas sa capacité. La contrainte (2) traite des trajets de cardinalité 2 et la contrainte 3 des trajets de cardinalité supérieure à 2. Dans ce modèle, un trajet est représenté par une séquence de nœuds. Une *bin* assignée à un trajet visitera ainsi séquentiellement chaque nœud du trajet auquel il est assigné. Pour chaque trajet  $p \in \mathcal{P}$ , le premier nœud de la séquence correspond au nœud de départ et le dernier nœud de la séquence correspond au nœud de destination. La fonction  $P_0(p)$  donne le nœud de départ du trajet et la fonction  $P_{fin}(p)$  donne le nœud de destination.

Si le trajet à une cardinalité de 2, sa séquence de nœud comprend uniquement un nœud de départ et un nœud de destination. Il n'y a donc aucun nœud intermédiaire. Les *items* peuvent être chargés uniquement au nœud de départ et être déchargés au nœud de destination. La contrainte (2) permet de vérifier que pour chaque *bin* assigné à un trajet de cardinalité 2, à une période donné, le volume total des items chargé au nœud de départ n'excède pas sa capacité.

Si le trajet à une cardinalité supérieur à 2, sa séquence de nœud comprend un nœud de départ, un nœud de destination et au moins un nœud intermédiaire. Le nombre de nœuds intermédiaires est égal à la cardinalité du trajet moins le nœud de départ et le nœud de destination. Pour un trajet  $p \in \mathcal{P}$  où  $card(p)$  représente la cardinalité de  $p$ , il y a  $card(p) - 2$  nœuds intermédiaires. Comme pour les trajets de cardinalité 2, les *items* peuvent être chargés au nœud de départ et être déchargés au nœud de destination. Cependant, il est également possible de charger et décharger un *item* à chaque nœud intermédiaire du trajet. La contrainte (3) permet de vérifier que pour chaque *bin* assigné à un trajet de cardinalité supérieur

à 2, à une période donnée, le volume total des items chargé au nœud de départ, additionné au volume total des items chargés aux nœuds intermédiaires, moins le volume total des items déchargés aux nœuds intermédiaires, n'excède pas sa capacité pour chaque nœud du trajet. Pour une bin associée à un trajet  $p \in \mathcal{P}$  à une période donnée, nous avons défini que pour chaque nœud intermédiaire  $n \in \mathcal{P}$ , il existe trois différents types de déplacement d'items dont nous devons tenir compte pour nous assurer que le chargement total des *bins* n'excède pas leur capacité : (d1) les items qui proviennent d'un nœud précédant  $n$  et dont la destination est un nœud subséquent à  $n$ , (d2) les items qui proviennent du nœud  $n$  et dont la destination est un nœud subséquent à  $n$ , et (d3) les items qui proviennent d'un nœud précédant  $n$  et dont la destination est le nœud  $n$ . Les deux premiers mouvements (d1) et (d2) entraînent un chargement d'item et le troisième (d3) entraîne un déchargement. Pour chaque nœud intermédiaire  $n$ , nous voulons nous assurer que le volume total d'item chargé, moins le volume total d'item déchargé n'excède pas la capacité de la *bin*. La contrainte (3) assure donc que pour chaque nœud intermédiaire  $n$ , le volume total des items qui proviennent d'un nœud précédant  $n$  et dont la destination est un nœud subséquent à  $n$ , additionné au volume total des items qui proviennent du nœud  $n$  et dont la destination est un nœud subséquent à  $n$ , moins le volume total des items qui proviennent d'un nœud précédant  $n$  et dont la destination est le nœud  $n$  n'excède pas la capacité de la *bin*.

La contrainte (4) permet de s'assurer que chaque *item* est soit assigné à une *bin* à une période donnée, soit est comptabilisé comme non-livré. L'option de la non-livraison permet au modèle de produire une solution réalisable, même si certains *items* sont impossibles à livrer dans les délais imposés, dans le cas où il n'existe aucune association *item/bin/période* réalisable.

La contrainte (5) permet de s'assurer que chaque *bin* est assigné à un maximum d'un trajet et à une seule période.

#### 4.3 Collecte de donnée et production d'instances

À la suite de l'analyse du plan directeur du parc Jean-Drapeau, nous avons pu identifier notre problématique et définir un réseau de transport en microcorridor permettant d'acheminer les marchandises entre les futurs hubs logistiques du parc. Après avoir identifié une première version du réseau de transport en microcorridor, nous avons sollicité une rencontre avec une conseillère en mobilité de la SPJD afin de valider notre hypothèse de départ et de mieux comprendre les enjeux concrets du transport sur le site.

Cette visite a également été l'occasion de discuter, sous la forme d'une entrevue non dirigée (voir tableau 4-4), des enjeux liés au transport sur le site et de valider notre hypothèse initiale portant sur le développement d'un modèle d'optimisation adapté aux microcorridors de transport.

Tableau 4-4 Participante de la première entrevue non dirigée

| Poste du participant | Direction du participant                                   | Objectif                                                                                                                                              |
|----------------------|------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| Conseillère mobilité | SPJD, Direction planification et partenariats stratégiques | <p>Visiter le parc afin de mieux comprendre la topologie.</p> <p>Valider notre hypothèse de départ et la présence de microcorridors de transport.</p> |

Les observations issues de cette première entrevue ont confirmé la pertinence de notre approche et mis en évidence le besoin de tester notre outil dans des conditions réalistes. Pour ce faire, il est indispensable de produire des instances de problème permettant : (1) de tester les limites du modèle et (2) de résoudre des problèmes concrets.

Sur le plan de la modélisation, une instance correspond à un ensemble de données associant une valeur à chacun des paramètres du modèle afin de générer une solution. Il est important de noter que les exigences particulières en matière de transport, telles que la réfrigération ou la fragilité, ne sont pas considérées comme des variables de décision, mais plutôt comme des contraintes physiques prises en compte lors du prétraitement avant la résolution du problème par le solveur. Les variables du modèle ont été définies lors des étapes de formulation et d'élaboration du modèle.

Pour générer ces instances, il est nécessaire de disposer de données pour chacun de ces paramètres afin d'estimer la nature et le volume des marchandises transitant en interne sur le site du parc Jean-Drapeau pour différents scénarios d'intérêt. L'objectif est de produire des instances représentatives de la réalité du terrain.

Initialement, nous souhaitions obtenir, de la part des différents acteurs du parc, des données existantes permettant d'estimer le volume et le type de marchandises circulant sur le site lors de périodes spécifiques. Ces données devaient couvrir différentes périodes, notamment la saison hivernale, la saison estivale hors événement et les périodes de grands événements tels que le Grand Prix, Piknic Électronik ou OSHEAGA. Afin de faciliter la structuration et la collecte des données nécessaires à la résolution du problème, nous avons conçu un classeur Excel selon une logique de base relationnelle, répertoriant les champs pertinents. Cet outil, décrit à l'annexe D, encadre le format des données essentielles à la génération des instances.

Malgré nos efforts pour obtenir des données réelles, les informations disponibles ne couvraient pas suffisamment notre problématique initiale. Nous avons donc opté pour la génération d'instances synthétiques afin d'évaluer la robustesse technique du modèle et de l'OAD. Cette étape constitue également une étape nécessaire à la validation du modèle, puisqu'elle permet de vérifier l'obtention de solutions correctes respectant l'ensemble des contraintes imposées, mais aussi d'évaluer la difficulté de résolution, notamment en termes d'effort computationnel requis lors de l'utilisation de solveurs commerciaux. Afin de produire des instances les plus représentatives possibles, nous avons échangé avec notre personne-ressource au sein de la SPJD afin d'identifier les types de marchandises les plus susceptibles d'être transportées dans le microcorridor. À partir de ces échanges, nous avons défini trois catégories d'items : petits, moyens et grands, chacune correspondante à une tranche de volume spécifique. Cette classification nous permet de générer divers jeux de données, contenant des proportions variables d'items de chaque catégorie, afin d'évaluer les performances et les limites du modèle.

Afin de valider la pertinence de l'outil et d'en clarifier le fonctionnement, nous avons élaboré trois histoires utilisateur illustrant des scénarios d'usage concrets : (1) un gestionnaire souhaitant planifier les opérations de maintenance réalisées par les agents du parc, (2) un conseiller en mobilité cherchant à optimiser la composition de la flotte de véhicules en fonction des besoins du parc et (3) un analyste en logistique visant à identifier l'emplacement optimal des hubs logistiques pour le chargement et le déchargement. Ces histoires, détaillées à l'annexe E, ont été présentées aux gestionnaires de la SPJD responsables des opérations lors d'une deuxième entrevue non dirigée (voir tableau 4-5), afin d'exposer les fonctionnalités de l'outil et de recueillir leurs avis sur ses applications potentielles.

Tableau 4-5 Participants à la deuxième entrevue non dirigée

| Poste du participant                                          | Direction du participant                                   | Objectif                                                                          |
|---------------------------------------------------------------|------------------------------------------------------------|-----------------------------------------------------------------------------------|
| Conseillère mobilité                                          | SPJD, Direction planification et partenariats stratégiques | Présenter l'OAD aux parties prenante et identifier des cas d'utilisation concrets |
| Chef de service, logistique durable et protection des milieux | SPJD, Direction exploitation                               |                                                                                   |
| Cheffe d'équipe, développement durable et mobilité            | SPJD, Direction planification et partenariats stratégiques |                                                                                   |

Les retours des gestionnaires du parc ont également mis en lumière une difficulté de compréhension fondamentale : la distinction entre un problème de transport routier, où l'objectif est de choisir la meilleure route dans un réseau complexe, et un problème de transport en microcorridor, où la décision porte sur l'optimisation des combinaisons véhicule/trajet/chargement dans un réseau linéaire, n'était pas immédiatement évidente pour les participants. Cette difficulté de compréhension rendait initialement complexe l'identification des problématiques de transport pouvant être modélisées à l'aide d'un microcorridor. L'utilisation de scénarios concrets a permis d'améliorer la compréhension du problème et de faciliter la communication avec la SPJD, rendant possible une exploration plus ciblée des cas d'application potentiels.

Les discussions avec les responsables du parc ont également permis de confirmer l'impossibilité de collecter des données directement exploitables pour notre problème initial. En conséquence, nous avons recentré notre approche sur deux cas d'utilisation concrets, pour lesquels des instances réalistes ont été générées à partir des informations transmises par les responsables du parc sur les véhicules et les marchandises transportées. Ces informations ont été collectées à la suite d'une troisième entrevue non dirigée (voir tableau 4-6).



Tableau 4-6 Participants à la troisième entrevue non dirigée

| Poste                                                         | Direction                    | Objectif                                                                                         |
|---------------------------------------------------------------|------------------------------|--------------------------------------------------------------------------------------------------|
| Chef de service, logistique durable et protection des milieux | SPJD, Direction exploitation | Préciser les données nécessaires pour produire des instances correspondant aux cas d'utilisation |
| Superviseur                                                   | SPJD, Direction exploitation |                                                                                                  |

Le premier cas concerne le transport de personnel entre les différents bureaux de la SPJD.

Le second cas d'utilisation porte sur le transport de marchandises pour l'exécution de tâches de maintenance récurrentes au sein du parc

Ces deux cas d'utilisation permettent de tester l'OAD dans des scénarios opérationnels variés tout en assurant la cohérence avec la structure du modèle et ses hypothèses. Les paramètres des instances générés ainsi que les résultats des tests sont présentés dans le chapitre V portant sur les tests, démonstration et évaluation de l'OAD

#### 4.4 Conception et développement de l'outil d'aide à la décision

Notre artéfact est un OAD conçu pour produire un plan de transport de marchandises optimal à partir d'un problème initial. Ce problème est présenté à l'OAD sous la forme d'une instance, laquelle regroupe l'ensemble des paramètres nécessaires à la résolution et permet de générer une solution adaptée. La figure 4-6 illustre la structure de notre OAD.

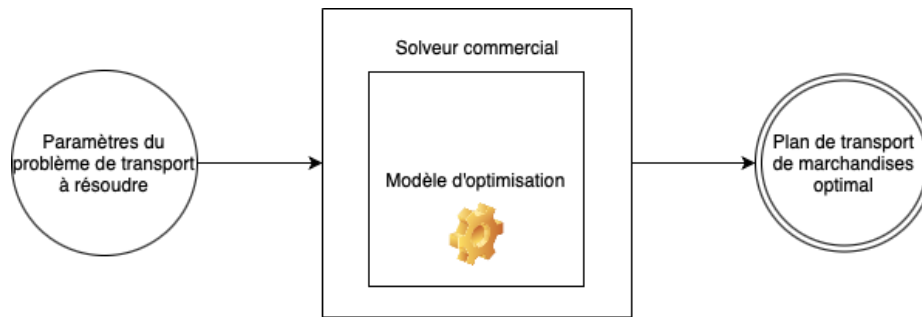


Figure 4-6 Structure de l'OAD

Une instance est composée de deux volets : (1) les marchandises à livrer (*Items*), décrites dans le tableau 4-7, qui inclut des informations telles que les volumes, les positions initiales, les destinations et les périodes de disponibilité, et (2) les véhicules disponibles pour le transport (*Bins*), détaillés dans le tableau 4-8, comprenant leurs capacités, coûts d'utilisation, positions initiales, et périodes de disponibilité.

Tableau 4-7 Description des champs des Items dans une instance de transport

| Nom du champ   | Type de donnée | Description                                     |
|----------------|----------------|-------------------------------------------------|
| item           | Nombre entier  | Numéro de l'item                                |
| volume         | Nombre décimal | Volume de l'item                                |
| dispoMin       | Nombre entier  | Période de disponibilité minimale               |
| dispoMax       | Nombre entier  | Période de disponibilité maximale               |
| livraisonDu    | Nombre entier  | Période à laquelle la livraison est due         |
| positionItem   | Nombre entier  | Numéro du nœud où se trouve présentement l'item |
| positionFinale | Nombre entier  | Numéro du nœud où l'item doit être livré        |

Tableau 4-8 Description des champs des Bins dans une instance de transport

| Nom du champ    | Type de donnée | Description                                 |
|-----------------|----------------|---------------------------------------------|
| bin             | Nombre entier  | Numéro de la bin                            |
| volumeMax       | Nombre décimal | Capacité maximale (volume) de la bin        |
| coutUtilisation | Nombre décimal | Coût d'utilisation fixe                     |
| dispoMin        | Nombre entier  | Période de disponibilité minimale de la bin |
| dispoMax        | Nombre entier  | Période de disponibilité maximale de la bin |
| positionBin     | Nombre entier  | Position initiale de la bin                 |

Ces informations sont compilées dans un chiffrier électronique, tel qu'Excel, avant d'être exportées au format .csv pour être utilisées comme paramètres lors de l'exécution du programme. La figure 4-7 présente un exemple de fichier regroupant les informations relatives aux *Items* et la figure 4-8 présente un exemple de fichier regroupant les informations relatives aux *Bins* d'une instance de démonstration.

| item | volume | dispoMin | dispoMax | livraisonDu | positionItem | positionFinale |
|------|--------|----------|----------|-------------|--------------|----------------|
| 1    | 30     | 10       | 24       | 18          | 1            | 3              |
| 2    | 2      | 11       | 24       | 18          | 1            | 2              |
| 3    | 14     | 10       | 24       | 18          | 2            | 3              |
| 4    | 28     | 11       | 24       | 18          | 3            | 1              |

Figure 4-7 Exemple de tableau Excel appelé item4\_demo.csv regroupant le volet item d'une instance de démonstration

| bin | volumeMax | coutUtilisation | dispoMin | dispoMax | positionBin |
|-----|-----------|-----------------|----------|----------|-------------|
| 1   | 441       | 36              | 9        | 24       | 1           |
| 2   | 422       | 34              | 15       | 23       | 2           |
| 3   | 406       | 24              | 6        | 24       | 3           |

Figure 4-8 Exemple de fichier appelé bin3\_demo.csv regroupant le volet bin d’une l’instance de démonstration

Les utilisateurs interagissent avec le système au moyen d’un programme écrit en Python 3.8, tirant parti de la bibliothèque *DOCPLEX* pour la modélisation et la résolution du problème. L’utilisation de la ligne de commande depuis un terminal offre une interface simple et directe, permettant à l’utilisateur de passer les paramètres nécessaires (fichiers d’instances, options de configuration, etc.) et de visualiser les résultats de manière immédiate.

Afin d’assurer le bon fonctionnement de l’OAD, il est nécessaire de disposer d’un environnement Python 3.8 dans lequel les bibliothèques suivantes sont installées: sys, numpy, csv, math, configparser, json, argparse, docplex.

De plus, il est impératif que CPLEX 22.1.1 soit installé sur l’ordinateur exécutant le programme. Le programme représente le problème de transport selon le modèle d’optimisation élaboré à la section 4.2, de façon que l’instance du problème puisse être résolue avec le solveur commercial CPLEX 22.1.1 d’IBM, et que l’OAD produise une solution optimale.

Pour standardiser l’utilisation de l’OAD et faciliter l’organisation des fichiers, nous avons structuré l’arborescence des répertoires comme illustrée à la figure 4-9. Le dossier principal, situé à la racine de l’arborescence, est nommé *OAD\_BinPacking\_prog*. Ce dossier, contient les quatre fichiers suivants : (1) *scriptOAD.py*, le fichier exécutable contenant le code source du programme python, (2) *param.cfg*, le fichier de configuration, dans lequel sont définis les paramètres du problème de transport nécessaire à la résolution de l’instance, (3) *default\_cost\_functions.py*, le fichier contenant les fonctions de coût par défaut utilisées par le programme et (4) *cost\_functions.py*, le fichier permettant de définir les fonctions de coût personnalisées. En plus de ces fichiers, le dossier principal contient également un sous-dossier dédié aux données des instances à résoudre. Ce dernier est divisé en deux répertoires distincts : (1) *Bins*, qui contient

les fichiers listant les *bins* des instances et leurs caractéristiques et (2) *Items* : Contient les fichiers listant les *items* des instances et leurs caractéristiques.

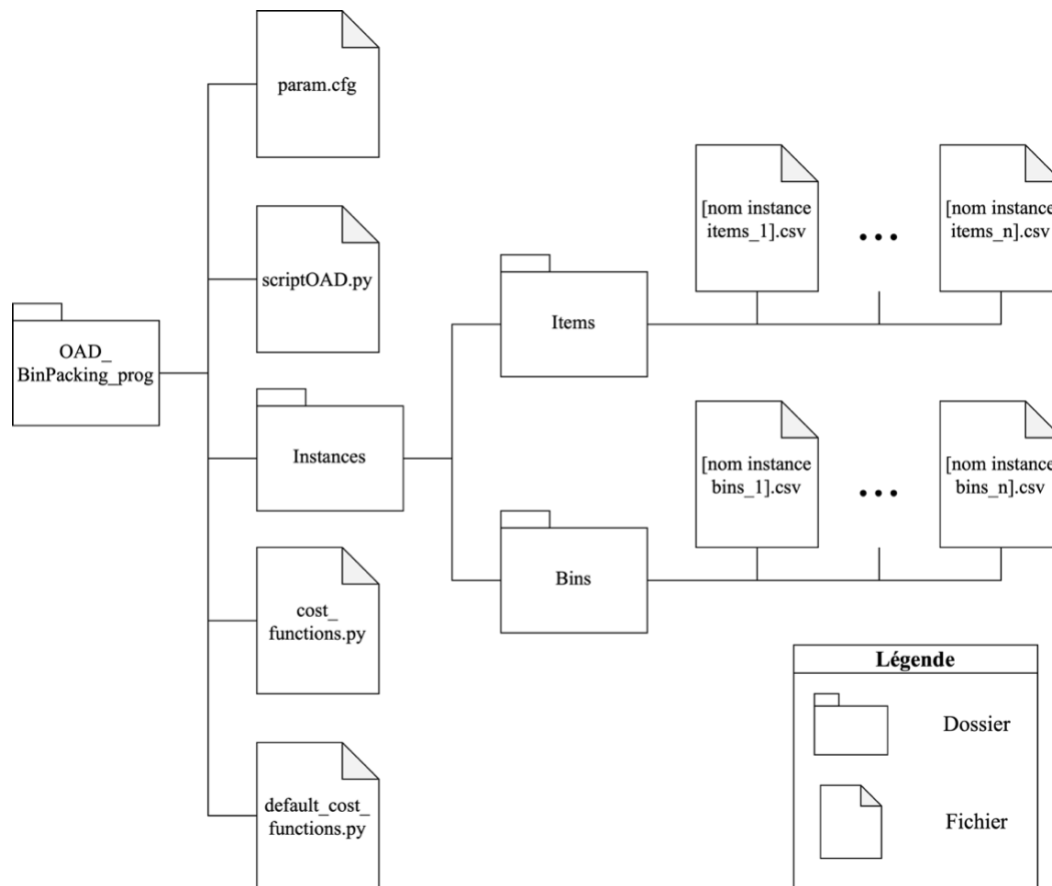


Figure 4-9 Arborescence des fichiers et dossiers de l'OAD

Pour exécuter le programme, il est nécessaire d'ouvrir le terminal et d'accéder au répertoire où se trouve le dossier principal de l'OAD, intitulé `OAD_BinPacking_prog`. Ce dossier contient les fichiers nécessaires à l'exécution du programme ainsi que les fichiers d'instances à résoudre. Dans notre exemple, le système d'exploitation utilisé est macOS, et le dossier principal est situé dans le répertoire *Desktop*. Pour accéder à ce répertoire, la commande suivante est utilisée à partir du terminal :

```
cd ~/Desktop/OAD_BinPacking_prog
```

Une fois dans le répertoire à la racine de l'arborescence, le programme peut être exécuté avec la commande suivante :

```
python3.8 scriptOAD.py [options] itemListData binListData
```

Les arguments attendus par cette commande sont décrits dans le tableau 4-9:

Tableau 4-9 Arguments attendus par le programme scriptOAD.py

| Arguments obligatoires | Description                                                                                                                       |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| itemListData           | Nom du fichier contenant les données des <i>items</i> . Ce fichier doit obligatoirement se trouver dans le dossier <i>Items</i> . |
| binListData            | Nom du fichier contenant les données des <i>bins</i> . Ce fichier doit obligatoirement se trouver dans le dossier <i>Bins</i> .   |
| Arguments optionnels   | Description                                                                                                                       |
| [-h/--help]            | Permet d'afficher un menu d'aide.                                                                                                 |
| [--cc]                 | Active les fonctions de coût personnalisées.                                                                                      |

L'argument optionnel [-h/--help] permet d'afficher un menu d'aide listant et décrivant les arguments du programme ainsi que leur position dans la ligne de commande. Ce menu, tel qu'illustré à la figure 4.4, est accessible via l'une des commandes suivantes :

```
python3.8 scriptOAD.py -h
```

```
python3.8 scriptOAD.py --help
```

```
python3.8 scriptOAD.py --help
```

```
usage: scriptOAD.py [-h] [--cc] itemListData binListData

positional arguments:
  itemListData  fichier contenant les données des items au format .csv
  binListData   fichier contenant les données des bins au format .csv

optional arguments:
  -h, --help      show this help message and exit
  --cc            utiliser les fonctions de coût personnalisées
```

Figure 4-10 Commande d'aide et arguments du script scriptOAD.py

Afin de permettre aux utilisateurs d'adapter facilement les configurations du programme à différents scénarios de transport sans avoir à modifier directement le code source, un fichier de configuration nommé *param.cfg* est fourni. Ce fichier contient l'ensemble des paramètres nécessaires à l'exécution du programme, que les utilisateurs peuvent modifier en fonction des besoins spécifiques.

Les différents paramètres configurables, ainsi que leur type de donnée et leur rôle, sont décrits dans le tableau 4-10 ci-dessous. Ces paramètres couvrent divers aspects d'un problème de transport, tels que les trajets possibles, les périodes de livraison autorisées, et les coûts associés aux différentes opérations.

Tableau 4-10 Description des paramètres du programme modifiables contenue dans le fichier param.cfg

| Nom du paramètre   | Type de donnée | Description                                                                                             |
|--------------------|----------------|---------------------------------------------------------------------------------------------------------|
| <i>trajets</i>     | Liste de liste | Tableau listant les trajets possibles. Un trajet est représenté par la séquence des nœuds le composant. |
| <i>periode_min</i> | Nombre entier  | Nombre entier Période de livraison minimale                                                             |
| <i>periode_max</i> | Nombre entier  | Période de livraison maximale                                                                           |

|                                           |                          |                                                                   |
|-------------------------------------------|--------------------------|-------------------------------------------------------------------|
| <i>Bin_trajet_assignment_cost</i>         | Nombre entier ou décimal | Coût d'assignation d'une <i>Bin</i> à un <i>Trajet</i> par défaut |
| <i>Item_bin_assignment_cost</i>           | Nombre entier ou décimal | Coût d'assignation d'un <i>Item</i> à une <i>Bin</i> par défaut   |
| <i>Item_non_livraison_assignment_cost</i> | Nombre entier ou décimal | Coût de non-livraison d'un <i>Item</i> par défaut                 |

La figure 4-11 illustre un exemple concret de configuration dans le fichier *param.cfg*. Ce dernier montre des valeurs types qui pourraient être utilisées pour définir un scénario particulier de transport.

```
#File param.cfg:
[Settings]
#trajets possibles

trajets=
[[1,2],[2,1],[1,3],[3,1],[2,3],[3,2],[1,2,3],[1,3,2],[2,3,1],[2,1,3],[3,2,1],[3,1,2],[1,3,2,1],[1,2,3,1],
[2,3,1,2],[2,1,3,1],[3,2,1,3],[3,1,2,3]]

#intervale de livraison

periode_min = 5
periode_max = 24

#cout d'utilisation par default

cout_default = True
bin_trajet_assignment_cost = 20
item_bin_assignment_cost = 2
item_non_livraison_assignment_cost = 300
```

Figure 4-11 Exemple de valeurs pour le fichier param.cfg

Tel que définie lors de l'élaboration du modèle, la fonction objectif de ce modèle vise à minimiser les coûts totaux associés aux trois décisions possibles : (1) le coût d'assignation d'une bin à un trajet pour une période donnée, (2) le coût d'assignation d'un item à une bin associée à un certain trajet pour une période donnée et (3) le coût de non-livraison d'un item à une période donnée.

Le coût d'assignation d'une bin à un trajet pour une période donnée comprend à la fois le coût fixe d'utilisation propre à chaque bin et le coût associé à son affectation sur un trajet pour une période précise.



Le coût fixe d'utilisation est systématiquement défini par la valeur du champ *coût d'utilisation* de chaque bin dans le fichier d'instances des *Bins* du problème de transport. Par défaut, le coût d'assignation d'une bin à un trajet pour une période donnée est identique pour toutes les affectations et correspond à la valeur du champ *Bin\_trajet\_assignment\_cost* du fichier de configuration.

Le coût d'assignation d'un item à une bin associée à un certain trajet pour une période donnée et le coût de non-livraison d'un item sont également fixes par défaut et se rapportent aux champs *Item\_bin\_assignment\_cost* et *Item\_non\_livraison-assignment\_cost* du fichier de configuration.

Comme indiqué lors de la description de l'arborescence de l'OAD, il existe deux fichiers de fonctions de coût : *default\_cost\_functions.py* et *cost\_functions.py*. Le fichier *default\_cost\_functions.py* contient les fonctions de coût par défaut décrites précédemment, qui s'appuient uniquement sur les valeurs spécifiées dans *Bin\_trajet\_assignment\_cost*, *Item\_bin\_assignment\_cost* et *Item\_non\_livraison-assignment\_cost* du fichier de configuration.

La figure 4-12 illustre le code permettant, par défaut, de calculer le coût d'assignation d'un item à une bin. Cette fonction, appelée *item\_bin\_assignment\_cost(n,model)* retourne simplement la valeur de la variable *Item\_bin\_assignment\_cost*, qui est initialisée d'après la valeur définie pour *Bin\_trajet\_assignment\_cost* dans le fichier de configuration. Les deux autres fonctions de coût par défaut sont appelées *bin\_trajet\_assignment\_cost(n,model)* et *item\_non\_livraison\_assignment\_cost(n,model)* et suivent la même logique.

```
def item_bin_assignment_cost(n,model): #ajouter coût selon interval de livraison
    return model.item_bin_assignment_cost_def
```

Figure 4-12 Fonction utilisée par défaut pour calculer le coût d'assignation d'un item à une bin

Le fichier `cost_functions.py` permet à l'utilisateur de définir des fonctions de coût personnalisées lorsque les fonctions par défaut ne répondent pas à ses besoins. Pour garantir le bon fonctionnement du programme, il est nécessaire de modifier le corps des trois fonctions de coût (`item_bin_assignment_cost(n, model)`, `bin_trajet_assignment_cost(n, model)` et `item_non_livraison_assignment_cost(n, model)`), sans toutefois changer leur nom d'origine.

```
# légende des composantes du tuple n
#     i = num item
#     j = num bin
#     p = num trajet
#     t = num periode

# composantes de model
# model.bin_volume_max : tableau regroupant la capacite de chaque bin.           Ex: model.bin_volume_max[2]
# model.item_volume     : tableau regroupant le volume de chaque item.           Ex: model.item_volume[4]
# model.item_destination : tableau regroupant la destination de chaque item       Ex: model.item_destination[1]
# model.item_position    : tableau regroupant la position initiale de chaque item  Ex: model.item_position[1]
# model.itemLivraisonDu  : tableau regroupant la période de livraison idéale de chaque item Ex: model.itemLivraisonDu[1]
# #1
# model.trajets_p        : tableau regroupant les différents trajets possibles     Ex: model.trajets_p[1]
# trajet #1
# model.item_bin_assignment_cost_def : coût fixe par défaut de l'assignation d'un item à une bin
# model.bin_trajet_assignment_cost_def : coût fixe par défaut de l'assignation d'une bin à un trajet
# model.item_non_livraison_assignment_cost_def : coût fixe par défaut de la non livraison d'un item
```

Figure 4-13 Aide-mémoire pour la création de fonctions de coût personnalisé dans le fichier `cost_functions.py`.

La variable `model` passée en paramètre offre un accès aux différentes composantes du modèle, notamment les caractéristiques des bins, des items et les paramètres généraux du problème. Le tableau 4-11 recense ces composantes et sert de référence à l'utilisateur en figurant en en-tête du fichier `cost_functions.py` (figure 4-13). Cette interface a été conçue pour permettre la création de fonctions de coût personnalisées sans qu'il soit nécessaire de modifier le code source principal de l'outil.

Tableau 4-11 Composantes du modèle utilisable pour les fonctions de coût personnalisées

| Composante                                   | Signification                                                                                                                       | Exemple                                                               |
|----------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------|
| model.bin_volume_max                         | Tableau regroupant la capacité de chaque bin                                                                                        | model.bin_volume_max[1] : capacité de la bin #2                       |
| model.item_volume                            | Tableau regroupant le volume de chaque item                                                                                         | model.item_volume[0] : volume de l'item #4                            |
| model.item_destination                       | Tableau regroupant la destination de chaque item                                                                                    | model.item_destination[0] : nœud de destination de l'item #1          |
| model.item_position                          | Tableau regroupant la position initiale de chaque item                                                                              | model.item_position[0] : position initiale de l'item #1               |
| model.itemLivraisonDu                        | Tableau regroupant la période de livraison idéale de chaque item                                                                    | model.itemLivraisonDu[0] : période idéale de livraison pour l'item #1 |
| model.trajets_p                              | Tableau décrivant les différents trajets possibles                                                                                  | model.trajets_p[1] : séquence de nœuds composant le trajet #1         |
| model.item_bin_assignment_cost_def           | Coût fixe par défaut de l'assignation d'un item à une bin (valeur de Item_bin_assignment_cost dans le fichier de configuration)     |                                                                       |
| model.bin_trajet_assignment_cost_def         | Coût fixe par défaut de l'assignation d'une bin à un trajet (valeur de Bin_trajet_assignment_cost dans le fichier de configuration) |                                                                       |
| model.item_non_livraison_assignment_cost_def | Coût fixe par défaut de la non-livraison d'un item (valeur de Item_non_livraison-assignment_cost dans le fichier de configuration)  |                                                                       |

La figure 4-14 propose un exemple de code personnalisé servant à calculer le coût d'assignation d'un item à une bin en tenant compte du volume de l'item, de la période de livraison et de l'écart entre la période prévue et la période réelle de livraison.

```
#composantes de n = [i,j,p,t] ==> n[0] = i ; n[1] = j; n[2] = p; n[3] = t
def item_bin_assignment_cost(n,model): #ajouter coût selon interval de livraison
    coutRetard = 0
    if model.itemLivraisonDu[n[0]-1] - n[3] < 0: coutRetard = abs(model.itemLivraisonDu[n[0]-1] - n[3]) * 20

    if n[3] <= 18:
        return 1 + 0.5 * model.item_volume[n[0]-1] + coutRetard
    else:
        return 2 + 0.5 * model.item_volume[n[0]-1] + coutRetard
```

Figure 4-14 Exemple de fonction personnalisée pouvant être utilisé pour calculer le coût d'assignation d'un item à une bin

La correspondance entre les différents types de coûts et leur implémentation (par défaut ou personnalisée) est résumée dans le tableau 4-12 ci-dessous.

Tableau 4-12 Principaux types de coût, Valeurs par défaut et valeurs personnalisées

| Types de coût                | Valeur par défaut (default_cost_functions.py)                                                               | Valeur personnalisée (cost_functions.py)                                                                                                          |
|------------------------------|-------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| Coût fixe d'utilisation      | Toujours égal à la valeur du champ <i>coût d'utilisation</i> de la bin dans le fichier d'instances des Bins | Toujours égal à la valeur du champ <i>coût d'utilisation</i> de la bin dans le fichier d'instances des Bins                                       |
| Coût d'assignation d'une bin | Toujours égal à la valeur du champ <i>Bin_trajet_assignment_cost</i> dans le fichier de configuration       | Calculé selon le code défini par l'utilisateur pour la fonction <i>item_bin_assignment_cost(n,model)</i> dans le fichier <i>cost_functions.py</i> |

|                                 |                                                                                                               |                                                                                                                                                             |
|---------------------------------|---------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Coût d'assignation d'un item    | Toujours égal à la valeur du champ <i>Item_bin_assignment_cost</i> dans le fichier de configuration           | Calculé selon le code défini par l'utilisateur pour la fonction <i>bin_trajet_assignment_cost(n,model)</i> dans le fichier <i>cost_functions.py</i>         |
| Coût de non-livraison d'un item | Toujours égal à la valeur du champ <i>Item_non_livraison-assignment_cost</i> dans le fichier de configuration | Calculé selon le code défini par l'utilisateur pour la fonction <i>item_non_livraison_assignment_cost(n,model)</i> dans le fichier <i>cost_functions.py</i> |

Lorsque l'option `--cc` est spécifiée sur la ligne de commande, le coût d'assignation d'une bin à un trajet pour une période donnée, le coût d'assignation d'un item à une bin pour une période donnée et le coût de non-livraison d'un item sont alors calculés en fonction des définitions contenues dans *cost\_functions.py* plutôt que dans *default\_cost\_functions.py*.

Pour illustrer l'utilisation de l'OAD, considérons une instance de démonstration reposant sur les fichiers *item4\_demo.csv* (données de quatre objets à livrer) et *bin3\_demo.csv* (données des véhicules disponibles pour la livraison). Le contenu de ces fichiers est présenté précédemment aux figures 4-7 et 4-8. Nous résolvons cette instance en appliquant la configuration décrite à la figure 4-11. Les trajets possibles englobent toutes les combinaisons de trajets au sein d'un graphe bidirectionnel cyclique à trois nœuds, une situation semblable au réseau de transport reliant les hubs du parc Jean-Drapeau. L'intervalle de livraison est défini avec une *periode\_min* de 5 et une *periode\_max* de 24, indiquant que les marchandises peuvent être livrées toutes les heures entre 5 h et 24 h. De plus, nous avons fixé le coût d'assignation d'un véhicule à un trajet à 20, le coût d'assignation d'un objet à un véhicule à 2 et le coût de non-livraison d'un objet à 300.

Pour produire une solution optimale à ce problème de transport en utilisant les fonctions de coût par défaut, la commande suivante est exécutée dans le terminal :

```
Python3.8 scriptOAD.py item4_demo.csv bin3_demo.csv
```

La figure 4-15 illustre la solution générée par l’OAD. Elle rassemble les trajets optimaux des bins, l’affectation des items et les résultats obtenus avec les fonctions de coût par défaut.

```
python3.8 scriptOAD.py item4_demo.csv bin3_demo.csv
```

```
Total (root+branch&cut) = 0.01 sec. (1.46 ticks)
solution for a cost of 162.0
* KPI: Bin/trajet assignment cost = 154.000
* KPI: Item/bin assignment cost = 8.000
* KPI: Item non livraison cost = 0.000

***** SOLUTION *****

***** Bin a trajet assignment y_j_p_t *****

+-----+-----+-----+-----+
| Variable: y_1_1_13 | Bin: 1 | Trajet 1 : 1 => 2 | Période: 13 |
+-----+-----+-----+-----+
| Variable: y_2_9_17 | Bin: 2 | Trajet 9 : 2 => 3 => 1 | Période: 17 |
+-----+-----+-----+-----+
| Variable: y_3_17_12 | Bin: 3 | Trajet 17: 3 => 2 => 1 => 3 | Période: 12 |
+-----+-----+-----+-----+

***** Item a bin assignment x_i_j_p_t *****

+-----+-----+-----+-----+
| Variable: x_1_3_17_12 | Item: 1 | Bin: 3 | Trajet 17: 3 - 2 - 1 - 3 | Période: 12 |
+-----+-----+-----+-----+
| Variable: x_2_1_1_13 | Item: 2 | Bin: 1 | Trajet 1 : 1 - 2 | Période: 13 |
+-----+-----+-----+-----+
| Variable: x_3_3_17_12 | Item: 3 | Bin: 3 | Trajet 17: 3 - 2 - 1 - 3 | Période: 12 |
+-----+-----+-----+-----+
| Variable: x_4_2_9_17 | Item: 4 | Bin: 2 | Trajet 9 : 2 - 3 - 1 | Période: 17 |
+-----+-----+-----+-----+

***** Non livraison Item u_i_t *****

+-----+
| AUCUNE NON LIVRAISON |
+-----+
```

Figure 4-15 Commande employée pour exécuter le programme avec deux fichiers de test, en appliquant les fonctions de coût par défaut, et aperçu des résultats

La solution est toujours présentée de la façon suivante :

La première section est intitulée *Bin a trajet assignment y\_j\_p\_t*. Elle précise, pour chaque période, quelle bin est assignée à quel trajet. Dans cet exemple, le véhicule 1 (*Bin 1*) est assigné au trajet 1 à la période 13, parcourant les nœuds 1 et 2 entre 13 h 00 et 14 h 00. Le véhicule 2 (*Bin 2*) est assigné au trajet 9 à la

période 17, parcourant les nœuds 2, 3 et 1 entre 17 h 00 et 18 h 00. Enfin, le véhicule 3 (*Bin 3*) est assigné au trajet 17 à la période 12, couvrant les nœuds 3, 2, 1 et 3 entre 12 h 00 et 13 h 00.

La deuxième section est intitulée *Item a Bin assignment  $x_{i_j_p_t}$* . Elle indique à quelle bin et à quelle période chaque item est assigné. Dans notre exemple, l'objet 1 (*Item 1*) et l'objet 3 sont assignés au bin 3 qui parcourt le trajet défini ci-dessus. L'objet 2 (*Item 2*) est assigné au bin 1, et l'objet 4 (*Item 4*) est assigné au bin 2, chacun effectuant le trajet correspondant à sa période d'affectation.

La troisième section, intitulée *Non-livraison Item  $u_{i_t}$* , répertorie les items qui ne seront pas livrés. En attribuant un coût de non-livraison arbitrairement élevé, on s'assure que cette option n'est rentable que si la livraison d'un ou de plusieurs items s'avère impossible compte tenu des paramètres du problème. Dans l'exemple présenté, tous les objets ont été livrés.

Pour produire une solution optimale à ce problème de transport en utilisant des fonctions de coût personnalisé, la commande suivante est exécutée dans le terminal :

```
Python3.8 scriptOAD.py item4_demo.csv bin3_demo.csv -cc
```

Dans cet exemple, les fonctions de coût *bin\_trajet\_assignment\_cost( $n$ ,  $model$ )* et *item\_non\_livraison\_assignment\_cost( $n$ ,  $model$ )* du fichier *cost\_functions.py* sont identiques aux fonctions de coût par défaut. Seule la fonction *item\_bin\_assignment\_cost( $n$ ,  $model$ )* diffère et correspond à celle présentée plus haut à la figure 4-14. La figure 4.11 montre la solution produite par l'OAD avec ces fonctions de coût personnalisées, incluant les trajets optimaux des bins, l'affectation des items et les résultats obtenus.

```
script0AD.py item4_demo.csv bin3_demo.csv --cc
```

```
Total (root+branch&cut) = 0.01 sec. (1.48 ticks)
solution for a cost of 195.0
* KPI: Bin/trajet assignment cost = 154.000
* KPI: Item/bin assignment cost = 41.000
* KPI: Item non livraison cost = 0.000

***** SOLUTION *****
***** Bin a trajet assignment y_j_p_t *****

+-----+-----+-----+-----+
| Variable: y_1_1_13 | Bin: 1 | Trajet 1 : 1 => 2 | Période: 13 |
+-----+-----+-----+-----+
| Variable: y_2_9_16 | Bin: 2 | Trajet 9 : 2 => 3 => 1 | Période: 16 |
+-----+-----+-----+-----+
| Variable: y_3_17_12 | Bin: 3 | Trajet 17: 3 => 2 => 1 => 3 | Période: 12 |
+-----+-----+-----+-----+

***** Item a bin assignment x_i_j_p_t *****

+-----+-----+-----+-----+
| Variable: x_1_3_17_12 | Item: 1 | Bin: 3 | Trajet 17: 3 - 2 - 1 - 3 | Période: 12 |
+-----+-----+-----+-----+
| Variable: x_2_1_1_13 | Item: 2 | Bin: 1 | Trajet 1 : 1 - 2 | Période: 13 |
+-----+-----+-----+-----+
| Variable: x_3_3_17_12 | Item: 3 | Bin: 3 | Trajet 17: 3 - 2 - 1 - 3 | Période: 12 |
+-----+-----+-----+-----+
| Variable: x_4_2_9_16 | Item: 4 | Bin: 2 | Trajet 9 : 2 - 3 - 1 | Période: 16 |
+-----+-----+-----+-----+

***** Non livraison Item u_i_t *****

+-----+-----+-----+-----+
| AUCUNE NON LIVRAISON |
+-----+-----+-----+-----+
```

Figure 4-16 Commande utilisée pour exécuter le programme avec deux fichiers de test, en appliquant les fonctions de coût personnalisées, ainsi que les résultats obtenus lors de l'exécution



## **CHAPITRE 5**

### **TEST, DÉMONSTRATION ET ÉVALUATION**

Dans ce chapitre, nous présentons les résultats des tests réalisés afin d'évaluer les performances du MO et de l'OAD développés dans le cadre de cette recherche. Ces tests correspondent à la quatrième phase du cadre méthodologique adopté, fondé sur une approche en science du design (O'Keefe, 2014) et sur le cadre méthodologique de la RO (Norbert, Ouellet et Parent, 2009).

Les premières sections de ce chapitre (5.1) exposent les résultats de l'évaluation technique du modèle d'optimisation dans un environnement contrôlé, à l'aide d'instances artificielles générées aléatoirement selon des paramètres réalistes. Ces tests permettent de mesurer les performances du MO en termes de qualité des solutions (gap d'optimalité) et de temps de calcul (CPU) à travers différents types d'instances représentatives de situations logistiques variées.

Les sections suivantes (5.2 et 5.3) présentent les résultats des évaluations en contexte d'utilisation réel, menées à partir de deux cas d'usage concrets identifiés avec la Société du parc Jean-Drapeau. Ces évaluations s'appuient sur la méthode Human Risk & Effectiveness Evaluation (Venable et al., 2016), permettant d'apprécier l'utilité, l'adoption et l'efficacité de l'OAD en situation semi-contrôlée, en intégrant les retours des utilisateurs.

Ainsi, ce chapitre permet de valider la robustesse technique du modèle et la pertinence opérationnelle de l'outil développé, tout en identifiant les leviers d'amélioration possibles pour une mise en œuvre effective dans des contextes similaires.

#### **5.1 Test et évaluation des performances du modèle d'optimisation**

Comme décrit dans le cadre méthodologique, nous avons initialement eu recours à la méthode d'évaluation des risques techniques et de l'efficacité pour tester la robustesse technique du modèle d'optimisation dans un environnement contrôlé. Notre objectif était de mesurer les performances de résolution de notre modèle, à la fois en termes d'optimalité, évaluée à l'aide du gap d'optimalité, et de vitesse de calcul, mesurée par le temps CPU. Le gap d'optimalité agit comme un certificat de qualité associé à une solution : il indique l'écart relatif, en valeur, entre la solution obtenue et la solution optimale du modèle d'optimisation. Le temps CPU correspond au temps requis par le processeur de l'ordinateur pour résoudre les instances; il permet d'évaluer si les solutions peuvent être produites dans des délais

raisonnables et appropriés au contexte opérationnel. Ces analyses ont été menées sur un ensemble d’instances artificielles, couvrant plusieurs types d’instances pertinentes.

Pour effectuer nos tests, nous avons développé une version alternative du programme de l’OAD dont le code source est nommé *scriptOAD\_instancesAlea.py* permettant d’exécuter automatiquement plusieurs résolutions d’instances générées aléatoirement à partir de paramètres prédéfinis. L’interface utilisateur a été supprimée et la sélection de la paire d’instances à résoudre ne se fait plus depuis le dossier *Instances*. Les instances à résoudre sont désormais générées aléatoirement à l’aide d’une fonction produisant une paire d’instances synthétiques de *bins* et d’*items*, en bornant les variables selon le type d’instance choisi. Pour chaque combinaison de nombre d’items et de type d’instance, 10 instances synthétiques ont été générées et résolues de façon à pouvoir compiler les données nécessaires à notre analyse. En plus du gap d’optimalité et du temps CPU, nous avons également compilé le statut de la résolution de chaque instance. Lorsqu’une solution optimale a été trouvée, Cplex retourne le statut *OPTIMAL\_SOLUTION*. Puisque nous avons établi un temps de résolution maximal, il est possible qu’une solution optimale (gap de 0) n’ait pas été trouvée dans les temps. Si une solution réalisable a été trouvée, Cplex retourne le statut *FEASIBLE\_SOLUTION*. Si aucune solution n’a été trouvée, Cplex retourne le statut *NO\_FEASIBLE\_SOLUTION*.

Le gap d’optimalité, noté *gap*, correspond à l’écart relatif entre la meilleure solution entière trouvée (*Best Integer*) et la borne inférieure (*Best Bound*) calculée par le solveur. Il est défini par la formule suivante :

$$gap = \frac{|Best\ Integer - Best\ Bound|}{|Best\ Integer|} \times 100$$

Ainsi, un *gap* nul indique que la solution entière trouvée est optimale, tandis qu’un *gap* positif reflète l’écart résiduel entre la meilleure solution connue et l’optimal théoriques.

Il convient de rappeler que CPLEX constitue une méthode de résolution exacte, reposant sur le paradigme des approches d’exploration arborescente de type *branch-and-bound*, combinées à l’ajout de plans coupants (*branch-and-cut*). Ce type de méthode construit un arbre d’exploration de l’espace décisionnel combinatoire, en identifiant des solutions entières admissibles (fournissant une borne supérieure dans un problème de minimisation) et en améliorant la borne inférieure grâce à l’ajout de plans coupants qui resserrent la relaxation linéaire aux nœuds de l’arbre.

Pour chaque instance résolue, nous avons compilé le gap, le temps CPU ainsi que le statut de la résolution dans un fichier au format .CSV afin d'analyser les performances du modèle. Les fichiers .CSV sont générés automatiquement au cours de l'exécution du programme.

L'arborescence des fichiers présentée au chapitre IV a été adaptée afin de permettre la conservation structurée des résultats intermédiaires et finaux produits lors des simulations. Le dossier initialement nommé *Instances* a été remplacé par un dossier *Résultats*, dans lequel les données issues des différentes exécutions sont organisées automatiquement.

Les résultats y sont classés en fonction du nom de l'ensemble de tests et du nombre d'instances d'items générées. Par ailleurs, les instances aléatoires générées sont elles-mêmes archivées et structurées selon trois critères : (1) le nom de l'ensemble de tests, (2) le nombre d'instances générées, et (3) le type d'instance.

L'exécution du programme repose sur la définition préalable du nombre d'items à générer. Pour chaque valeur, il produit des résultats correspondant à un ensemble de types d'instances, composé de  $i$  types distincts, tels que présentés au tableau 4.1. Par exemple, les résultats des simulations de 100 instances d'items appartenant aux 5 types définis pour l'ensemble 1 seront enregistrés dans un fichier nommé : *ensemble1\_100\_results.csv*. Ce fichier sera sauvegardé dans le dossier suivant : *Résultats > Simul\_ensemble1 > data\_100*.

Chaque instance produite sera enregistrée dans un dossier nommé *[items/bins]\_ensemble1\_100.csv*. Ce fichier sera sauvegardé selon son type dans le dossier suivant : *Résultats > Simul\_ensemble1 > data\_100 > setType[0-i]*.

Cette procédure est répétée pour chaque cardinalité d'items  $|I| = n$  définie en entrée.

La figure 5-1 illustre l'arborescence des fichiers et dossiers générés par cette version de l'OAD, ainsi que la nomenclature adoptée pour leur organisation.

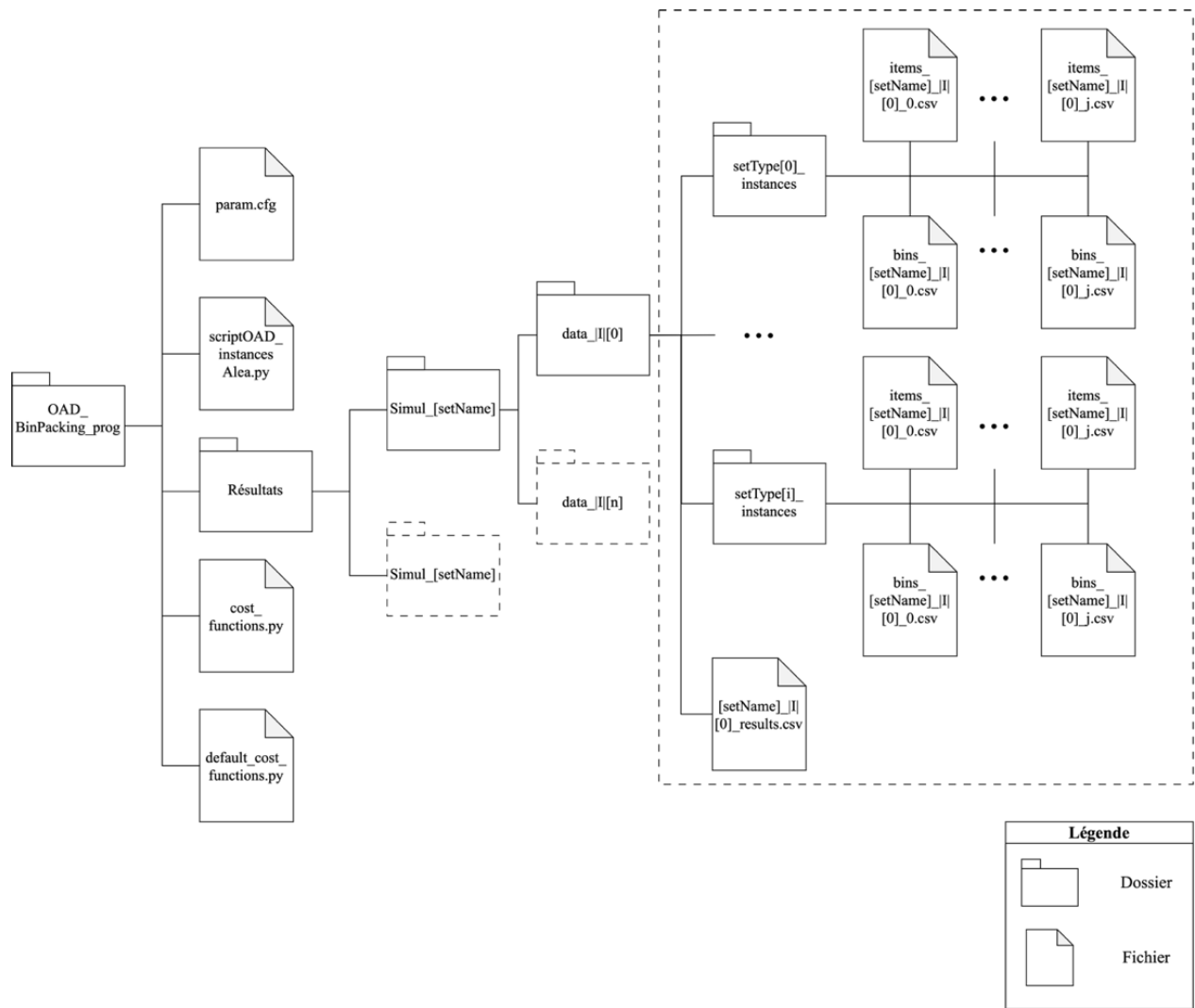


Figure 5-1 Arborescence des fichiers et dossiers pour la version aléatoire de l'OAD

Finalement, les valeurs des coûts d'utilisation par défaut utilisés par les fonctions de coût par défaut ont été paramétrées de la manière suivante :  $\text{bin\_trajet\_assignment\_cost} = 20$ ,  $\text{item\_bin\_assignment\_cost} = 2$  et  $\text{item\_non\_livraison\_assignment\_cost} = 300$ .

Afin de générer les instances de test, nous avons repris les types d'instances utilisés par Crainic et *al.* (2021) pour tester les performances de leur modèle de BPP. Ces instances sont réparties en deux ensembles. Le premier ensemble est constitué d'instances où les items sont de petite taille par rapport à la capacité des

Bins. Il est décliné en cinq types. Le deuxième ensemble est constitué d'instances où les items sont de moyenne à grande taille par rapport à la capacité des bins. Il est décliné en six types.

Pour chaque type d'instance, le nombre d'items et le nombre de périodes sont fixes. La quantité de bins et leurs capacités ainsi que le volume des items sont générés aléatoirement parmi une tranche de valeurs prédéfinie propre à chaque type d'instance. Les caractéristiques spécifiques des instances générées pour l'expérimentation computationnelle sont fournies dans le tableau 5.1. Les nœuds de départ et d'arrivée des items sont choisis aléatoirement parmi les nœuds du réseau en s'assurant qu'ils soient distincts. La fenêtre de disponibilité des items et des Bins est générée aléatoirement selon le processus suivant : (1) une durée de disponibilité est générée aléatoirement dans une plage de valeur prédéfinie; (2) La disponibilité minimale est générée aléatoirement entre 0 et le nombre de périodes de l'instance moins la durée de disponibilité générée à l'étape précédente; (3) La disponibilité maximale correspond à la somme de la disponibilité minimale et de la durée de disponibilité; (4) La livraison due de l'item est générée aléatoirement entre la disponibilité minimale et la disponibilité maximale. Pour l'ensemble des instances générées, la durée de disponibilité se situe dans l'intervalle suivant : [3-6].

En ce qui concerne le calcul des coûts d'assignation bin/trajet, item/bin et item/non-livraison, nous avons utilisé les fonctions de coût d'utilisation par défaut, telles que définies au chapitre IV, paramétrées de la manière suivante :  $\text{bin\_trajet\_assignment\_cost} = 20$ ,  $\text{item\_bin\_assignment\_cost} = 2$  et  $\text{item\_non\_livraison\_assignment\_cost} = 300$ .

Tableau 5-1 Types d'instances générées aléatoirement

|                   | Nb. Bins                                                     | Nb. Périodes | Volume items | Capacité Bins                                   |
|-------------------|--------------------------------------------------------------|--------------|--------------|-------------------------------------------------|
| <b>Ensemble 1</b> | <b>Nb. Items <math> I  = 100,200,300,500,750,1000</math></b> |              |              | <b>Nb. Nœuds <math> \mathcal{N}  = 3</math></b> |
| <b>Type 1</b>     | [10,100]                                                     | 10           | [1,20]       | [50,250]                                        |
| <b>Type 2</b>     | [20,100]                                                     | 20           | [1,20]       | [50,250]                                        |
| <b>Type 3</b>     | [50,100]                                                     | 30           | [1,20]       | [50,250]                                        |
| <b>Type 4</b>     | [50,150]                                                     | 10           | [1,20]       | [50,250]                                        |
| <b>Type 5</b>     | [50,200]                                                     | 15           | [1,20]       | [50,250]                                        |
| <b>Ensemble 2</b> | <b>Nb. Items <math> I  = 100,200,300,500,750,1000</math></b> |              |              | <b>Nb. Nœuds <math> \mathcal{N}  = 3</math></b> |
| <b>Type 1</b>     | [75,100]                                                     | 10           | [1,100]      | [20,250]                                        |
| <b>Type 2</b>     | [100,150]                                                    | 10           | [20,100]     | [20,250]                                        |

|               |           |    |          |          |
|---------------|-----------|----|----------|----------|
| <b>Type 3</b> | [150,200] | 10 | [50,100] | [20,250] |
| <b>Type 4</b> | [75,100]  | 20 | [1,100]  | [20,250] |
| <b>Type 5</b> | [100,150] | 20 | [20,100] | [20,250] |
| <b>Type 6</b> | [150,200] | 20 | [50,100] | [20,250] |

Les tests ont été réalisés sur un ordinateur doté d'un processeur à quatre cœurs de 2,6 Ghz et disposant de 16 Go de mémoire vive. Le temps de résolution maximal d'une instance a été limité à 1800 secondes. Afin de favoriser la reproductibilité des résultats générés lors de la résolution des instances, le nombre de threads utilisés par Cplex a été restreint à un seul. Les résultats des tests pour le premier ensemble d'instances sont présentés dans le tableau 5-2 et ceux du deuxième ensemble sont présentés dans le tableau 5-3. Dans ces tableaux, le *gap* d'optimalité est exprimé en pourcentage tandis que les temps de résolution sont indiqués en secondes.

Tableau 5-2 Résultats des tests de performances pour les instances du premier ensemble

| <b>Type</b> | <b>Nb items</b> | <b>Gap moyen</b> | <b>Temps CPU moyen<br/>(secondes)</b> | <b>Nb Optimales</b> | <b>Nb Réalisables</b> |
|-------------|-----------------|------------------|---------------------------------------|---------------------|-----------------------|
| Type 1      | 100             | 0,05             | 1103,27                               | 4                   | 10                    |
|             | 200             | 0,06             | 1800,02                               | 0                   | 10                    |
|             | 300             | 0,04             | 1800,02                               | 0                   | 10                    |
|             | 500             | 0,02             | 1800,05                               | 0                   | 10                    |
|             | 750             | 0,02             | 1800,05                               | 0                   | 10                    |
|             | 1000            | 0,02             | 1800,03                               | 0                   | 10                    |
| Type 2      | 100             | 0,00             | 14,99                                 | 10                  | 10                    |
|             | 200             | 0,00             | 480,96                                | 10                  | 10                    |
|             | 300             | 0,02             | 1273,95                               | 4                   | 10                    |
|             | 500             | 0,03             | 1800,03                               | 0                   | 10                    |
|             | 750             | 0,04             | 1800,03                               | 0                   | 10                    |
|             | 1000            | 0,04             | 1800,04                               | 0                   | 10                    |
| Type 3      | 100             | 0,00             | 4,87                                  | 10                  | 10                    |
|             | 200             | 0,00             | 245,65                                | 9                   | 10                    |
|             | 300             | 0,01             | 778,52                                | 8                   | 10                    |
|             | 500             | 0,03             | 1800,03                               | 0                   | 10                    |
|             | 750             | 0,11             | 1800,04                               | 0                   | 10                    |
|             | 1000            | 0,14             | 1800,04                               | 0                   | 10                    |
| Type 4      | 100             | 0,07             | 1296,52                               | 3                   | 10                    |

|        |      |      |         |   |    |
|--------|------|------|---------|---|----|
|        | 200  | 0,06 | 1800,03 | 0 | 10 |
|        | 300  | 0,04 | 1800,03 | 0 | 10 |
|        | 500  | 0,03 | 1800,07 | 0 | 10 |
|        | 750  | 0,02 | 1800,11 | 0 | 10 |
|        | 1000 | 0,01 | 1800,15 | 0 | 10 |
| Type 5 | 100  | 0,05 | 1112,97 | 5 | 10 |
|        | 200  | 0,05 | 1762,37 | 2 | 10 |
|        | 300  | 0,05 | 1800,05 | 0 | 10 |
|        | 500  | 0,04 | 1800,06 | 0 | 10 |
|        | 750  | 0,04 | 1800,08 | 0 | 10 |
|        | 1000 | 0,03 | 1800,06 | 0 | 10 |

Pour le premier ensemble d'instances, nous avons observé des temps de calcul relativement élevés. En effet, 235 des 300 instances résolues, soit 78 % d'entre elles, ont utilisé les 1800 secondes allouées pour leur résolution. À partir de 500 items, toutes les instances ont utilisé la totalité du temps imparti. Pour 100 items, les meilleures performances sont observées pour le type 3 (temps moyen de 4,87 s, gap nul), suivies du type 2 (temps moyen de 14,99 secondes, gap nul) et du type 1 (temps moyen de 1103,27 s, gap de 0,05). La principale différence entre les trois premiers types d'instances réside dans le nombre de périodes sur lesquelles le problème se déroule : 10 périodes pour le type 1, 20 pour le type 2 et 30 pour le type 3. Les moins bonnes performances ont été observées pour les instances de type 4 (temps moyen de 1296,52 secondes, gap de 0,07) et de type 5 (temps moyen de 1112,97 secondes, gap de 0,05), caractérisées par un nombre moyen de bins supérieur et un nombre de périodes inférieur.

Globalement, une augmentation du nombre de périodes améliore les performances, tandis qu'un nombre élevé de bins les dégrade. Cette tendance se maintient jusqu'à 300 items ; au-delà de 500, toutes les instances atteignent la limite de temps, mais les solutions demeurent de très bonne qualité. Cette tendance est illustrée à la figure 5-2, où l'on observe l'effet du nombre de périodes sur la qualité des solutions selon la taille des instances.

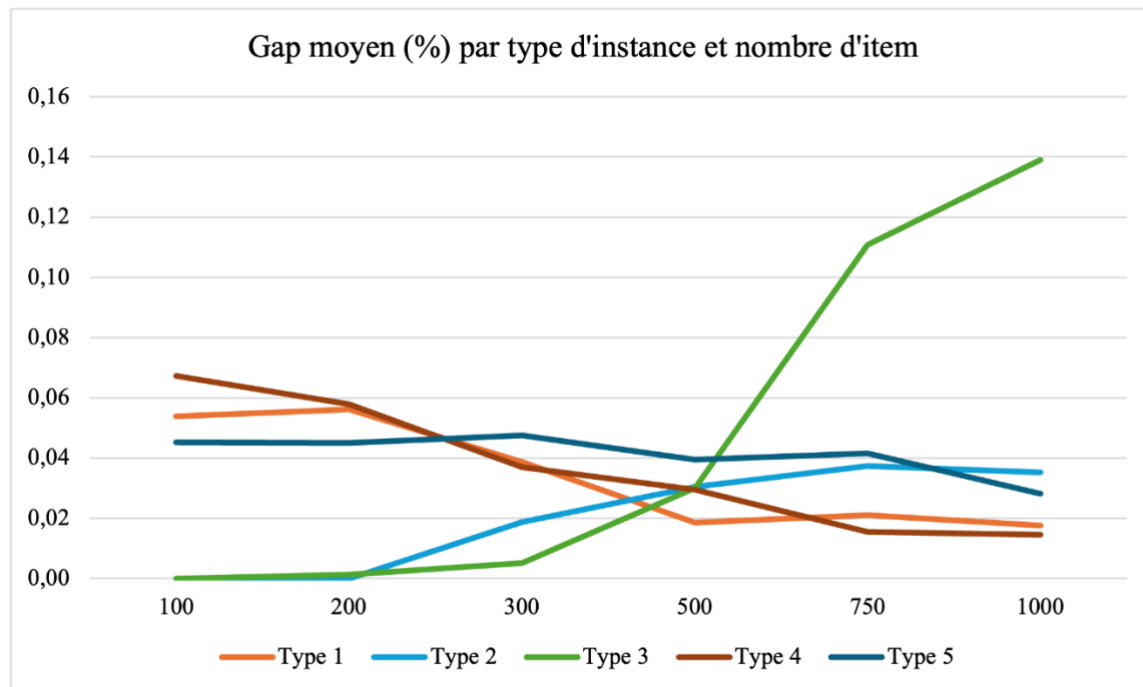


Figure 5-2 Gap moyen (%) par type d'instance et nombre d'items pour le premier ensemble

Cependant, les mauvaises performances au niveau du temps de calcul n'ont pas affecté significativement la qualité des solutions proposées. En effet, même si le délai maximal de résolution a été atteint pour 78% des instances, c'est-à-dire qu'une solution optimale, présentant un gap de 0, n'a pas pu être trouvée en 1800 secondes, nous avons été en mesure d'obtenir une solution réalisable pour l'ensemble des instances. De plus le gap d'optimalité moyen des solutions est de 0,03 %, avec un gap d'optimalité de 0,6 % dans le pire cas (c.-à-d., gap maximum observé dans pour l'ensemble des instances). Nous pouvons ainsi considérer que les solutions proposées sont d'excellente qualité même si le temps de résolution est élevé et que l'optimum n'est pas atteint.



Tableau 5-3 Résultats des tests de performances pour les instances du deuxième ensemble

| Type   | Nb Items | Gap moyen | Temps CPU moyen | Nb Optimales | Nb Réalisables |
|--------|----------|-----------|-----------------|--------------|----------------|
| Type 1 | 100      | 0,04      | 1800,01         | 0            | 10             |
|        | 200      | 0,04      | 1800,03         | 0            | 10             |
|        | 300      | 0,06      | 1800,03         | 0            | 10             |
|        | 500      | 0,02      | 1800,05         | 0            | 10             |
|        | 750      | 0,01      | 1800,09         | 0            | 10             |
|        | 1000     | 0,01      | 1800,08         | 0            | 10             |
| Type 2 | 100      | 0,04      | 1800,02         | 0            | 10             |
|        | 200      | 0,04      | 1800,05         | 0            | 10             |
|        | 300      | 0,10      | 1800,05         | 0            | 10             |
|        | 500      | 0,03      | 1800,09         | 0            | 10             |
|        | 750      | 0,02      | 1800,12         | 0            | 10             |
|        | 1000     | 0,01      | 1800,12         | 0            | 10             |
| Type 3 | 100      | 0,03      | 1800,03         | 0            | 10             |
|        | 200      | 0,03      | 1800,06         | 0            | 10             |
|        | 300      | 0,08      | 1800,07         | 0            | 10             |
|        | 500      | 0,03      | 1800,09         | 0            | 10             |
|        | 750      | 0,02      | 1800,13         | 0            | 10             |
|        | 1000     | 0,01      | 1800,16         | 0            | 10             |
| Type 4 | 100      | 0,03      | 1800,01         | 0            | 10             |
|        | 200      | 0,09      | 1800,02         | 0            | 10             |
|        | 300      | 0,11      | 1800,02         | 0            | 10             |
|        | 500      | 0,03      | 1800,03         | 0            | 10             |
|        | 750      | 0,02      | 1800,04         | 0            | 10             |
|        | 1000     | 0,01      | 1800,14         | 0            | 10             |
| Type 5 | 100      | 0,04      | 1800,01         | 0            | 10             |
|        | 200      | 0,07      | 1800,03         | 0            | 10             |
|        | 300      | 0,13      | 1800,03         | 0            | 10             |
|        | 500      | 0,04      | 1800,04         | 0            | 10             |
|        | 750      | 0,02      | 1800,05         | 0            | 10             |
|        | 1000     | 0,01      | 1800,08         | 0            | 10             |
| Type 6 | 100      | 0,02      | 1800,02         | 0            | 10             |
|        | 200      | 0,05      | 1800,03         | 0            | 10             |
|        | 300      | 0,13      | 1800,03         | 0            | 10             |
|        | 500      | 0,04      | 1800,04         | 0            | 10             |
|        | 750      | 0,02      | 1800,08         | 0            | 10             |
|        | 1000     | 0,02      | 1800,10         | 0            | 10             |

Pour le deuxième ensemble, toutes les instances ont utilisé la totalité des 1800 secondes allouées. Les performances temporelles ne permettent donc pas de comparaison significative entre types, mais la qualité des solutions demeure élevée : le gap moyen s'établit à 0,04 %, avec un écart maximal de 0,25 % observé sur l'ensemble des instances.

Les performances se dégradent progressivement jusqu'à 300 items, puis s'améliorent entre 300 et 1000 items, tandis que l'écart entre les différents types d'instances tend à s'atténuer (figure 5-3). On observe également que les instances comportant 20 périodes performant mieux pour les petits jeux (100 items), mais que cette tendance s'inverse à partir de 200 items, où les instances à 10 périodes deviennent plus performantes. Comme pour le premier ensemble, les temps de calcul élevés n'ont pas affecté la qualité des solutions obtenues : bien qu'aucune solution optimale n'ait été trouvée dans le délai imparti, une solution réalisable a pu être obtenue pour chacune des instances.

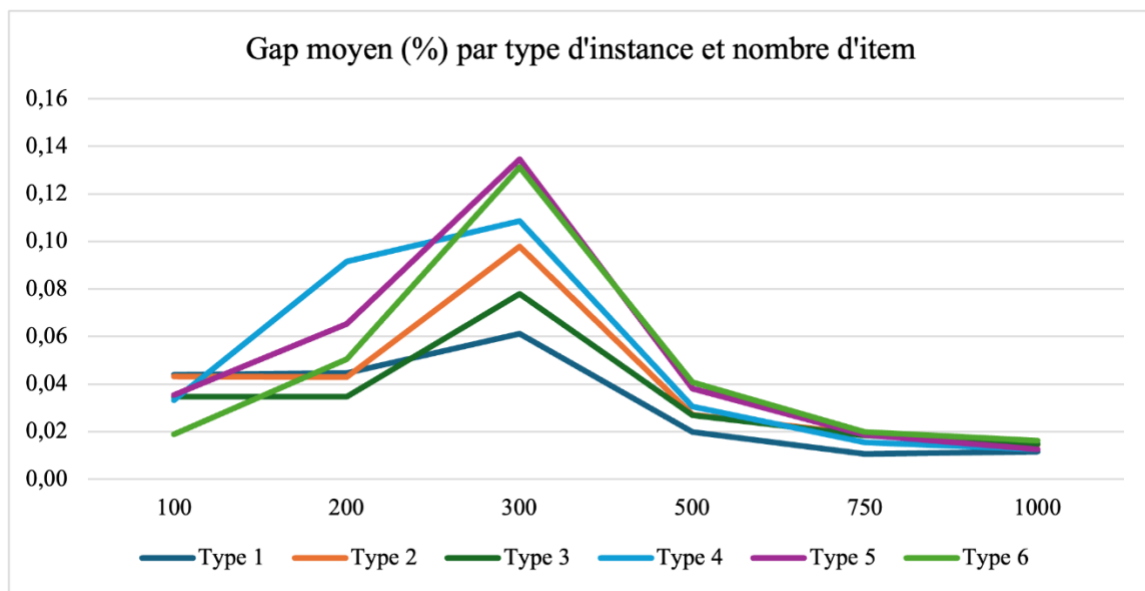


Figure 5-3 Gap (%) par type d'instance et nombre d'items pour le deuxième ensemble

En comparant le temps CPU moyen des deux ensembles (voir figure 5-4) ainsi que le gap moyen (voir figure 5-5), on constate que, dans les deux cas, les performances du premier ensemble sont supérieures à celles du deuxième. Ces résultats sont particulièrement étonnants, car ils vont à l'encontre de ceux obtenus par Crainic et al. (2021) lors des tests de performance de leur modèle de bin packing. Il convient toutefois de rappeler que l'effort computationnel ne se limite pas à l'identification d'une bonne solution réalisable : dans bien des cas, l'essentiel du temps de calcul est consacré à démontrer formellement que cette solution

est optimale. En d'autres termes, obtenir une solution proche de l'optimum peut être relativement rapide, alors que prouver son optimalité requiert un effort de calcul nettement plus important.

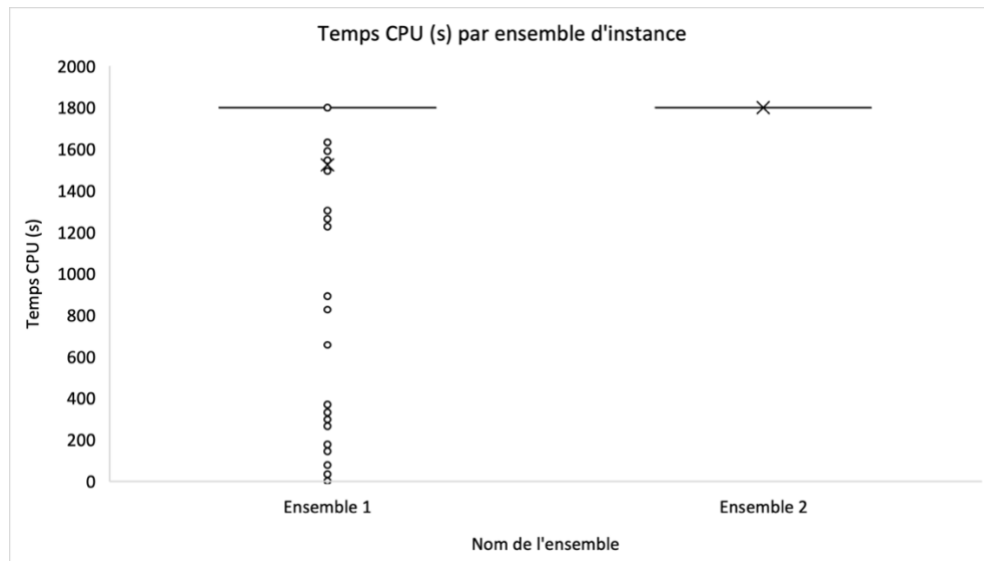


Figure 5-4 Temps CPU (s) par ensemble d'instances

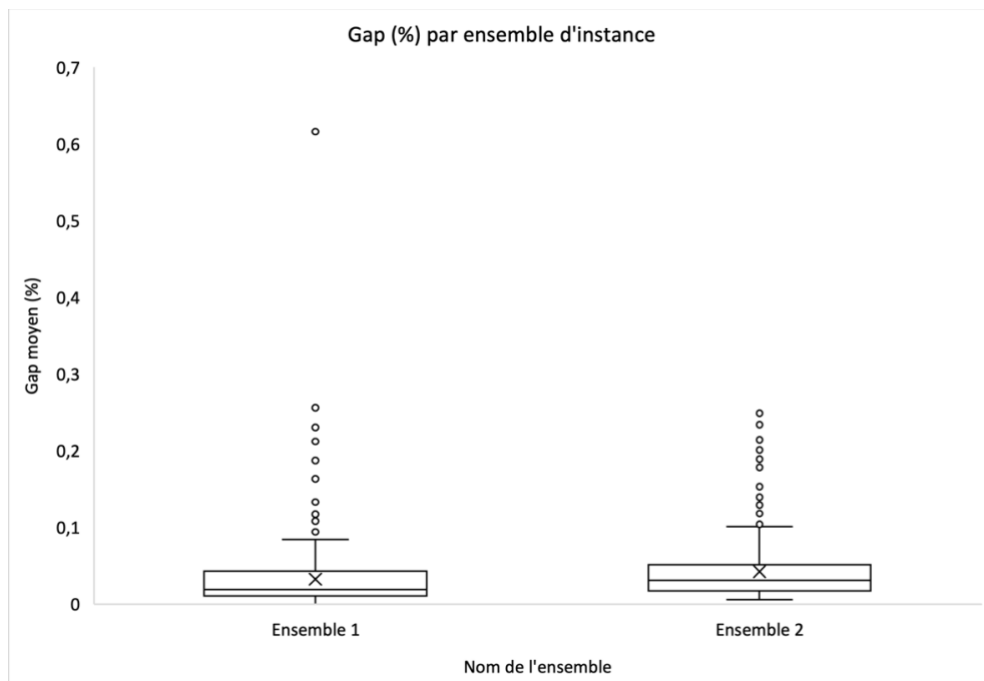


Figure 5-5 Gap (%) par ensemble d'instances

## 5.2 Test et évaluation de l’OAD selon les cas d’application au Parc Jean-Drapeau

Pour évaluer l’OAD dans son ensemble, nous avons ensuite appliqué la méthode d’évaluation des risques humains et de l’efficacité (Human Risk & Effectiveness Evaluation) proposée par Venable et al. (2016). Comme présenté dans le cadre méthodologique, cette approche met l’accent sur l’interaction entre les utilisateurs et l’artéfact, en évaluant sa facilité d’utilisation, son efficacité et son potentiel d’adoption. En mobilisant des scénarios concrets et des retours d’utilisateurs dans un environnement semi-contrôlé, elle vise à s’assurer que l’OAD répond aux exigences pratiques tout en favorisant l’adhésion des parties prenantes.

Dans notre cas, nous avons utilisé deux histoires d’utilisateurs pour illustrer le fonctionnement de l’OAD auprès des parties prenantes. Ces histoires ont permis de mettre en lumière deux cas d’usage concrets, correspondant à des problématiques de transport observées sur le terrain. Puisque les performances générales du modèle d’optimisation ont déjà été analysées plus tôt dans ce chapitre, notre objectif ici est d’évaluer plus spécifiquement ses performances lorsqu’il est appliqué à la résolution d’instances artificielles représentatives de ces situations réelles. Cette évaluation vise à valider l’efficacité opérationnelle de l’OAD et à estimer son potentiel d’adoption dans un contexte concret.

La première série de tests a montré qu’avec un temps de calcul maximal fixé à 1800 secondes, le MO était systématiquement en mesure de produire des solutions réalisables avec un bon niveau d’optimalité. Toutefois, l’obtention d’une solution optimale dans ce délai s’est révélée relativement rare. Or, en pratique, un temps de calcul de 1800 secondes peut s’avérer trop long selon les exigences du contexte opérationnel. Il devient donc pertinent de tester les performances de l’OAD sur des instances artificielles adaptées au contexte, tout en restreignant le temps de calcul à une durée plus réaliste. Cela permet de vérifier si une contrainte plus stricte sur le temps de résolution permet néanmoins d’obtenir des solutions de qualité, adaptées aux besoins du terrain.

Nous présenterons les deux cas d’utilisation mobilisés ainsi que les résultats des tests effectués sur les instances qu’ils ont permis de produire. Pour réaliser ces tests, nous avons utilisé la version alternative du programme de l’OAD, dont le code source est nommé *scriptOAD\_instancesAlea.py*, présenté plus haut dans ce chapitre.

Les tests pour les deux cas d'utilisation ont été effectués sur un ordinateur équipé d'un processeur à quatre cœurs cadencé à 2,6 GHz et de 16 Go de mémoire vive. Le temps de résolution maximal alloué à chaque instance variait selon le cas d'utilisation : il était limité à 300 secondes pour le premier cas et à 600 secondes pour le second. Afin de garantir la reproductibilité des résultats obtenus lors de la résolution des instances, le nombre de threads utilisés par CPLEX a été limité à un seul.

### **Premier cas d'utilisation**

Le premier cas d'utilisation concerne le transport de personnel entre les différents bureaux utilisés par la SPJD. Un gestionnaire de la SPJD nous a présenté la problématique suivante : les employés du parc travaillent depuis des bureaux répartis dans trois bâtiments situés sur l'île Notre-Dame. Le premier bâtiment est le pavillon des services, le deuxième est le pavillon 76, et le troisième est le pavillon du Canada. La figure 5-6 illustre la position de ces différents bâtiments sur le site du parc.

Dans le cadre de leur travail, ils doivent régulièrement se déplacer d'un bureau à l'autre. Ces bureaux étant suffisamment éloignés les uns des autres, les déplacements à pied sont peu efficaces, en particulier vers le bureau 3, situé à 3,3 km du bureau 1 et à 3,7 km du bureau 2, ce qui correspond à environ 24 à 25 minutes de marche selon Google Maps, comparativement à 8 à 9 minutes en voiture ou en voiturette de golf.

Actuellement, chaque employé dispose de son propre véhicule ou voiturette de golf, qu'il utilise individuellement selon ses besoins. Toutefois, selon le modèle considéré, chaque véhicule pourrait accueillir entre un et trois passagers supplémentaires. Les gestionnaires souhaitent donc mettre en place un système de covoiturage afin d'optimiser l'utilisation des véhicules et de réduire la circulation interne. Ils envisagent également le développement d'une application permettant aux employés de planifier leurs déplacements de façon à maximiser l'occupation des véhicules, dans une perspective écologique.



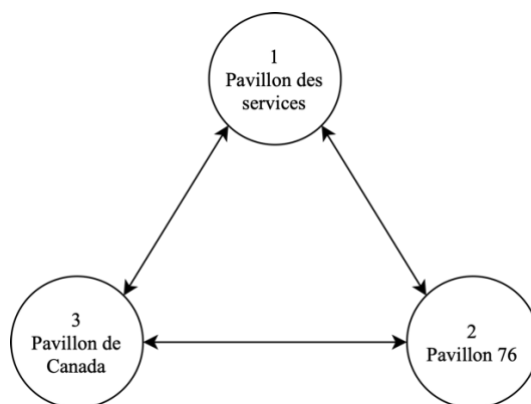


Figure 5-7 Représentation sous forme de graphe du problème de transport de personnel

Cette situation peut être modélisée comme un problème de transport dans un microcorridor, tel que présenté précédemment. En effet, nous avons : (1) un réseau de transport pouvant être divisé en microcorridors, représenté par un graphe cyclique bidirectionnel (voir figure 5-7), (2) un horaire de déplacement pouvant être décomposé en plusieurs périodes, et (3) des items (les employés de la SPJD) à déplacer dans un réseau en les transportant dans des bins (les différents véhicules).

Les détails de cette modélisation sont présentés dans le tableau 5-4, et les caractéristiques des items et des bins sont présentées dans le tableau 5-5.

Tableau 5-4 Modélisation du problème de transport de personnel dans un microcorridor

|                                   |                                                                                                                                                                  |
|-----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Microcorridor de transport</b> | Graphe cyclique bidirectionnel avec trois nœuds et trois sommets $G=(3,3)$                                                                                       |
| <b>Horaire de livraison</b>       | Transport ponctuel selon les besoins des employés. Possibilité de décortiquer l'horaire en période d'au moins 15 minutes (durée minimum pour effectuer un cycle) |
| <b>Items</b>                      | Employés de la SPJD                                                                                                                                              |
| <b>Bins</b>                       | Véhicules disponibles pour le transport des employés                                                                                                             |

Tableau 5-5 Principaux types de coût, Valeurs par défaut et valeurs personnalisées

| Catégories d'item  | Volume   | Exemple                            |
|--------------------|----------|------------------------------------|
| Employé de la SPJD | 1        |                                    |
| Catégorie de bin   | Capacité | Exemple                            |
| Petit              | 1        | Trottinette électrique             |
| Moyen              | 2        | Petite voiturette de golf          |
| Grand              | 4        | Voiture, grande voiturette de golf |

Pour tester les performances de notre OAD dans le contexte de cette problématique, nous avons choisi de générer trois ensembles, chacun composé de quatre types d'instances. Le premier ensemble contient un assortiment de bins de petite, moyenne et grande capacité. Le deuxième ensemble regroupe des bins de moyenne et grande capacité, tandis que le troisième ensemble comprend uniquement des bins de petite et moyenne capacité.

La différence entre les types d'instances au sein de chaque ensemble réside dans le nombre de périodes sur lesquelles se déroule le problème. Nous avons choisi de tester des cas avec 5, 10, 15 et 20 périodes, car ces valeurs sont réalistes dans notre contexte, une période correspondant à un intervalle minimal de 15 minutes. Par exemple, un problème se déroulant sur 20 périodes équivaut à la planification de déplacements aux 30 minutes sur une plage horaire de 10 heures, soit de 8 h à 18 h.

Pour chaque instance, le nombre de bins a été généré aléatoirement entre la moitié du nombre d'items et le nombre total d'items. Dans notre cas, le nombre d'items correspond au nombre d'employés souhaitant effectuer un déplacement, et on considère qu'il y a approximativement un véhicule disponible par employé. De plus chaque véhicule est disponible pour une durée allant de 1 jusqu'au nombre de périodes du problème.

Les tests ont été réalisés pour des tailles d'instances de 25, 50, 75, 100 et 200 items. Ces valeurs représentant des effectifs pertinents pour ce type de problématique, allant d'un faible à un fort volume opérationnel. Pour chaque type d'instance, les items sont disponibles pour une durée allant de 1 à 3 heures.



En ce qui concerne le calcul des coûts d'assignation bin/trajet, item/bin et item/non-livraison, nous avons utilisé les fonctions de coût d'utilisation par défaut, telles que définies au chapitre IV, paramétrées de la manière suivante :  $\text{bin\_trajet\_assignment\_cost} = 20$ ,  $\text{item\_bin\_assignment\_cost} = 2$  et  $\text{item\_non\_livraison\_assignment\_cost} = 300$ .

Le tableau 5-6 résume les différents paramètres utilisés pour générer les instances artificielles. Nous avons généré 10 instances pour chaque type, soit un total de 600 instances.

Tableau 5-6 Principaux types de coût, Valeurs par défaut et valeurs personnalisées

|                   | Nb. Bins                                             | Nb. Périodes | Volume items | Capacité bins                         | Dispo bins |
|-------------------|------------------------------------------------------|--------------|--------------|---------------------------------------|------------|
| <b>Ensemble 1</b> | <b>Nb. Items <math> I  = 25,50,75,100,200</math></b> |              |              | <b>Nb. Nœuds <math> N  = 3</math></b> |            |
| <b>Type 1</b>     | $[ I /2,  I ]$                                       | 5            | 1            | [1-4]                                 | [1-5]      |
| <b>Type 2</b>     | $[ I /2,  I ]$                                       | 10           | 1            | [1-4]                                 | [1-10]     |
| <b>Type 3</b>     | $[ I /2,  I ]$                                       | 15           | 1            | [1-4]                                 | [1-15]     |
| <b>Type 4</b>     | $[ I /2,  I ]$                                       | 20           | 1            | [1-4]                                 | [1-20]     |
| <b>Ensemble 2</b> | <b>Nb. Items <math> I  = 25,50,75,100,200</math></b> |              |              | <b>Nb. Nœuds <math> N  = 3</math></b> |            |
| <b>Type 1</b>     | $[ I /2,  I ]$                                       | 5            | 1            | [2-4]                                 | [1-5]      |
| <b>Type 2</b>     | $[ I /2,  I ]$                                       | 10           | 1            | [2-4]                                 | [1-10]     |
| <b>Type 3</b>     | $[ I /2,  I ]$                                       | 15           | 1            | [2-4]                                 | [1-15]     |
| <b>Type 4</b>     | $[ I /2,  I ]$                                       | 20           | 1            | [2-4]                                 | [1-20]     |
| <b>Ensemble 3</b> | <b>Nb. Items <math> I  = 25,50,75,100,200</math></b> |              |              | <b>Nb. Nœuds <math> N  = 3</math></b> |            |
| <b>Type 1</b>     | $[ I /2,  I ]$                                       | 5            | 1            | [1-2]                                 | [1-5]      |
| <b>Type 2</b>     | $[ I /2,  I ]$                                       | 10           | 1            | [1-2]                                 | [1-10]     |
| <b>Type 3</b>     | $[ I /2,  I ]$                                       | 15           | 1            | [1-2]                                 | [1-15]     |
| <b>Type 4</b>     | $[ I /2,  I ]$                                       | 20           | 1            | [1-2]                                 | [1-20]     |

Comme mentionné au début de cette section, le temps de résolution maximal d'une instance a été fixé à 300 secondes pour ce cas d'utilisation, puisqu'il s'agit d'un délai approprié dans ce contexte opérationnel. En plus du *gap* et du temps CPU, nous avons également choisi d'évaluer les performances du modèle en fonction du gain opérationnel, mesuré par comparaison avec le fonctionnement actuel, où un véhicule est requis par personne pour chaque déplacement. Par exemple, déplacer 25 personnes nécessiterait 25 véhicules (bins).

Le gain opérationnel, donné en pourcentage, pour chaque instance résolue a été calculé de la manière suivante :  $\frac{(NbItems - NbBinsSol)}{NbItems} \times 100$ , où : *NbItems* désigne le nombre d'items, c'est-à-dire le nombre de personnes souhaitant se déplacer dans l'instance ; *NbBinsSol* désigne le nombre de véhicules (*bins*) utilisés pour produire la solution optimale. Dans ce contexte, un gain opérationnel de 40 % indique une réduction équivalente du nombre de véhicules nécessaires pour répondre au même besoin de transport. Cette métrique permet ainsi de mieux évaluer l'impact de l'outil auprès des gestionnaires, en montrant dans quelle mesure il répond à leurs besoins réels de planification et d'optimisation des déplacements.

Les résultats des tests pour le premier ensemble d'instances sont présentés au tableau 5.7, ceux du deuxième ensemble au tableau 5-8, et ceux du troisième ensemble au tableau 5-9.

Tableau 5-7 Résultats des tests de performance pour les instances du premier ensemble

| Type   | Nb Items | Gap moyen (%) | Temps CPU moyen (secondes) | Nb Optimales | Nb Réalisables | Gain opérationnel moyen (%) |
|--------|----------|---------------|----------------------------|--------------|----------------|-----------------------------|
| Type 1 | 25       | 0,00          | 1,92                       | 10           | 10             | 62,4                        |
|        | 50       | 0,01          | 177,79                     | 5            | 10             | 70,8                        |
|        | 75       | 0,01          | 213,00                     | 3            | 10             | 72,3                        |
|        | 100      | 0,00          | 157,72                     | 5            | 10             | 72,3                        |
|        | 200      | 0,00          | 140,06                     | 6            | 10             | 71,7                        |
| Type 2 | 25       | 0,00          | 0,32                       | 10           | 10             | 47,6                        |
|        | 50       | 0,00          | 82,77                      | 8            | 10             | 63,2                        |
|        | 75       | 0,01          | 212,35                     | 3            | 10             | 71,5                        |
|        | 100      | 0,01          | 261,82                     | 2            | 10             | 71,6                        |
|        | 200      | 0,01          | 276,56                     | 1            | 10             | 71,7                        |
| Type 3 | 25       | 0,00          | 0,04                       | 10           | 10             | 40,0                        |
|        | 50       | 0,00          | 24,52                      | 10           | 10             | 55,6                        |
|        | 75       | 0,01          | 113,47                     | 7            | 10             | 64,5                        |
|        | 100      | 0,02          | 246,15                     | 2            | 10             | 68,5                        |
|        | 200      | 0,02          | 300,03                     | 0            | 10             | 71,1                        |
| Type 4 | 25       | 0,00          | 0,08                       | 10           | 10             | 38,4                        |
|        | 50       | 0,00          | 2,05                       | 10           | 10             | 51,8                        |
|        | 75       | 0,00          | 19,22                      | 10           | 10             | 59,6                        |
|        | 100      | 0,02          | 258,46                     | 3            | 10             | 63,5                        |
|        | 200      | 0,02          | 300,03                     | 0            | 10             | 69,7                        |

Pour le premier ensemble, nous avons obtenu un temps CPU moyen de 139,42 secondes et un gap d'optimalité moyen des solutions de 0,01 %, avec un gap d'optimalité de 0,06 % dans le pire cas (c.-à-d., le gap maximal observé pour l'ensemble des instances). De plus, 57,5 % des instances résolues ont produit une solution optimale. Nous observons également un gain opérationnel moyen de 62,9 % (variant de 28 % à 78,7 %), avec un gain minimal de 38,4 %, obtenu pour les instances de type 4 avec 25 items, et un gain maximal de 72,3 %, atteint pour les instances de type 1 avec 100 items.

Tableau 5-8 Résultats des tests de performance pour les instances du deuxième ensemble

| Type          | Nb Items | Gap moyen (%) | Temps CPU moyen (secondes) | Nb Optimales | Nb Réalisables | Gain opérationnel moyen (%) |
|---------------|----------|---------------|----------------------------|--------------|----------------|-----------------------------|
| <b>Type 1</b> | 25       | 0,00          | 3,82                       | 10           | 10             | 62,0                        |
|               | 50       | 0,01          | 201,54                     | 4            | 10             | 71,0                        |
|               | 75       | 0,02          | 300,00                     | 0            | 10             | 72,5                        |
|               | 100      | 0,01          | 300,01                     | 0            | 10             | 72,9                        |
|               | 200      | 0,00          | 300,02                     | 0            | 10             | 73,5                        |
| <b>Type 2</b> | 25       | 0,00          | 0,20                       | 10           | 10             | 51,6                        |
|               | 50       | 0,00          | 11,63                      | 10           | 10             | 63,4                        |
|               | 75       | 0,02          | 270,76                     | 2            | 10             | 69,9                        |
|               | 100      | 0,01          | 300,01                     | 0            | 10             | 72,3                        |
|               | 200      | 0,01          | 288,90                     | 2            | 10             | 73,8                        |
| <b>Type 3</b> | 25       | 0,00          | 0,03                       | 10           | 10             | 44,0                        |
|               | 50       | 0,00          | 38,31                      | 9            | 10             | 55,8                        |
|               | 75       | 0,01          | 134,01                     | 7            | 10             | 65,2                        |
|               | 100      | 0,02          | 257,83                     | 2            | 10             | 68,5                        |
|               | 200      | 0,03          | 300,03                     | 0            | 10             | 72,1                        |
| <b>Type 4</b> | 25       | 0,00          | 0,02                       | 10           | 10             | 39,6                        |
|               | 50       | 0,00          | 0,77                       | 10           | 10             | 54,4                        |
|               | 75       | 0,01          | 168,59                     | 5            | 10             | 58,7                        |
|               | 100      | 0,02          | 272,22                     | 1            | 10             | 64,5                        |
|               | 200      | 0,04          | 300,04                     | 0            | 10             | 70,9                        |

Pour le deuxième ensemble, nous avons obtenu un temps CPU moyen de 172,44 secondes et un gap d'optimalité moyen des solutions de 0,01 %, avec un gap d'optimalité de 0,09 % dans le pire cas (c.-à-d., le gap maximal observé pour l'ensemble des instances). De plus, 46 % des instances résolues ont produit une solution optimale. Nous observons également un gain opérationnel moyen de 63,8 %, avec un gain minimal de 39,6 %, obtenu pour les instances de type 4 avec 25 items, et un gain maximal de 73,8 %, atteint pour les instances de type 2 avec 200 items.

Tableau 5-9 Résultats des tests de performance pour les instances du troisième ensemble

| Type          | Nb Items | Gap moyen (%) | Temps CPU moyen (secondes) | Nb Optimales | Nb Réalisables | Gain opérationnel moyen (%) |
|---------------|----------|---------------|----------------------------|--------------|----------------|-----------------------------|
| <b>Type 1</b> | 25       | 0,01          | 62,84                      | 8            | 10             | 41,6                        |
|               | 50       | 0,00          | 0,18                       | 10           | 10             | 43,2                        |
|               | 75       | 0,00          | 0,35                       | 10           | 10             | 44,1                        |
|               | 100      | 0,00          | 0,19                       | 10           | 10             | 40,4                        |
|               | 200      | 0,00          | 0,64                       | 10           | 10             | 39,9                        |
| <b>Type 2</b> | 25       | 0,00          | 0,10                       | 10           | 10             | 35,6                        |
|               | 50       | 0,00          | 60,04                      | 8            | 10             | 40,4                        |
|               | 75       | 0,00          | 0,21                       | 10           | 10             | 39,5                        |
|               | 100      | 0,00          | 0,82                       | 10           | 10             | 41,0                        |
|               | 200      | 0,00          | 30,79                      | 9            | 10             | 42,3                        |
| <b>Type 3</b> | 25       | 0,00          | 0,03                       | 10           | 10             | 35,6                        |
|               | 50       | 0,00          | 34,48                      | 9            | 10             | 40,0                        |
|               | 75       | 0,00          | 90,46                      | 7            | 10             | 43,5                        |
|               | 100      | 0,00          | 0,50                       | 10           | 10             | 40,5                        |
|               | 200      | 0,00          | 2,87                       | 10           | 10             | 43,3                        |
| <b>Type 4</b> | 25       | 0,00          | 0,14                       | 10           | 10             | 36,0                        |
|               | 50       | 0,00          | 60,36                      | 8            | 10             | 40,0                        |
|               | 75       | 0,01          | 69,01                      | 8            | 10             | 42,1                        |
|               | 100      | 0,00          | 90,17                      | 7            | 10             | 42,3                        |
|               | 200      | 0,00          | 69,28                      | 8            | 10             | 42,3                        |

Pour le troisième ensemble, nous avons obtenu un temps CPU moyen de 28,67 secondes et un gap d'optimalité moyen des solutions de 0,00 %, avec un gap d'optimalité de 0,045 % dans le pire cas (c.-à-d., le gap maximal observé pour l'ensemble des instances). De plus, 91 % des instances résolues ont mené à

une solution optimale. Nous observons également un gain opérationnel moyen de 40,7 %, avec un gain minimal de 35,6 %, obtenu pour les instances de types 1 et 2 avec 25 items, et un gain maximal de 44,1 %, atteint pour les instances de type 1 avec 75 items.

En comparant les temps CPU des instances des trois ensembles (voir figure 5-8) ainsi que les gaps moyens (voir figure 5-9), on constate que les performances de résolution sont nettement supérieures pour le troisième ensemble. Contrairement aux deux premiers ensembles, l'ensemble 3 ne contient aucun véhicule de grande taille, c'est-à-dire ayant une capacité de 3 à 4 personnes. Étant donné que le volume des items est toujours fixé à 1 (chaque item représentant un individu à déplacer), une faible capacité des véhicules revient à augmenter la taille relative des items par rapport aux bins. Cependant, puisque chaque personne occupe minimalement 25 % de la capacité d'un véhicule (c.-à-d., dans le cas où une personne est seule dans un véhicule de 4 places), nous pouvons considérer que, dans le contexte de ce problème, tous les items sont de grande taille, si l'on veut comparer les résultats avec ceux obtenus lors de la première phase de tests.

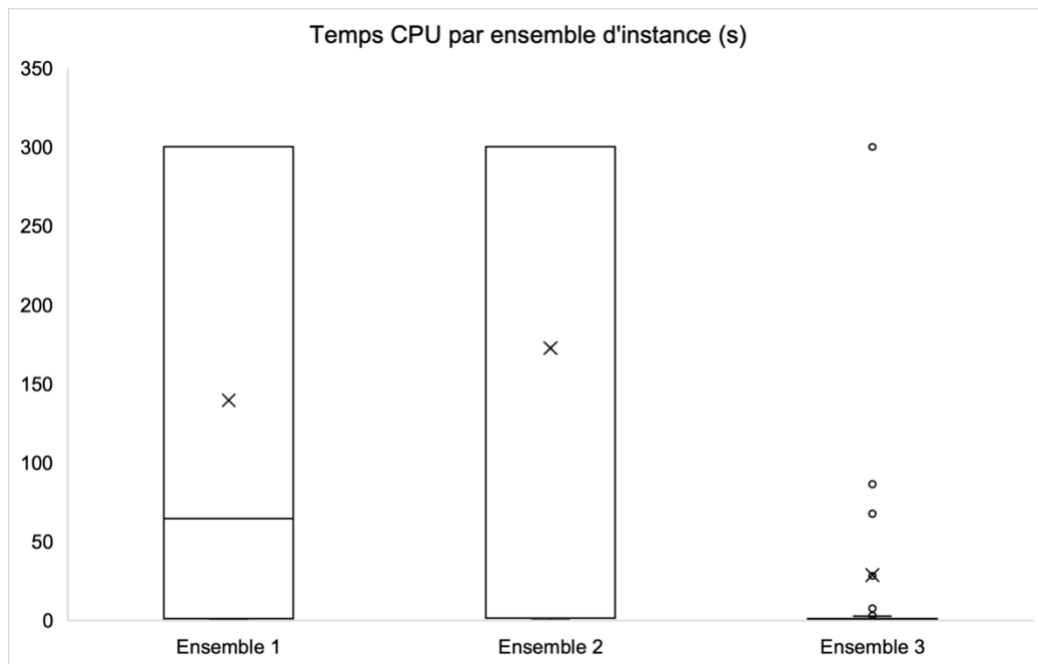


Figure 5-8 Temps CPU (s) par ensemble d'instances

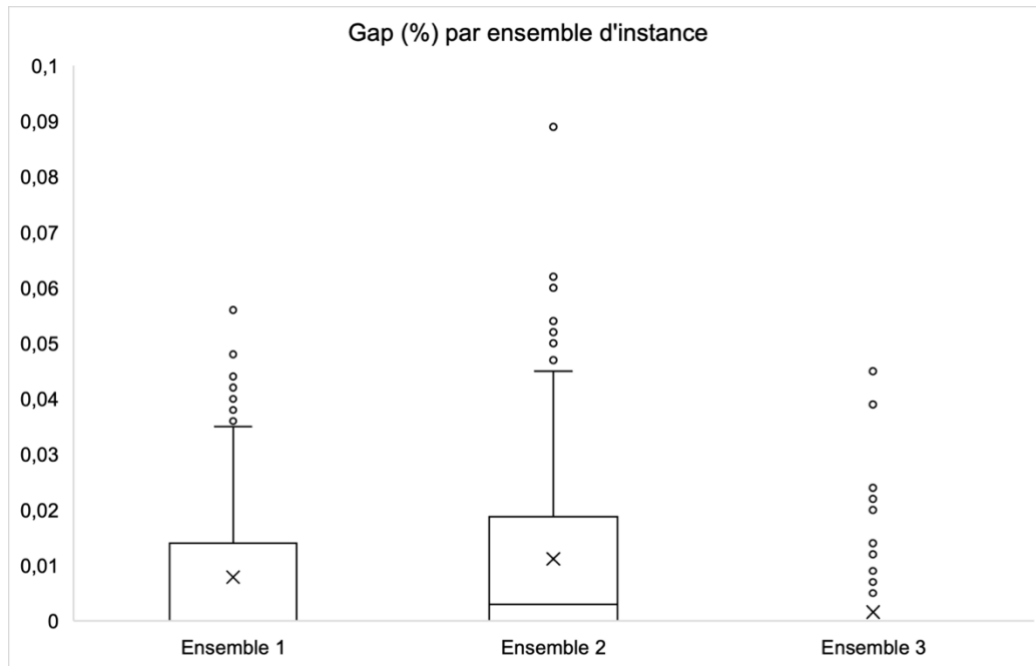


Figure 5-9 Gap (%) par ensemble d'instances

Dans le cas présent, l'impact de la présence de véhicules à plus grande capacité est significatif : les temps de résolution moyens des deux premiers ensembles sont en effet 5,8 fois plus élevés que ceux du troisième ensemble. Cet écart de performance pourrait être attribué à deux facteurs: (1) L'hétérogénéité des volumes des bins pour les instances générées aléatoirement et (2) le taux d'occupation moyen des bins par item (calculé en divisant le volume occupé par une personne, soit 1, par le volume moyen des bins de l'ensemble).

Concernant l'hétérogénéité, plus la dispersion des volumes entre les bins est élevée (c.-à-d., écart-type important), plus l'hétérogénéité du problème initial est grande. Cette hétérogénéité peut accroître le nombre de combinaisons possibles pour agencer les items et, par conséquent, augmenter la difficulté de trouver une solution optimale.

Dans le présent problème, la dispersion des items est nulle puisque le volume occupé par une personne est toujours fixé à 1. La seule source d'hétérogénéité provient donc du volume moyen des bins. En comparant les écarts-types des trois ensembles (voir tableau 5-10), nous constatons que le troisième

ensemble est considérablement plus homogène que les deux autres, l'écart-type étant 1,62 fois plus grand pour l'ensemble 2 et 2,22 fois plus grand pour l'ensemble 1.

Un autre facteur important à considérer est le taux d'occupation des bins par item. Un taux plus faible peut accroître le nombre de combinaisons possibles pour agencer les items au sein des bins et, par conséquent, augmenter la difficulté pour trouver une solution optimale. En comparant les taux d'occupation des trois ensembles (voir tableau 5-10), nous constatons que celui-ci est considérablement plus élevé pour le troisième ensemble, avec un taux d'occupation 1,66 fois plus grand que celui de l'ensemble 1 et 1,99 fois plus grand que celui de l'ensemble 2. Le temps de calcul moyen légèrement plus long pour le deuxième ensemble par rapport au premier pourrait également s'expliquer par un taux d'occupation moyen plus faible.

Tableau 5-10 Volume, écart-type et taux d'occupation par item moyen des bins par ensemble

| Ensemble | Capacité moyenne | Écart-type de la capacité moyenne | Taux d'occupation par item moyen |
|----------|------------------|-----------------------------------|----------------------------------|
| 1        | 2,506            | 1,108                             | 0,399                            |
| 2        | 3,000            | 0,810                             | 0,333                            |
| 3        | 1,507            | 0,499                             | 0,663                            |

Ensuite, l'analyse du gain opérationnel à travers les trois ensembles révèle deux tendances majeures. Premièrement, la capacité des véhicules a un impact significatif sur le potentiel de gain. Les ensembles 1 et 2, qui incluent des véhicules de moyenne et grande capacité, présentent des gains opérationnels élevés (respectivement 62,9 % et 63,8 %), tandis que l'ensemble 3, limité aux véhicules de petite capacité, enregistre un gain inférieur (40,7 %). Cette différence s'explique par le fait que des véhicules de plus grande capacité permettent de consolider davantage de déplacements, réduisant ainsi le nombre total de véhicules requis pour transporter tous les employés.

De plus, au sein de chaque ensemble, le gain opérationnel augmente avec le nombre d'items. Cette tendance est visible dans les trois ensembles : pour de petits nombres d'items (par exemple, 25 items), les gains sont plus modestes, tandis qu'ils deviennent beaucoup plus élevés lorsque le nombre d'items atteint 100 ou 200. Par exemple, dans l'ensemble 1, le gain passe de 38,4 % pour 25 items à 72,3 % pour 100 items ; dans l'ensemble 2, il passe de 39,6 % à 73,8 % ; et dans l'ensemble 3, de 35,6 % à 44,1 %. Cette

augmentation pourrait s'expliquer par un effet d'échelle : plus il y a d'items à regrouper, plus il existe de combinaisons permettant d'optimiser l'utilisation des véhicules, augmentant ainsi la qualité de la solution.

Nous pouvons également observer que les deux premiers ensembles présentent des temps de calcul supérieurs et des gaps d'optimalité légèrement plus élevés que ceux du troisième ensemble, mais qu'ils offrent en contrepartie un gain opérationnel supérieur. L'introduction de véhicules de plus grande capacité permet en effet de réduire significativement le nombre de véhicules nécessaires pour effectuer les déplacements, améliorant ainsi les gains opérationnels. Toutefois, cette amélioration se fait au détriment des performances du modèle en termes de temps de calcul et de difficulté accrue pour atteindre l'optimalité. Autrement dit, l'augmentation de la capacité des véhicules complexifie légèrement le problème d'optimisation, ce qui se traduit par des temps CPU plus longs et des gaps d'optimalité parfois légèrement supérieurs, mais elle génère en retour des bénéfices opérationnels importants.

Globalement, nous pouvons conclure que les performances du MO pour résoudre ce problème de transport sont satisfaisantes en termes de temps CPU, avec un temps moyen de 113,5 secondes, et excellentes en termes d'optimalité, avec un écart moyen de 0,01 % sur l'ensemble des trois types d'instances. Pour toutes les instances testées, l'écart d'optimalité n'a jamais dépassé 0,09 %, et une solution réalisable a toujours été obtenue. Ces indicateurs, le temps CPU et le gap d'optimalité, permettent donc d'évaluer la performance mathématique du modèle, c'est-à-dire sa capacité à produire rapidement des solutions de haute qualité.

Toutefois, la performance algorithmique seule ne garantit pas la pertinence managériale de l'approche. C'est pourquoi nous avons introduit la mesure du gain opérationnel, qui permet d'évaluer l'impact concret de l'OAD fondé sur notre modèle, en examinant dans quelle mesure il répond aux besoins réels des gestionnaires. En effet, un modèle performant sur le plan computationnel peut demeurer sans utilité s'il ne permet pas d'améliorer les opérations dans un contexte organisationnel donné.

Dans notre cas, l'analyse des gains opérationnels montre que l'utilisation d'un OAD fondé sur notre MO représente une piste de solution prometteuse pour un gestionnaire de la SPJD souhaitant implanter un système de partage de véhicules. Sur l'ensemble des instances générées, le gain opérationnel moyen atteint 55,8 %, ce qui signifie que la solution proposée permet de réduire de plus de la moitié le nombre de véhicules nécessaires pour assurer les déplacements internes entre les pavillons. Même dans la classe d'instances la moins performante (type 3 / 25 items), le gain opérationnel demeure de 35,6 %,



représentant une amélioration substantielle par rapport au système actuellement en place, où chaque employé dispose de son propre véhicule. Ces résultats présentent plusieurs retombées managériales : ils suggèrent que la mutualisation des véhicules permettrait d'optimiser les ressources existantes et de réduire les coûts d'exploitation, notamment en carburant, en entretien et en stationnement, tout en contribuant aux objectifs environnementaux du parc par la diminution du trafic interne. Ils mettent également en lumière le potentiel d'une intégration de l'OAD au sein d'une application numérique de gestion des déplacements, qui viendrait soutenir les gestionnaires dans la planification en temps réel et la prise de décision stratégique. Ainsi, le modèle proposé ne se limite pas à une optimisation mathématique, mais constitue une piste prometteuse pour améliorer la gestion durable et intelligente de la mobilité interne au sein du Parc Jean-Drapeau.

## **Deuxième cas d'utilisation**

Le second cas d'utilisation porte sur le transport de marchandises dans le cadre de tâches de maintenance récurrentes au sein du Parc Jean-Drapeau. Il s'appuie sur les données disponibles concernant le type de marchandises (volume, poids, destination) et les itinéraires requis. Les données fournies par les gestionnaires du parc nous ont permis d'identifier la nature des objets et des véhicules utilisés dans les opérations courantes, ainsi que de définir un itinéraire représentatif pour le transport de marchandises.



Figure 5-10 Position des points de livraisons pour le problème de transport de marchandises

Nous avons ainsi sélectionné trois lieux entre lesquels transitent régulièrement les marchandises : le Pavillon des services, où sont entreposées la majorité des fournitures (point 1 sur la figure 5-10), le Complexe aquatique, qui requiert une grande quantité de matériel pour l'entretien de ses installations (point 2), et le Pavillon du Canada (point 3).

Afin de générer des instances correspondant à nos scénarios d'intérêt, nous avons classé les items et les bins en trois catégories. Pour chacune de ces catégories, nous avons défini un intervalle de valeurs possibles pour le volume des items et la capacité des bins.

Les volumes sont exprimés en litres (dm<sup>3</sup>) et les intervalles ont été établis à partir d'exemples concrets de marchandises et de véhicules figurant dans les données fournies par les responsables du parc. Le tableau 5-11 présente les différentes catégories d'items et de bins, les plages de valeurs associées ainsi que des exemples concrets. Le vélo cargo, bien qu'actuellement non utilisé par le parc, a été inclus dans les scénarios testés puisqu'il est envisagé comme solution potentielle pour réduire l'empreinte écologique des déplacements internes.

Tableau 5-11 Principaux types de coût, Valeurs par défaut et valeurs personnalisées

| Catégories d'item | Intervalle de volume   | Exemple               |
|-------------------|------------------------|-----------------------|
| Petit             | [1,10]                 | Bidon de savon à main |
| Moyen             | [10,50]                | Chaudière de chlore   |
| Grand             | [50,120]               | Outils volumineux     |
| Catégorie de bin  | Intervalle de capacité | Exemple               |
| Petit             | [250-350]              | Vélo cargo            |
| Moyen             | [430-600]              | Voiturette de golf    |
| Grand             | [850-1200]             | Camionnette           |

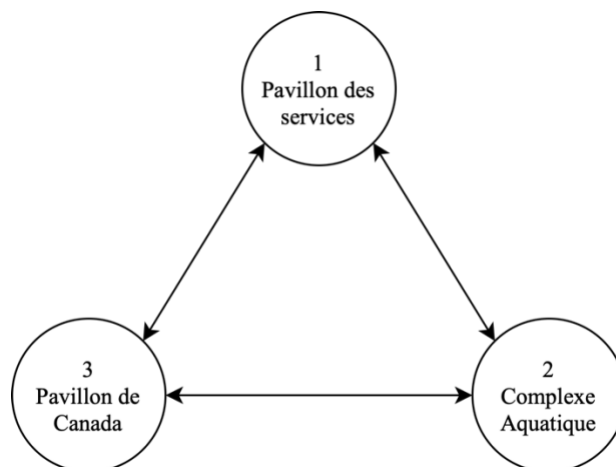


Figure 5-11 Représentation sous forme de graphe du problème de transport de marchandises

Cette situation peut être modélisée comme un problème de transport dans un microcorridor, tel que présenté précédemment. En effet, elle comprend : (1) un réseau de transport représenté par un graphe cyclique bidirectionnel (voir figure 5-11), (2) un horaire de livraison pouvant être découpé en périodes de 15 minutes, et (3) des items (marchandises) à acheminer à l'aide de bins (véhicules).

Tableau 5-12 Modélisation du problème de transport de marchandises pour l'exécution de tâches de maintenance récurrentes

|                                   |                                                                                                                                                   |
|-----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Microcorridor de transport</b> | Graphe cyclique bidirectionnel avec trois nœuds et trois sommets $G=(3,3)$                                                                        |
| <b>Horaire de livraison</b>       | Transport ponctuel de marchandises. Possibilité de décortiquer l'horaire en période d'au moins 15 minutes (durée minimum pour effectuer un cycle) |
| <b>Items</b>                      | Divers type de marchandises                                                                                                                       |
| <b>Bins</b>                       | Véhicules disponibles pour le transport de marchandises                                                                                           |

Pour tester les performances de l'OAD dans ce contexte, nous avons généré trois ensembles d'instances. Les deux premiers sont composés de six types d'instances, et le troisième de quatre types. Le premier ensemble comporte des bins de capacité moyenne à grande avec des items de faible volume. Le deuxième ensemble présente des bins de capacité moyenne à grande également, mais avec des items de volume moyen à élevé. Le troisième ensemble, quant à lui, regroupe des bins de petite à moyenne capacité, associés à des items de toutes tailles.

La distinction entre les types d'instances repose sur le nombre de périodes de planification, soit 5, 10, 15 ou 20 périodes, des valeurs réalistes considérant qu'une période correspond à un minimum de 15 minutes. Les tests ont été réalisés pour des tailles d'instances de 50, 100, 200, 300 et 500 items. Ces effectifs représentant des scénarios plausibles dans ce contexte, allant d'un faible à un fort volume opérationnel. Pour chaque instance, les items sont disponibles pour une durée aléatoire comprise entre 2 périodes et la période maximale de l'instance. Le tableau 5-13 résume les différents paramètres utilisés pour générer les instances artificielles. Nous avons généré et résolu 10 instances pour chaque type, pour un total de 800 instances.

Tableau 5-13 Principaux types de coût, Valeurs par défaut et valeurs personnalisées

|                   | Nb. Bins                                               | Nb. Périodes | Volume items | Capacité bins                         | Dispo bins |
|-------------------|--------------------------------------------------------|--------------|--------------|---------------------------------------|------------|
| <b>Ensemble 1</b> | <b>Nb. Items <math> I  = 50,100,200,300,500</math></b> |              |              | <b>Nb. Nœuds <math> N  = 3</math></b> |            |
| <b>Type 1</b>     | [15,35]                                                | 5            | [1-10]       | [430-1200]                            | [5-5]      |
| <b>Type 2</b>     | [15,35]                                                | 10           | [1-10]       | [430-1200]                            | [10-10]    |
| <b>Type 3</b>     | [15,35]                                                | 15           | [1-10]       | [430-1200]                            | [15-15]    |
| <b>Type 4</b>     | [20,100]                                               | 5            | [1-10]       | [430-1200]                            | [5-5]      |
| <b>Type 5</b>     | [20,100]                                               | 10           | [1-10]       | [430-1200]                            | [10-10]    |
| <b>Type 6</b>     | [20,100]                                               | 15           | [1-10]       | [430-1200]                            | [15-15]    |
| <b>Ensemble 2</b> | <b>Nb. Items <math> I  = 50,100,200,300,500</math></b> |              |              | <b>Nb. Nœuds <math> N  = 3</math></b> |            |
| <b>Type 1</b>     | [10,50]                                                | 5            | [1-120]      | [430-1200]                            | [5-5]      |
| <b>Type 2</b>     | [10,50]                                                | 10           | [1-120]      | [430-1200]                            | [10-10]    |
| <b>Type 3</b>     | [10,50]                                                | 15           | [1-120]      | [430-1200]                            | [15-15]    |
| <b>Type 4</b>     | [20,100]                                               | 5            | [1-120]      | [430-1200]                            | [5-5]      |
| <b>Type 5</b>     | [20,100]                                               | 10           | [1-120]      | [430-1200]                            | [10-10]    |
| <b>Type 6</b>     | [20,100]                                               | 15           | [1-120]      | [430-1200]                            | [15-15]    |
| <b>Ensemble 3</b> | <b>Nb. Items <math> I  = 50,100,200,300,500</math></b> |              |              | <b>Nb. Nœuds <math> N  = 3</math></b> |            |
| <b>Type 1</b>     | [15,35]                                                | 10           | [1-10]       | [250-600]                             | [10-10]    |
| <b>Type 2</b>     | [20,100]                                               | 10           | [1-120]      | [250-600]                             | [10-10]    |
| <b>Type 3</b>     | [15,35]                                                | 20           | [1-10]       | [250-600]                             | [20-20]    |
| <b>Type 4</b>     | [20,100]                                               | 20           | [1-120]      | [250-600]                             | [20-20]    |

De plus, pour ces instances, nous avons choisi d'utiliser des fonctions de coût personnalisées pour l'assignation item/bin et bin/trajet.

Lorsque l'item est livré après son moment de livraison dû, noté  $\delta_i$ , le coût d'assignation item/bin est calculé selon la formule suivante :  $\text{coûtFixeItemBin} + \sqrt{v_i} + (|\delta_i - t| * 5)$ , où :  $\text{coûtFixeItemBin}$  correspond au coût fixe d'assignation définie par le champ *item\_bin\_assignment\_cost* du fichier *param.cfg*;  $v_i$  correspond au volume de l'item  $i \in I$ ;  $\delta_i$  correspond au moment de livraison dû de l'item  $i \in I$ ;  $t$  correspond à la période  $t \in \mathcal{T}$

Si l'item est livré avant son moment de livraison dû, la pénalité est nulle, et le coût d'assignation est réduit à :  $\text{coûtFixeItemBin} + \sqrt{v_i}$ . Cette formule permet de pénaliser les livraisons en retard, ce qui est crucial dans ce contexte où le respect des délais est une priorité opérationnelle.

Le coût d’assignation bin/trajet est quant à lui calculé selon la formule :  $\sqrt{|p| * V_j} + f_j$ , où :  $|p|$  Correspond à la cardinalité du trajet  $p \in \mathcal{P}$ ;  $V_j$  correspond à la capacité maximale de la bin  $j \in J$ ;  $f_j$  correspond au coût fixe d’utilisation de la bin  $j \in J$ . L’objectif de cette formule est de pondérer le coût d’affectation d’une bin à un trajet en fonction de la longueur du trajet effectué.

La figure 5-12 présente le code utilisé par le programme pour calculer le coût associé aux différentes assignations. Le code des fonctions de coût personnalisé se trouve dans le fichier *cost\_function.py*.

```
#composantes de n = [i,j,p,t] ==> n[0] = i ; n[1] = j ; n[2] = p ; n[3] = t
def item_bin_assignment_cost(n,model): #ajouter coût selon interval de livraison
    coutRetard = 0
    if model.itemLivraisonDu[n[0]-1] - n[3] < 0: coutRetard = abs(model.itemLivraisonDu[n[0]-1] - n[3]) * 5
    return model.item_bin_assignment_cost_def + math.sqrt(model.item_volume[n[0]-1]) + coutRetard

#composantes de n = [j,p,t] ==> n[0] = j ; n[1] = p ; n[2] = t
def bin_trajet_assignment_cost(n,model):
    #print(math.sqrt(len(model.trajets_p[n[1]-1]) * model.bin_volume_max[n[0]-1]) + model.bin_trajet_assignment_cost_def)
    return math.sqrt(len(model.trajets_p[n[1]-1]) * model.bin_volume_max[n[0]-1]) + model.bin_trajet_assignment_cost_def

#composantes de n = [i,t]
def item_non_livraison_assignment_cost(n,model):
    return model.item_non_livraison_assignment_cost_def
```

Figure 5-12 Fonctions de coût personnalisées dans le fichier *cost\_function.py*

Les tests ont été réalisés sur un ordinateur doté d’un processeur à quatre cœurs de 2,6 Ghz et disposant de 16 Go de mémoire vive. Le temps de résolution maximal d’une instance a été limité à 600 secondes. Afin de favoriser la reproductibilité des résultats générés lors de la résolution des instances, le nombre de threads utilisés par Cplex a été restreint à un seul. Les résultats des tests pour le premier ensemble d’instances sont présentés dans le tableau 5-14, ceux du deuxième ensemble sont présentés dans le tableau 5-15 et ceux du troisième ensemble sont présentés dans le tableau 5-16.

Tableau 5-14 Résultats des tests de performance pour les instances du premier ensemble

| Type      | Nb Items | Gap moyen (%) | Temps CPU moyen (secondes) | Nb Optimales | Nb Réalisables |
|-----------|----------|---------------|----------------------------|--------------|----------------|
| <b>T1</b> | 50       | 0,00          | 2,71                       | 10           | 10             |
|           | 100      | 0,00          | 7,93                       | 10           | 10             |
|           | 200      | 0,00          | 26,59                      | 10           | 10             |
|           | 300      | 0,00          | 54,47                      | 10           | 10             |
|           | 500      | 0,00          | 132,41                     | 10           | 10             |
| <b>T2</b> | 50       | 0,00          | 7,07                       | 10           | 10             |
|           | 100      | 0,00          | 21,22                      | 10           | 10             |
|           | 200      | 0,00          | 111,01                     | 10           | 10             |
|           | 300      | 0,00          | 223,06                     | 10           | 10             |
|           | 500      | 0,01          | 441,12                     | 7            | 10             |
| <b>T3</b> | 50       | 0,00          | 13,07                      | 10           | 10             |
|           | 100      | 0,00          | 67,27                      | 10           | 10             |
|           | 200      | 0,01          | 253,99                     | 9            | 10             |
|           | 300      | 0,01          | 393,62                     | 7            | 10             |
|           | 500      | 0,09          | 600,06                     | 0            | 10             |
| <b>T4</b> | 50       | 0,02          | 211,79                     | 7            | 10             |
|           | 100      | 0,02          | 151,81                     | 8            | 10             |
|           | 200      | 0,04          | 303,32                     | 7            | 10             |
|           | 300      | 0,04          | 404,84                     | 5            | 10             |
|           | 500      | 0,07          | 483,99                     | 3            | 10             |
| <b>T5</b> | 50       | 0,02          | 204,56                     | 8            | 10             |
|           | 100      | 0,02          | 180,55                     | 8            | 10             |
|           | 200      | 0,08          | 394,87                     | 6            | 10             |
|           | 300      | 0,09          | 434,12                     | 5            | 10             |
|           | 500      | 0,30          | 586,40                     | 1            | 10             |
| <b>T6</b> | 50       | 0,01          | 172,70                     | 9            | 10             |
|           | 100      | 0,08          | 393,19                     | 4            | 10             |
|           | 200      | 0,16          | 550,85                     | 1            | 10             |
|           | 300      | 0,21          | 512,60                     | 3            | 10             |
|           | 500      | 0,47          | 600,12                     | 0            | 10             |

Pour le premier ensemble, nous avons obtenu un temps CPU moyen de 264,71 secondes et un gap moyen de 0,06. Le gap le plus élevé observé est de 0,6. De plus, 208 des 300 instances résolues, soit 69 %, ont mené à une solution optimale, mais les 300 solutions sont réalisables.

Tableau 5-15 Résultats des tests de performance pour les instances du deuxième ensemble

| Type      | Nb Items | Gap moyen (%) | Temps CPU moyen | Nb Optimales | Nb Réalisables |
|-----------|----------|---------------|-----------------|--------------|----------------|
| <b>T1</b> | 50       | 0,00          | 7,35            | 10           | 10             |
|           | 100      | 0,01          | 451,87          | 4            | 10             |
|           | 200      | 0,02          | 600,01          | 0            | 10             |
|           | 300      | 0,04          | 600,02          | 0            | 10             |
|           | 500      | 0,07          | 600,04          | 0            | 10             |
| <b>T2</b> | 50       | 0,00          | 14,22           | 10           | 10             |
|           | 100      | 0,00          | 232,16          | 8            | 10             |
|           | 200      | 0,04          | 600,02          | 0            | 10             |
|           | 300      | 0,05          | 600,03          | 0            | 10             |
|           | 500      | 0,08          | 600,06          | 0            | 10             |
| <b>T3</b> | 50       | 0,00          | 93,07           | 10           | 10             |
|           | 100      | 0,01          | 452,49          | 4            | 10             |
|           | 200      | 0,06          | 600,03          | 0            | 10             |
|           | 300      | 0,09          | 600,04          | 0            | 10             |
|           | 500      | 0,18          | 600,09          | 0            | 10             |
| <b>T4</b> | 50       | 0,00          | 65,55           | 10           | 10             |
|           | 100      | 0,03          | 470,95          | 5            | 10             |
|           | 200      | 0,05          | 600,02          | 0            | 10             |
|           | 300      | 0,05          | 600,03          | 0            | 10             |
|           | 500      | 0,03          | 600,08          | 0            | 10             |
| <b>T5</b> | 50       | 0,01          | 166,73          | 9            | 10             |
|           | 100      | 0,06          | 490,95          | 3            | 10             |
|           | 200      | 0,08          | 600,04          | 0            | 10             |
|           | 300      | 0,14          | 600,08          | 0            | 10             |
|           | 500      | 0,13          | 600,10          | 0            | 10             |
| <b>T6</b> | 50       | 0,01          | 352,42          | 8            | 10             |
|           | 100      | 0,08          | 418,14          | 4            | 10             |
|           | 200      | 0,07          | 600,04          | 0            | 10             |
|           | 300      | 0,19          | 600,08          | 0            | 10             |
|           | 500      | 0,21          | 600,11          | 0            | 10             |



Pour le deuxième ensemble, nous avons obtenu un temps CPU moyen de 467,23 secondes et un gap moyen de 0,06. Le gap le plus élevé observé est de 0,21. De plus, 85 des 300 instances résolues, soit 28 %, ont mené à une solution optimale, mais les 300 solutions sont réalisables.

Tableau 5-16 Résultats des tests de performance pour les instances du troisième ensemble

| Type      | Nb Items | Gap moyen (%) | Temps CPU moyen (secondes) | Nb Optimales | Nb Réalisables |
|-----------|----------|---------------|----------------------------|--------------|----------------|
| <b>T1</b> | 50       | 0,00          | 9,26                       | 10           | 10             |
|           | 100      | 0,00          | 24,85                      | 10           | 10             |
|           | 200      | 0,00          | 113,84                     | 10           | 10             |
|           | 300      | 0,00          | 225,07                     | 9            | 10             |
|           | 500      | 0,00          | 431,77                     | 9            | 10             |
| <b>T2</b> | 50       | 0,00          | 304,59                     | 7            | 10             |
|           | 100      | 0,04          | 600,03                     | 0            | 10             |
|           | 200      | 0,11          | 600,04                     | 0            | 10             |
|           | 300      | 0,10          | 600,08                     | 0            | 10             |
|           | 500      | 0,06          | 600,09                     | 0            | 10             |
| <b>T3</b> | 50       | 0,00          | 68,11                      | 10           | 10             |
|           | 100      | 0,00          | 136,22                     | 10           | 10             |
|           | 200      | 0,01          | 425,89                     | 5            | 10             |
|           | 300      | 0,03          | 474,98                     | 4            | 10             |
|           | 500      | 0,19          | 600,06                     | 0            | 10             |
| <b>T4</b> | 50       | 0,02          | 407,95                     | 6            | 10             |
|           | 100      | 0,07          | 600,03                     | 0            | 10             |
|           | 200      | 0,19          | 600,07                     | 0            | 10             |
|           | 300      | 0,21          | 600,57                     | 0            | 10             |
|           | 500      | 0,12          | 600,15                     | 0            | 10             |

Pour le troisième ensemble, nous avons obtenu un temps CPU moyen de 401,18 secondes et un gap moyen de 0,06. Le gap le plus élevé observé est de 0,43. De plus, 90 des 200 instances résolues, soit 45 % ont mené à une solution optimale, mais les 300 solutions sont réalisables.

En comparant le temps CPU moyen des trois ensembles (voir figure 5-13) ainsi que le gap moyen (voir figure 5-14), on constate que le premier ensemble, qui comporte des items de plus petite taille, offre de meilleures performances que le deuxième en termes de temps de calcul et d'optimalité. Le troisième ensemble, qui inclut à la fois des instances avec de petits items et d'autres avec des items de plus grande taille, présente des performances intermédiaires entre les deux premiers ensembles.

Nous avons également observé qu'en se limitant à des instances de 500 items et à un temps de résolution maximal de 600 secondes, les performances tendent à décroître avec l'augmentation du nombre d'items pour les trois ensembles d'instances. Les gains de performance observés à partir de 300 items lors de la première phase de tests ne sont ici pas reproduits, ce qui laisse présager qu'un temps de calcul plus long ou une augmentation de la taille des instances soit nécessaire pour que cet effet soit de nouveau observable.

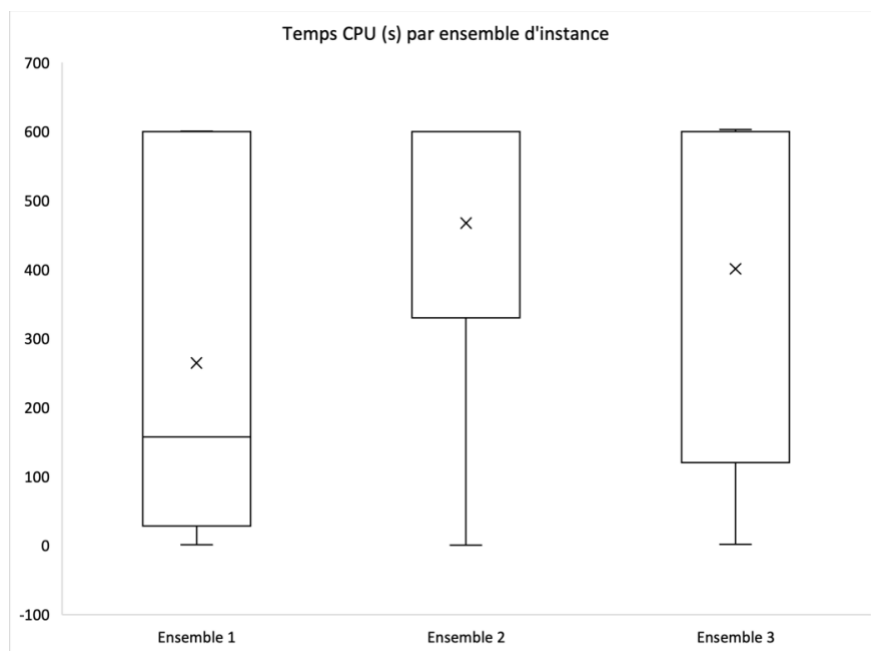


Figure 5-13 Temps CPU moyen (s) par ensemble d'instances

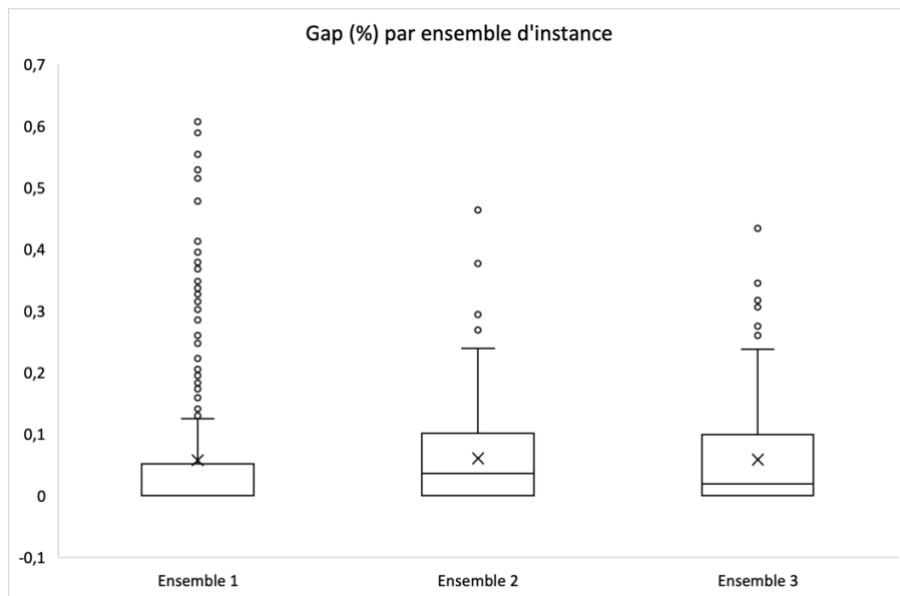


Figure 5-14 Gap moyen (%) par ensemble d'instances

Ces résultats confirment une nouvelle fois les constats formulés lors de la première phase de test selon lesquels les instances comportant des items de grande taille relative par rapport à la capacité des bins sont généralement plus complexes à résoudre.

Globalement, nous pouvons conclure que les performances du MO pour résoudre ce problème de transport sont satisfaisantes en termes de temps de calcul, avec un temps moyen de 374,77 secondes, et également en termes d'optimalité, avec un gap moyen de 0,06 sur l'ensemble des trois ensembles d'instances. Pour toutes les instances testées, le gap est resté inférieur à 0,6 et une solution réalisable a toujours pu être obtenue. Ces résultats suggèrent que l'OAD est en mesure de produire des solutions de qualité, même dans un contexte contraint par un temps de calcul limité et l'usage de fonctions de coût personnalisées.

Au-delà de ces performances techniques, il importe d'examiner la portée managériale des résultats obtenus, afin d'évaluer dans quelle mesure l'OAD peut concrètement soutenir la planification et l'organisation des opérations logistiques au sein du Parc Jean-Drapeau.

Ainsi, les résultats obtenus dans ce second cas d'utilisation suggèrent que l'intégration d'un OAD fondé sur notre modèle d'optimisation constitue une piste de solution prometteuse pour les gestionnaires du Parc Jean-Drapeau souhaitant améliorer la planification et la logistique interne des livraisons. En tenant compte à la fois des délais de livraison et des capacités de charge, l'OAD favorise une meilleure coordination entre les équipes de maintenance et une utilisation plus efficiente des véhicules disponibles. Ces résultats laissent entrevoir la possibilité de réduire les déplacements redondants, d'améliorer la ponctualité des livraisons et de mieux anticiper les besoins opérationnels. De plus, le fait que le modèle soit en mesure de produire des solutions de qualité en un temps restreint constitue un avantage notable, puisqu'il permet de réaliser de nouvelles simulations en moins de dix minutes lorsque le contexte opérationnel l'exige. Il demeure toutefois plus difficile de quantifier précisément les gains opérationnels dans ce cas précis, en raison du manque de données sur le fonctionnement actuel des livraisons, ce qui limite les possibilités de comparaison directe avec la situation existante.

### 5.3 Discussion et généralisation de la théorie

À la lumière de ces tests, nous pouvons conclure que l'OAD développé, en particulier le MO, présente un réel potentiel pour assister à la planification des tâches de transport de marchandises dans des microcorridors de transport. En effet, les performances se sont révélées satisfaisantes en termes d'optimalité. Les tests réalisés dans le cadre des deux cas d'utilisation ont démontré qu'il est possible d'obtenir des solutions réalisables et d'un degré d'optimalité acceptable, même en limitant le temps de résolution afin de l'adapter aux contraintes opérationnelles.

D'un point de vue managérial, les deux cas d'utilisation mettent en évidence la valeur opérationnelle et stratégique que peut représenter l'intégration d'un OAD fondé sur le modèle d'optimisation proposé. Dans le premier cas, consacré au transport de personnel, l'outil permettrait de réduire significativement le nombre de véhicules nécessaires, contribuant ainsi à une meilleure utilisation des ressources internes et à la réduction des coûts d'exploitation. Dans le second cas, portant sur le transport de marchandises, l'OAD soutient la planification et la coordination des opérations logistiques, en tenant compte des délais de livraison, des capacités de charge et des contraintes temporelles. Ensemble, ces résultats démontrent que l'approche développée ne se limite pas à une optimisation mathématique, mais qu'elle a le potentiel d'offrir un véritable soutien décisionnel aux gestionnaires en leur permettant d'évaluer différents

scénarios, de tester des politiques de transport alternatives et de favoriser une gestion plus durable et intégrée des déplacements internes.

Une observation particulièrement intéressante concerne les performances des instances comprenant des items de petite taille par rapport aux bins, en comparaison avec celles contenant des items de taille moyenne à grande. En effet, nous nous attendions à ce que les instances avec des items de grande taille soient plus faciles à résoudre, comme ce fut le cas pour les résultats obtenus par Crainic *et al.* (2021) lors des tests de performance de leur modèle de bin packing. À l'inverse, nous anticipions que les instances avec des items de petite taille seraient plus difficiles à résoudre. Cette configuration introduit de la symétrie dans le problème en augmentant les permutations d'items possibles, ce qui nuit généralement à la performance du solveur CPLEX (Crainic *et al.*, 2011).

De leur côté, nous nous attendions à ce que les ensembles composés d'items de taille moyenne à grande, soient plus faciles à résoudre, comme cela avait été observé dans les travaux précédents. En théorie, l'augmentation de la taille des items réduit la symétrie en limitant le nombre de permutations possibles, ce qui est supposé améliorer le temps de résolution. Or, dans notre cas, il semble que la possibilité d'associer une même bin à plusieurs trajets, combinée à l'introduction de contraintes de flot, rende le problème plus complexe à résoudre pour les instances comportant des items de taille moyenne à grande. Cette complexité supplémentaire semble annuler les bénéfices attendus de la réduction de la symétrie généralement liée à la taille des bins par rapport à celle des items, ce qui empêche d'observer un gain de performance.

Une hypothèse pour expliquer l'augmentation du temps de résolution est que la taille supérieure des items nécessite de mobiliser un plus grand nombre de bins parmi ceux disponibles. Dans notre version du problème, chaque bin doit être associé à un trajet pour une période donnée, ce qui introduit une forme supplémentaire de symétrie. En effet, plusieurs combinaisons bin/période/trajet peuvent mener à des solutions quasi équivalentes, augmentant ainsi le nombre de configurations à explorer pour le solveur. En effet, plusieurs combinaisons bin/période/trajet peuvent mener à des solutions quasi équivalentes, augmentant ainsi le nombre de configurations à explorer pour le solveur. De plus, puisque le réseau de transport est modélisé sous forme de graphe cyclique bidirectionnel, il existe plusieurs trajets plus ou moins équivalents permettant de relier un nœud A à un nœud B, ce qui accentue ce phénomène.

Par ailleurs, pour que l'outil puisse être utilisé effectivement par les gestionnaires du parc, il est essentiel que les données nécessaires à la génération des instances de transport soient rigoureusement collectées et structurées. L'outil de collecte de données présenté à l'annexe C établit les bases d'une base de données relationnelle qui permettrait de recueillir et de gérer adéquatement ces informations. Une version plus avancée de l'OAD devrait intégrer directement ce type de base de données afin de simplifier son utilisation et d'automatiser davantage les processus de planification. Elle devrait également troquer l'interface via le terminal et la ligne de commande pour une interface utilisateur plus interactive et conviviale.

Ensuite, malgré son potentiel, l'OAD pourrait être difficile à déployer sur le terrain, principalement en raison de sa dépendance au solveur CPLEX. Ce dernier a été choisi pour sa puissance, sa robustesse et sa facilité d'utilisation et car il s'agit également du solveur utilisé par Crainic *et al.* (2021) pour tester les performances du modèle de bin packing qui a inspiré notre propre MO. CPLEX nous semblait donc le choix le plus pertinent pour répondre à notre question et objectif de recherche, qui était de concevoir, développer et tester un modèle d'optimisation intégré à un OAD, permettant d'analyser différents scénarios logistiques afin d'améliorer l'organisation des transports dans le parc Jean-Drapeau.

Ce choix nous a permis d'évaluer de manière fiable les performances et la pertinence du modèle développé. Cependant, CPLEX nécessite une licence pour un usage commercial, dont le coût pourrait ne pas être justifiable pour la Société du parc Jean-Drapeau.

Pour pallier ce problème, il serait possible et facile d'adapter l'OAD pour qu'il utilise un solveur gratuit et open source, comme CBC (<https://github.com/coin-or/Cbc>). Cependant, ce changement pourrait entraîner une diminution notable des performances.

Enfin, le MO développé ne se limite pas aux cas du parc Jean-Drapeau. Il est transposable à d'autres contextes logistiques impliquant des microcorridors. Les échanges avec les employés de la SPJD et les histoires utilisateur ont permis d'identifier d'autres contextes pertinents pour tester le modèle et démontrer son applicabilité, son efficacité ainsi que son potentiel d'adoption.

Notre MO peut être appliqué à tout problème de transport structuré autour d'un microcorridor, c'est-à-dire un réseau de transport pouvant être modélisé sous forme de graphe cyclique ou linéaire bidirectionnel, découpé en périodes de durée égale, dans lequel des items sont déplacés à l'aide de bins. La nature des items et des bins peut être très variée, allant d'individus à des contenants de chlore, ce qui démontre la flexibilité et la robustesse du modèle développé.

Le concept de microcorridor de transport, bien qu'inspiré du contexte spécifique du parc Jean-Drapeau, a ainsi été généralisé à travers notre modélisation et nos tests. Il peut désormais être appliqué à d'autres contextes présentant des caractéristiques similaires.

## CONCLUSION

Ce projet de recherche avait pour objectif de concevoir, développer et mettre à l'épreuve un OAD intégrant un MO, afin de soutenir les équipes de la SPJD dans la planification de leurs opérations de transport de marchandises. À la suite de la mise en lumière des défis logistiques posés par les contraintes du réseau routier insulaire, notamment les accès limités au site et la nécessité de mettre en place des stratégies logistiques plus durables, nous avons proposé d'adapter le problème du bin-packing au contexte des microcorridors de transport en milieu urbain, en nous inspirant des travaux de Crainic *et al.* (2021).

L'objectif principal de la recherche a été atteint. Pour y parvenir, nous avons mobilisé une méthodologie fondée sur la RSD et sur la RO orientée design, ce qui nous a permis de concevoir deux artéfacts : le MO et l'OAD. L'OAD, basé sur le MO, a été formellement défini, implémenté, testé à l'aide d'instances synthétiques, puis validé à partir de cas réalistes inspirés du terrain. Il permet de générer des solutions optimales, respectant les contraintes logistiques propres au site, tout en favorisant une meilleure allocation des ressources et une réduction du nombre de trajets nécessaires. Le MO a démontré sa capacité à produire systématiquement des solutions réalisables, de bonne qualité, même dans des délais de calcul restreints, compatibles avec les contraintes du terrain.

Les tests réalisés dans deux cas d'utilisation distincts, le transport de personnel et le transport de marchandises ont permis de valider la pertinence de l'approche pour les utilisateurs finaux, tout en identifiant des pistes d'amélioration pour une version future de l'OAD, notamment en ce qui concerne l'interface utilisateur, l'intégration d'une base de données, ainsi que le remplacement éventuel de CPLEX par un solveur libre tel que CBC.

Les objectifs spécifiques ont également été remplis : (1) la problématique a été formulée à l'aide d'outils d'analyse (QQOQCCP, diagramme d'Ishikawa) ; (2) un modèle mathématique a été développé en s'appuyant sur une représentation en graphe des trajets possibles dans le parc ; (3) une structure de données compatible avec le modèle a été conçue et validée ; (4) le modèle a été mis en œuvre dans un environnement Python avec le solveur CPLEX ; (5) l'outil a été évalué auprès d'utilisateurs potentiels à travers deux scénarios concrets. Ces étapes ont permis de démontrer la faisabilité et la pertinence de l'approche retenue.

Certaines limites doivent toutefois être reconnues. Premièrement, la collecte de données sur le terrain s'est révélée difficile, ce qui a limité la production d'instances pleinement représentatives des conditions



opérationnelles. Cette contrainte d'accessibilité aux données constitue un enjeu important pour la validation empirique du modèle. Ensuite, l'outil repose actuellement sur un solveur exact (CPLEX), ce qui restreint son utilisation à des problèmes de taille modeste, comme celui du parc Jean-Drapeau. De plus, il fonctionne en ligne de commande, ce qui limite son accessibilité pour des utilisateurs non techniques. Le développement d'une interface conviviale permettrait une meilleure appropriation par les parties prenantes. Par ailleurs, le modèle demeure déterministe et ne prend pas en compte la variabilité de certains paramètres tels que les retards, les imprévus ou les événements climatiques. Finalement, bien qu'il soit théoriquement possible de résoudre des problèmes de transport dans un réseau comportant plus de trois nœuds, tous les tests ont été réalisés sur des cas à trois nœuds. Il serait donc pertinent de tester les performances du modèle dans des réseaux plus complexes afin d'évaluer son applicabilité dans ce type de contexte.

Plusieurs pistes d'amélioration sont également envisagées. Une version du modèle intégrant des heuristiques de résolution permettrait de résoudre plus rapidement des instances plus complexes. Une interface interactive faciliterait la manipulation de scénarios par les équipes opérationnelles.

Sur le plan théorique, cette recherche contribue à la formalisation du concept de microcorridor de transport dans la littérature scientifique. À notre connaissance, il s'agit d'une première tentative de formalisation et de modélisation dédiée à ce type de problématique. Cette contribution pourrait inspirer d'autres recherches en logistique urbaine appliquée à ce type d'espace urbain.

Par ailleurs, les histoires utilisateur ont été mobilisées dans la phase d'évaluation de l'outil d'aide à la décision. Cette approche, encore peu exploitée dans les recherches en logistique urbaine, a permis de tester l'artéfact dans des contextes opérationnels simulés, en s'appuyant sur des scénarios représentatifs des besoins réels des usagers.

En conclusion, ce projet s'enracine dans une problématique réelle, à fort impact opérationnel et théorique. L'OAD développé ouvre la voie à une meilleure gestion des flux dans des contextes urbains atypiques comme celui du parc Jean-Drapeau. Il pourrait être adapté à d'autres environnements urbains similaires, tels que des quartiers centraux ou des zones à accès restreint, contribuant ainsi à une logistique plus durable, mieux planifiée et plus résiliente.

**ANNEXE A**  
**LISTE DES CAUSES**

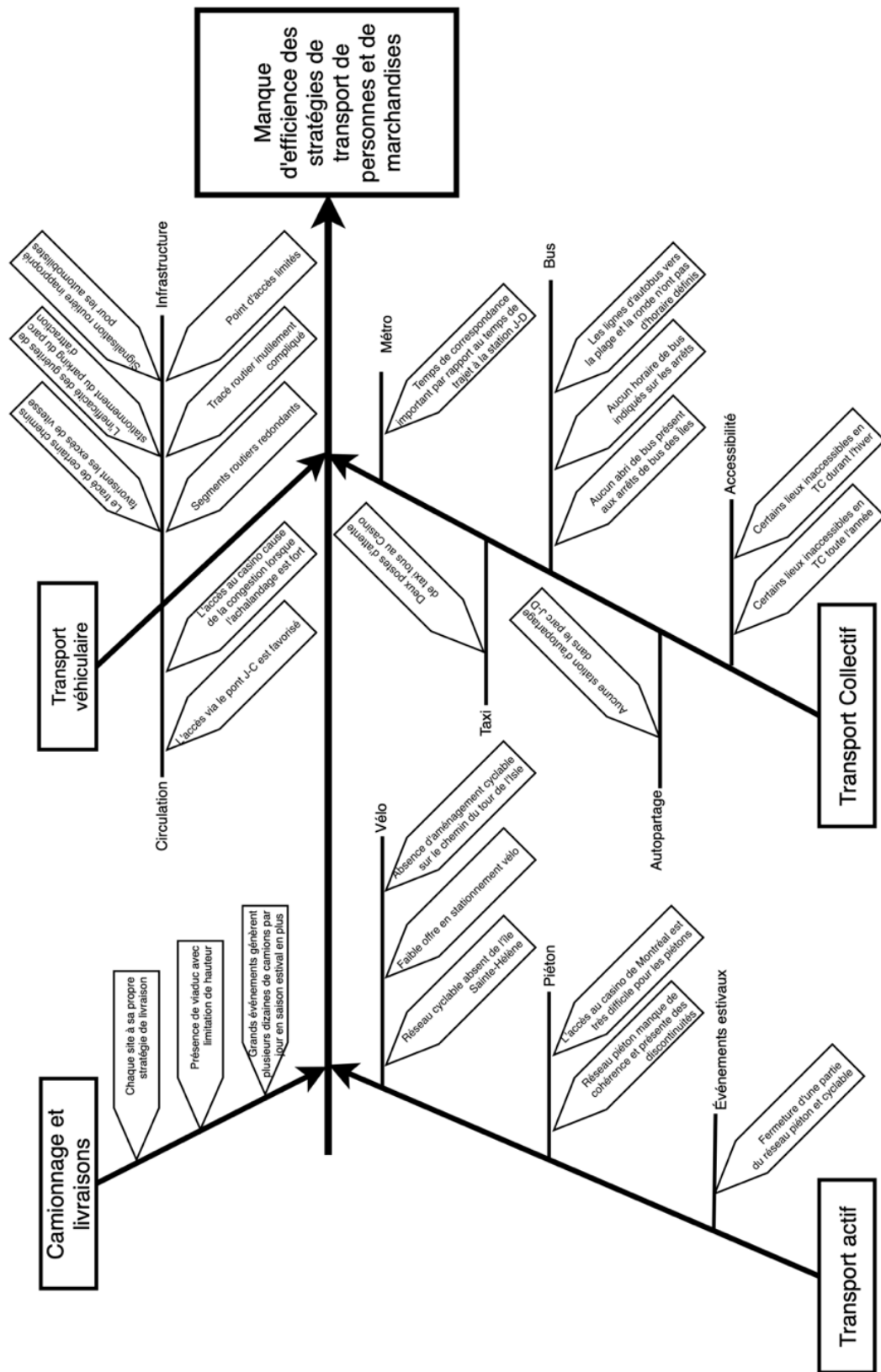
Le tableau complet des causes des problèmes de circulation au Parc Jean-Drapeau

| <b>Causes</b>                                                                                                                                                                                               | <b>Famille</b>           | <b>Sous-Famille</b> |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------|---------------------|
| Grands événements génèrent plusieurs dizaines de camions par jour en saison estivale en plus des livraisons régulières                                                                                      | Camionnage et livraisons |                     |
| Chaque site à sa propre stratégie de livraison                                                                                                                                                              | Camionnage et livraisons |                     |
| Présence de viaduc avec limitation de hauteur                                                                                                                                                               | Camionnage et livraisons |                     |
| Point d'accès limités sur l'île                                                                                                                                                                             | Transport véhiculaire    | Infrastructure      |
| Segments routiers redondants                                                                                                                                                                                | Transport véhiculaire    | Infrastructure      |
| Tracé routier inutilement compliqué à proximité des boucles du pont J-C                                                                                                                                     | Transport véhiculaire    | Infrastructure      |
| Certains chemins sont à sens unique, long, rectiligne et comportent peu de traversées piétonnes, ce qui peut favoriser les excès de vitesse                                                                 | Transport véhiculaire    | Infrastructure      |
| L'inefficacité des guérites de stationnement du stationnement du parc d'attractions cause un refoulement de voiture sur le chemin du Tour de l'Isle pouvant nuire à l'accessibilité des services d'urgence. | Transport véhiculaire    | Infrastructure      |
| Signalisation routière inappropriée pour les automobilistes (police trop de petite et signalisation manquante à proximité de certains embranchements)                                                       | Transport véhiculaire    | Infrastructure      |
| L'accès au casino cause de la congestion au niveau du pont de la Concorde et du chemin Macdonald lors des périodes de fort achalandage (samedi soir)                                                        | Transport véhiculaire    | Circulation         |

|                                                                                                                                 |                       |                     |
|---------------------------------------------------------------------------------------------------------------------------------|-----------------------|---------------------|
| Les utilisateurs favorisent l'accès au parc J-D via le pont J-C                                                                 | Transport véhiculaire | Circulation         |
| En semaine le casino génère la majorité des déplacements en provenance des deux points d'accès principaux                       | Transport véhiculaire | Circulation         |
| Réseau cyclable absent de l'île Sainte-Hélène                                                                                   | Transport actif       | Vélo                |
| Réseau piéton permet de se rendre partout, mais manque de cohérence et présente des discontinuités                              | Transport actif       | Piéton              |
| L'accès au casino de Montréal est très difficile pour les piétons                                                               | Transport actif       | Piéton              |
| Les événements estivaux entraînent la fermeture d'une partie du réseau piéton et cyclable                                       | Transport actif       | Événements estivaux |
| Faible offre en stationnement vélo                                                                                              | Transport actif       | Vélo                |
| Absence d'aménagement cyclable sur le chemin du tour de l'Isle                                                                  | Transport Actifs      | Vélo                |
| Aucune station d'autopartage dans le parc J-D                                                                                   | Transport collectif   | Autopartage         |
| Deux postes d'attente de taxi tous au Casino (36 places au total)                                                               | Transport collectif   | Taxi                |
| Temps de correspondance important par rapport au temps de trajet à la station J-D                                               | Transport collectif   | Métro               |
| Les secteurs du Pavillon des services, du Bassin olympique, des Paddocks et des espaces locatifs ne sont pas accessibles en TC. | Transport collectif   | Accessibilité       |
| Aucun abri de bus présent aux arrêts de bus des Îles                                                                            | Transport collectif   | Bus                 |
| Aucun horaire de bus indiqués sur les arrêts                                                                                    | Transport collectif   | Bus                 |
| Les lignes d'autobus vers la plage et la ronde n'ont pas d'horaire défini                                                       | Transport collectif   | Bus                 |
| Les secteurs de la Ronde et du musée Stewart, ainsi que le secteur du pavillon du Canada inaccessible en TC durant l'hiver      | Transport collectif   | Accessibilité       |

**ANNEXE B**  
**DIAGRAMME D'ISHIKAWA**

La figure du diagramme d'Ishikawa est disponible à la page suivante au format paysage.



## **ANNEXE C**

### **QQOQCCP**

Quoi?

#### **Que fait-on?**

Le parc Jean-Drapeau est un parc Montréalais populaire à vocation récréative. Étant situé au milieu du fleuve Saint-Laurent, il n'est accessible que par deux ponts, par métro, et occasionnellement par une navette maritime.

#### **Quels sont le problème et ses symptômes?**

Le transport de personnes et de marchandises sur le site du parc Jean-Drapeau manque d'efficacité. Les problèmes de circulations touchent les mêmes zones de manières ponctuelles.

#### **Quelles sont ses conséquences ?**

Déplacements sous-optimaux, plus longs, plus coûteux. Le site est difficilement accessible pour certaines personnes. Augmentation des risques d'accident et de la pollution. Perte d'attrait pour le parc.

#### **Que voulons-nous faire ?**

Trouver de nouvelles stratégies efficaces (à la fois écoresponsables, sécuritaires et rapides) pour assurer le transport des personnes et des marchandises.

Qui?

#### **Qui le fait ?**

Les situations problématiques sont causées par les utilisateurs ainsi que par les gestionnaires du réseau.

### **Qui sont les acteurs et les responsables (personnes et organisations) ?**

La Ville de Montréal, le Casino de Montréal, la Ronde, les groupes organisant les événements saisonniers ainsi que tous les gens se déplaçant sur le site du parc et ses alentours.

### **Qui est touché par le problème ?**

Les usagers du Parc, ainsi que tous les gens qui y transitent.

Où?

### **Où le fait-on ?**

Le parc-Jean drapeau est composé de deux îles. Sur celles-ci, quatre lieux principaux peuvent être identifiés : le pont de la Concorde, le pont Jacques-Cartier, le casino et le site du parc Jean-drapeau. La figure 1 illustre la manière dont ces sites communiquent entre eux. Il est également important de souligner que le pont Jacques-Cartier ainsi que le pont de la Concorde sont les deux seules portes d'accès aux sites du Parc Jean-Drapeau et du Casino (en excluant la voie maritime) à partir de l'extérieur des deux îles.

### **À quel endroit se produit le problème ? Quelles sont les distances en cause ?**

Les situations problématiques se produisent principalement aux quatre sites identifiés précédemment (sur la figure 1, les nœuds), ainsi que sur les voies les interreliant (sur la figure 1, les arcs). Les problèmes d'accessibilité en transport collectif quant à eux touchent plus particulièrement les sites du parc J-D et du casino tant au niveau des trajets à partir de l'extérieur que ceux au sein du parc en lui-même. Les différentes causes du problème sont décrites plus en détail dans le diagramme d'Ichikawa.

Quand?

### **Quand le fait-on ? Quand le problème est-il apparu ? Quelle est sa fréquence, sa durée ?**

Le parc a subi de gros réaménagement pour accueillir l'exposition universelle de Montréal de 1967. Depuis, l'achalandage du site augmente d'année en année, ce qui oblige les différents acteurs responsables de la gestion du site à développer de nouvelles stratégies pour faire face à cette demande toujours croissante. Les problèmes relatifs aux infrastructures sont présents à l'année longue. Les problèmes liés à la circulation et à l'accessibilité se manifestent plus ou moins fortement selon le jour et la saison. Les périodes les plus problématiques correspondent aux forts pics d'achalandages : durant les événements estivaux (Osheaga par exemple) et durant la fin de semaine. Les problèmes liés à l'accessibilité sont surtout présents durant la période hivernale, car l'offre en transport en commun est limitée, ainsi que durant les événements estivaux qui peuvent perturber les services.

Comment?

#### **Comment le fait-on? Sous quelle forme le problème se présente-t-il ?**

Comme mentionné dans la section sur l'élaboration du diagramme de cause et effet, le problème peut être défini selon les quatre familles suivantes : la gestion du camionnage et des livraisons, le transport véhiculaire, les modes de transports actifs et le transport collectif.

Les situations problématiques se présentent entre autres sous la forme d'embouteillage, de file d'attente à l'entrée des stationnements, de problème d'accessibilité à certaines zones du parc, de tracés routiers et signalisation incohérente, etc.

#### **Comment le solutionner ?**

Une modernisation des infrastructures (les routes et les sentiers par exemple), que ce soit pour les transports actifs et les véhicules ainsi qu'une bonification de l'offre en transport collectif pourrait constituer une bonne base à la résolution du problème. Il est également important d'aborder la question de la logistique, tant au niveau du transport de marchandises que celui des personnes, principalement à cause des limitations physiques du réseau routier. Ce point sera élaboré à la prochaine section.



Combien?

**Quelles sont les unités de mesure ? Combien cela coûte-t-il ? Quels sont les écarts par rapport aux objectifs ?**

Pour l'instant le « combien » n'est pas explicité dans la documentation auquel nous avons accès. Il est donc impossible de répondre à ces questions avec certitude. Cependant, il est clair que la problématique entraîne des pertes d'ordre temporel et monétaire chez les différents acteurs concernés.

## ANNEXE D

### DESCRIPTION DE L'OUTIL DE COLLECTE DE DONNÉES

Afin d'alimenter notre modèle d'optimisation, nous ne procéderons pas nous-mêmes à la collecte de données. Nous définirons plutôt le format des données nécessaires à la génération des instances du problème, puis nous transmettrons cette information aux services informatiques du parc. Notre objectif est d'accéder aux données pertinentes déjà collectées, tout en limitant les données manquantes ou superflues.

Pour faciliter le transfert d'information, nous avons conçu un classeur Excel qui permet de représenter les différents champs de données pertinents par entités, selon la logique d'une base de données relationnelle. Dans le cadre du présent problème, les données ont été regroupées en cinq tables : la table VÉHICULE, la table ITEM, la table ITEM-ITEM, la table LIEU et la table LIVRAISON.

#### **La table VÉHICULE**

La table VÉHICULE (Tableau D-1) regroupe les différentes caractéristiques des véhicules utilisés pour effectuer les livraisons. Les champs de cette table correspondent aux variables de décision ainsi qu'aux compétences particulières en matière de transport liées aux *bins*, telles que décrites au chapitre I. Le premier champ est celui de l'identifiant du véhicule (*ID*), un identifiant unique de type numérique. Le deuxième champ correspond au nom du véhicule, de type texte. Le troisième champ indique le volume cargo maximum, soit le volume maximal d'items (en  $\text{cm}^3$ ) que le véhicule peut transporter, et il est de type numérique. Le quatrième champ concerne le poids cargo maximum, c'est-à-dire le poids maximal d'items (en kilogrammes) que le véhicule est en mesure de transporter ; il est également de type numérique. Le cinquième champ est celui du coût fixe d'utilisation, représentant le coût initial associé à l'activation du véhicule (en dollars), et est de type numérique. Le sixième champ correspond au coût variable d'utilisation, qui dépend de la distance parcourue (en \$/km), et est aussi de type numérique. Le septième champ est le taux d'émission de  $\text{CO}_2$ , exprimé en  $\text{CO}_2/\text{km}$ , et il est de type numérique. Le huitième champ est un marqueur booléen indiquant si le véhicule est apte à transporter des marchandises réfrigérées. Enfin, le

neuvième champ est un autre marqueur booléen, permettant de déterminer si le véhicule peut transporter des marchandises fragiles.

Tableau D-1 Table VÉHICULE

| VÉHICULE                    |             |                                                   |
|-----------------------------|-------------|---------------------------------------------------|
| Nom champs                  | Type        | Description                                       |
| Véhicule Id                 | Numérique   | Id du véhicule de transport                       |
| Nom véhicule                | Texte court | Le nom du véhicule de transport                   |
| Volume cargo max            | Numérique   | Le volume cargo maximum disponible (cm3)          |
| Poids cargo max             | Numérique   | Le poids cargo maximum disponible (kg)            |
| Coût fixe d'utilisation     | Monétaire   | Coût fixe d'utilisation du véhicule (\$)          |
| Coût variable d'utilisation | Monétaire   | Coût d'utilisation par kilomètre parcouru (\$/km) |
| Émission CO2                | Numérique   | Émission de CO2 par kilomètre (CO2/km)            |
| Réfrigéré                   | Booléen     | Capable de transporter des produits réfrigérés    |
| Fragile                     | Booléen     | Capable de transporter du matériel fragile?       |

### La table ITEM

La table ITEM (voir Tableau D-2) regroupe les différentes caractéristiques des items à livrer. Les champs de cette table correspondent aux variables de décision ainsi qu'aux besoins particuliers en matière de transport relatifs aux items, tels que décrits au chapitre I. Le premier champ est l'identifiant de l'item (ID), un identifiant unique de type numérique. Le deuxième champ correspond au nom de l'item, un champ de

type texte. Le troisième champ indique le volume de l'item, exprimé en cm<sup>2</sup>, et est de type numérique. Le quatrième champ correspond au moment de disponibilité, c'est-à-dire à la période à partir de laquelle l'item peut être pris en charge pour la livraison ; il est de type numérique. Le cinquième champ correspond à la période de livraison requise, soit le moment avant lequel l'item doit impérativement être livré, et il est également de type numérique. Le sixième champ indique le lieu d'origine, soit l'endroit où se trouve initialement l'item ; il est de type *lieu*, tel que défini dans la table LIEU. Enfin, le septième champ correspond à la destination, c'est-à-dire au lieu où l'item doit être livré ; il est lui aussi de type *lieu*, conformément à la table LIEU.

Tableau D-2 Table ITEM

| ITEM                      |             |                                    |
|---------------------------|-------------|------------------------------------|
| Nom champs                | Type        | Description                        |
| Item Id                   | Numérique   | Id de l'item                       |
| Nom Item                  | Texte court | Le nom de l'item                   |
| Volume Item               | Numérique   | Le volume de l'item (cm3)          |
| Moment disponibilité Item | Numérique   | Le moment où l'item est disponible |
| Moment livraison Item     | Numérique   | Le moment où l'item a été livré    |
| Lieu d'origine            | Lieu        | Lieu d'origine de l'item           |
| Lieu de livraison         | Lieu        | La destination de l'item           |
| Réfrigéré                 | Booléen     | Besoin d'être réfrigérés?          |
| Fragile                   | Booléen     | Fragile?                           |

### La table LIEU

La table LIEU (Tableau D-3) regroupe les différents emplacements du parc où peuvent s'effectuer les livraisons. Ces lieux correspondent aux nœuds de notre réseau logistique. Le premier champ est l'identifiant du lieu (ID), un identifiant unique de type numérique. Le deuxième champ correspond au nom du lieu, un champ de type texte.

TABLEAU D-3 Table LIEU

| LIEU            |             |                                                                                  |
|-----------------|-------------|----------------------------------------------------------------------------------|
| Nom champs      | Type        | Description                                                                      |
| Destination Id  | Numérique   | Id de la destination                                                             |
| Nom destination | Texte court | Le nom de la destination (La Ronde, le Casino, le parc, les événements spéciaux) |

### La table LIEU-LIEU

La table LIEU-LIEU (Tableau D-4) regroupe les différents chemins directs entre les lieux du parc. Ces associations correspondent aux arcs de notre réseau logistique. Le premier champ est celui de lieu-1, représentant le premier lieu de l'association, tel que défini dans la table LIEU. Le deuxième champ est celui de lieu-2, correspondant au second lieu de l'association, également défini dans la table LIEU. Le troisième champ indique la distance entre les deux lieux, exprimée en unités numériques (par exemple en mètres ou en kilomètres), et est de type numérique.

TABLEAU D.4 Table LIEU-LIEU

| LIEU-LIEU  |      |             |
|------------|------|-------------|
| Nom champs | Type | Description |

|          |           |                               |
|----------|-----------|-------------------------------|
| Lieu_1   | Lieu      | Id de la destination 1        |
| Lieu_2   | Lieu      | Id de la destination 2        |
| Distance | Numérique | Distance entre les deux lieux |

### La Table VÉHICULE-ITEM

La table VÉHICULE-ITEM (Tableau D-5) permet d'associer les items aux véhicules chargés de les transporter. Le premier champ est l'identifiant de l'association véhicule-item, un identifiant unique de type numérique. Le deuxième champ correspond à l'item transporté, tel que défini dans la table ITEM. Le troisième champ indique le véhicule utilisé pour la livraison, tel que défini dans la table VÉHICULE. Le quatrième champ est le coût d'assignation, soit le coût associé à l'attribution d'un item à un véhicule ; il est de type numérique. Enfin, le cinquième champ correspond au moment de la livraison, c'est-à-dire à la période à laquelle l'item a été livré ; il est également de type numérique.

TABLEAU D-5 Table LIVRAISON

| LIVRAISON          |           |                                             |
|--------------------|-----------|---------------------------------------------|
| Nom champs         | Type      | Description                                 |
| Véhicule-Item Id   | Numérique | Id de l'item                                |
| Item               | Item      | Le nom de l'item                            |
| Véhicule           | Véhicule  | Le nom du véhicule de transport             |
| Coût d'assignation | Numérique | Le coût d'assignation de l'item au véhicule |
| Moment_livraison   | Numérique | Moment où l'item a été livré                |

## ANNEXE E

### HISTOIRES UTILISATEUR

#### **Première mise en situation : Soutenir les opérations de maintenance par les agents du parc.**

Chaque jour, diverses opérations de maintenance doivent être effectuées afin d'entretenir le site. Pour chaque opération, un agent utilise le véhicule qui lui est assigné pour transporter le matériel nécessaire à sa tâche. Un gestionnaire de la SPJD souhaite planifier les opérations de maintenance prévues le lendemain pour minimiser les coûts opérationnels. Le lieu d'origine du véhicule, le lieu d'entreposage du matériel et le lieu d'opération correspondent à des nœuds du réseau de transport. Pour ce faire, il utilise l'OAD afin de produire un plan de livraison optimal selon les marchandises à livrer et les véhicules disponibles.

- 1- Créer un fichier .csv listant les caractéristiques des véhicules à transporter pour chacune des tâches.
  - a. Volume maximal (capacité);
  - b. Coût d'opération fixe;
  - c. Fenêtre de disponibilité;
  - d. Nœud de départ du véhicule.
  
- 2- Créer un fichier .csv listant les caractéristiques des marchandises disponibles pour effectuer les tâches
  - a. Volume de l'item;
  - b. Fenêtre de disponibilité;
  - c. Période de livraison due;
  - d. Nœud de départ de l'item;
  - e. Nœud de destination de l'item
  
- 3- Lancer le programme en utilisant les deux fichiers créés aux étapes 1 et 2 comme paramètre.
  
- 4- Utiliser le plan de transport optimal proposé pour planifier les opérations du lendemain. Le plan de transport nous permet de déterminer sur quel chemin et à quelle période chaque véhicule

effectuera son trajet, dans quel véhicule et à quelle période un item sera transporté vers sa destination et quel sera le coût de la solution optimale?

### **Deuxième mise en situation : Assembler une flotte de véhicule.**

La SPJD envisage de moderniser sa flotte de véhicules pour la rendre plus respectueuse de l'environnement et mieux adaptée à ses besoins actuels. Après avoir examiné les options de transport disponibles sur le marché, elle a identifié trois modèles de véhicules susceptibles de répondre à ses exigences. Chacun de ces modèles se distingue par sa capacité de chargement, son coût d'utilisation et sa taille. Dans le cadre de cette initiative, le parc envisage de mettre en place de nouvelles réglementations restreignant les heures de circulation des véhicules en fonction de leur taille. Ainsi, les véhicules de moins de 2 mètres pourront circuler librement toute la journée, ceux mesurant entre 2 et 4,5 mètres seront autorisés à circuler de 5h à 14h, tandis que les véhicules dépassant les 4,5 mètres ne pourront circuler que de 5h à 9h. Un conseiller en mobilité est chargé de déterminer le nombre de véhicules à acquérir afin de constituer une flotte capable de répondre efficacement aux besoins de la société du parc Jean-Drapeau. Pour ce faire, il utilise l'OAD pour simuler l'acheminement des marchandises correspondant à différents volumes prévus en utilisant la potentielle nouvelle flotte. L'objectif est de générer des plans de livraison optimaux pour des jours normaux ainsi que pour des journées de forte affluence, en se basant sur les données réelles de volume de marchandises collectées par le parc. En analysant les plans de livraison optimaux pour les différents scénarios, le gestionnaire peut observer la quantité et le type de véhicules nécessaires pour assurer une livraison optimale des marchandises. Cette analyse lui permettra de prendre une décision plus éclairée.

- 1- En utilisant le même format de données que celui utilisé dans le scénario précédent, générer des fichiers .csv contenant des exemples de marchandises à livrer pour des jours ordinaires et des jours de forte affluence.
- 2- En utilisant le même format de données qu'à la mise en situation précédente, créer un fichier .csv listant les caractéristiques des véhicules pouvant potentiellement constituer la flotte.



- 3- Pour chaque fichier de marchandises créé à l'étape 1, lancer le programme en utilisant également le fichier de véhicules créé à l'étape 2 comme paramètre.
- 4- Analyser les plans de transport optimaux pour chaque scénario afin d'observer la quantité et le type de véhicules nécessaires pour assurer une livraison optimale des marchandises.

Troisième mise en situation : Décider de l'emplacement des points de livraison

La SPJD envisage d'installer des points de livraison pour consolider les déplacements de marchandises entre ses différents sites internes. Elle examine actuellement les emplacements les plus appropriés pour ces points afin de minimiser les coûts de livraison entre les sites. Six sites potentiels ont été identifiés, et il est nécessaire de sélectionner les trois meilleurs parmi eux. Pour chaque combinaison de sites envisagée, le gestionnaire génère des scénarios de marchandises à livrer utilisant la flotte de véhicules du parc, en adaptant les coûts de livraison à cette nouvelle configuration, et il utilise l'OAD pour produire un plan de livraison optimal. En comparant les différents plans de livraison, il est en mesure d'identifier la combinaison de site la plus adéquate.

- 1- En utilisant le même format de données que celui utilisé dans les scénarios précédents, créer un fichier .csv listant les caractéristiques des véhicules à transporter pour chacune des tâches.
- 2- En utilisant le même format de données que celui utilisé dans le scénario précédent, générer des fichiers .csv contenant des exemples de marchandises à livrer pour des jours ordinaires et des jours de forte affluence.
- 3- Pour chaque fichier de marchandises créé à l'étape 2, lancer le programme en utilisant également le fichier de véhicules créé à l'étape 1 comme paramètre.
- 4- Analyser les plans de transport optimaux pour chaque scénario afin d'identifier la combinaison de site la plus adéquate. 0000-00-00 00:00:00

## BIBLIOGRAPHIE

- Acquier, A., Carbone, V. et Massé, D. (2019). How to Create Value(s) in the Sharing Economy: Business Models, Scalability, and Sustainability. *IDEAS Working Paper Series from RePEc*, 2223055032. <https://www.proquest.com/working-papers/how-create-value-s-sharing-economy-business/docview/2223055032/se-2>
- Acquier, A., Daudigeos, T. et Pinkse, J. (2017). Promises and Paradoxes of the Sharing Economy: An Organizing Framework. *Technological Forecasting and Social Change*, 125, 1-10. <https://doi.org/10.1016/j.techfore.2017.07.006>
- AFP. (2010, 1<sup>er</sup> février). *E-Commerce: +22% En Europe En 2009*. LEFIGARO § Flash Actu. <https://www.lefigaro.fr/flash-actu/2010/02/01/01011-20100201FILWWW00015-e-commerce-22-en-europe-en-2009.php>
- Boysen, N., Fedtke, S. et Schwerdfeger, S. (2021). Last-Mile Delivery Concepts: A Survey from an Operational Research Perspective. *OR Spectrum*, 43(1), 1-58. <https://doi.org/10.1007/s00291-020-00607-8>
- Braunstein, D. N., Lauer, T. W. et Doane, D. P. (1991). Information requirements prototyping for decision support systems. *Journal of Information Technology*, 6(1), 23-26.
- Bruni, M.E., Crainic, T.G., Perboli, G. (2024). Bin Packing Methodologies for Capacity Planning in Freight Transportation and Logistics. In: Crainic, T.G., Gendreau, M., Frangioni, A. (eds) *Combinatorial Optimization and Applications*. International Series in Operations Research & Management Science, vol 358. Springer, Cham. [https://doi.org/10.1007/978-3-031-57603-4\\_6](https://doi.org/10.1007/978-3-031-57603-4_6)
- Cohn, M. (2004). *User stories applied: For agile software development*. Addison-Wesley.
- Crainic, T. G., Djeumou Fomeni, F. et Rei, W. (2021). Multi-Period Bin Packing Model and Effective Constructive Heuristics for Corridor-Based Logistics Capacity Planning. *Computers & Operations Research*, 132, 105308. <https://doi.org/10.1016/j.cor.2021.105308>
- Crainic, T. G., Gendreau, M. et Potvin, J.-Y. (2009). Intelligent Freight-Transportation Systems: Assessment and the Contribution of Operations Research. *Transportation Research Part C: Emerging Technologies*, 17(6), 541-557. <https://doi.org/10.1016/j.trc.2008.07.002>
- Crainic, T. G., Perboli, G., Rei, W. et Tadei, R. (2011). Efficient lower bounds and heuristics for the variable cost and size bin packing problem. *Computers & Operations Research*, 38(11), 1474-1482. <https://doi.org/10.1016/j.cor.2011.01.001>
- Demir, E., Huang, Y., Scholts, S. et Van Woensel, T. (2015). A Selected Review on the Negative Externalities of the Freight Transportation: Modeling and Pricing. *Transportation Research Part E: Logistics and Transportation Review*, 77, 95-114. <https://doi.org/10.1016/j.tre.2015.02.020>

- Farahani, A., Genga, L., Schrottenboer, A. H., & Dijkman, R. M. (2024). Capacity planning in logistics corridors: Deep reinforcement learning for the dynamic stochastic temporal bin packing problem. *Transportation Research Part E: Logistics and Transportation Review*, 191, 103742. <https://doi.org/10.1016/j.tre.2024.103742>
- Goldkuhl, G. (2012). Design Research in Search for a Paradigm: Pragmatism Is the Answer. Dans M. Helfert et B. Donnellan (dir.), *Practical Aspects of Design Science* (vol. 286, p. 84-95). Springer Berlin Heidelberg. [https://doi.org/10.1007/978-3-642-33681-2\\_8](https://doi.org/10.1007/978-3-642-33681-2_8)
- Hevner, A. R. et Gregor, S. (2013). Positioning and Presenting Design Science Research for Maximum Impact. *MIS Quarterly*, 37(2), 337-355. <https://doi.org/10.25300/MISQ/2013/37.2.01>
- Hevner, A. R., March, S. T., Park, J. et Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, 28(1), 75-105.
- Hughes-Cromwick, E. et Cregger, J. (2013). *Financing the Infrastructure to Support Alternative Fuel Vehicles: How Much Investment is Needed and How Will It Be Funded*. Cargroup.
- Johannesson, P. et Perjons, E. (2014). Explicate Problem. Dans P. Johannesson et E. Perjons, *An Introduction to Design Science* (p. 91-102). Springer International Publishing. [https://doi.org/10.1007/978-3-319-10632-8\\_5](https://doi.org/10.1007/978-3-319-10632-8_5)
- Kervall, M. et Pålsson, H. (2022). Barriers to Change in Urban Freight Systems: A Systematic Literature Review. *European Transport Research Review*, 14(1), 29. <https://doi.org/10.1186/s12544-022-00553-2>
- Laine-Devroede, N. (2022, 19 mai). Bilan E-Commerce Au 1er Trimestre 2022: Un Chiffre D'affaires En Hausse De 12%. *Fevad, la Fédération du e-commerce et de la vente à distance*. <https://www.fevad.com/bilan-e-commerce-au-1er-trimestre-2022-un-chiffre-daffaires-en-hausse-de-12/>
- Leavitt, H. J. et Whisler, T. L. (1953). MANAGEMENT in the 1980's. *Harvard Business Review*, 36(6), 41-48.
- Llorca, C. et Moeckel, R. (2021). Assesment of the Potential of Cargo Bikes and Electrification for Last-Mile Parcel Delivery by Means of Simulation of Urban Freight Flows. *European Transport Research Review*, 13(1), 33. <https://doi.org/10.1186/s12544-021-00491-5>
- March, S. T. et Smith, G. F. (1995). Design and Natural Science Research on Information Technology. *Decision Support Systems*, 15(4), 251-266. [https://doi.org/10.1016/0167-9236\(94\)00041-2](https://doi.org/10.1016/0167-9236(94)00041-2)
- Marrekchi, E., Besbes, W., Dhouib, D. et Demir, E. (2021). A Review of Recent Advances in the Operations Research Literature on the Green Routing Problem and Its Variants. *Annals of Operations Research*, 304(1-2), 529-574. <https://doi.org/10.1007/s10479-021-04046-8>
- Mayer, Robert, Ouellet, Francine, Saint-Jacques, Marie-Christine et Turcotte, Daniel. (2000). L'entrevue de recherche. Dans *Méthodes de recherche en intervention sociale* (p. 115-133). Gaëtan Morin.

- Moncef, B. et Monnet Dupuy, M. (2021). Last-Mile Logistics in the Sharing Economy: Sustainability Paradoxes. *International Journal of Physical Distribution & Logistics Management*, 51(5), 508-527. <https://doi.org/10.1108/IJPDLM-10-2019-0328>
- Monks, T. (2015). Operational Research as Implementation Science: Definitions, Challenges and Research Priorities. *Implementation Science*, 11(1), 81. <https://doi.org/10.1186/s13012-016-0444-0>
- Montecinos, J., Ouhimmou, M., Chauhan, S., Paquet, M. et Gharbi, A. (2021). Transport Carriers' Cooperation on the Last-Mile Delivery in Urban Areas. *Transportation*, 48(5), 2401-2431. <https://doi.org/10.1007/s11116-020-10134-8>
- Muratovski, G. et Vogel, C. (dir.). (2019). *Philosophical Frameworks and Design Processes: Re:Research, Volume 2*. Intellect Books. <https://doi.org/10.2307/j.ctv36xw19h>
- Norbert, R., Ouellet, F. et Parent, G. (2009). *Méthodes d'optimisation pour la gestion* Chenelière Éducation.
- Office québécois de la langue française. (2023). *corridor de transport*. <https://vitrlinguistique.oqlf.gouv.qc.ca/fiche-gdt/fiche/26561119/corridor-de-transport>
- O'Keefe, R. (2014). Design Science, the Design of Systems and Operational Research: Back to the Future? *Journal of the Operational Research Society*, 65(5), 673-684. <https://doi.org/10.1057/jors.2012.175>
- Oliveira, L. K. de, Oliveira, I. K. de, França, J. G. da C. B., Balieiro, G. W. N., Cardoso, J. F., Bogo, T., Bogo, D. et Littig, M. A. (2022). Integrating Freight and Public Transport Terminals Infrastructure by Locating Lockers: Analysing a Feasible Solution for a Medium-Sized Brazilian Cities. *Sustainability*, 14(17), 10853. <https://doi.org/10.3390/su141710853>
- Parc Jean-Drapeau. (2022, 23 octobre). Dans *Wikipédia*. Récupéré le 5 novembre 2022 de [https://fr.wikipedia.org/w/index.php?title=Parc\\_Jean-Drapeau&oldid=198032094](https://fr.wikipedia.org/w/index.php?title=Parc_Jean-Drapeau&oldid=198032094)
- Peffer, K., Tuunanen, T., Rothenberger, M. A. et Chatterjee, S. (2008). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, 24(3), 45-77.
- Perboli, G., Brotcorne, L., Bruni, M. E. et Rosano, M. (2021). A new model for Last-Mile Delivery and Satellite Depots management: The impact of the on-demand economy. *Transportation Research Part E: Logistics and Transportation Review*, 145, 102184. <https://doi.org/10.1016/j.tre.2020.102184>
- Quak, H., Nesterova, N., van Rooijen, T. et Dong, Y. (2016). Zero Emission City Logistics: Current Practices in Freight Electromobility and Feasibility in the Near Future. *Transportation Research Procedia*, 14, 1506-1515. <https://doi.org/10.1016/j.trpro.2016.05.115>
- Savelsbergh, M. et Van Woensel, T. (2016). 50th Anniversary Invited Article—City Logistics: Challenges and Opportunities. *Transportation Science*, 50(2), 579-590. <https://doi.org/10.1287/trsc.2016.0675>
- Simon, Herbert Alexander. (2004). *Les sciences de l'artificiel*. Gallimard.

- Taniguchi, E. (2014). Concepts of City Logistics for Sustainable and Liveable Cities. *Procedia - Social and Behavioral Sciences*, 151, 310-317. <https://doi.org/10.1016/j.sbspro.2014.10.029>
- Venable, J., Pries-Heje, J. et Baskerville, R. (2016). FEDS: a Framework for Evaluation in Design Science Research. *European Journal of Information Systems*, 25(1), 77-89. <https://doi.org/10.1057/ejis.2014.36>
- Wikipédia. (2022, 23 octobre). *Parc Jean-Drapeau*. Wikipédia. [https://fr.wikipedia.org/w/index.php?title=Parc\\_Jean-Drapeau&oldid=198032094](https://fr.wikipedia.org/w/index.php?title=Parc_Jean-Drapeau&oldid=198032094)
- Working group of the Scientific Advisory Board of German Logistics Association (BVL), Delfmann, W., Dangelmaier, W., Günthner, W., Klaus, P., Overmeyer, L., Rothengatter, W., Weber, J. et Zentes, J. (2010). Towards a Science of Logistics: Cornerstones of a Framework of Understanding of Logistics as an Academic Discipline. *Logistics Research*, 2(2), 57-63. <https://doi.org/10.1007/s12159-010-0034-5>