

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

**Problèmes de tournées en viabilité hivernale utilisant la prévision des volumes
d'épandage**

CHAHID AHABCHANE

Département de mathématiques et de génie industriel

Thèse présentée en vue de l'obtention du diplôme de *Philosophiæ Doctor*

Génie industriel

Avril 2020

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

Cette thèse intitulée :

**Problèmes de tournées en viabilité hivernale utilisant la prévision des volumes
d'épandage**

présentée par **Chahid AHABCHANE**

en vue de l'obtention du diplôme de *Philosophiæ Doctor*

a été dûment acceptée par le jury d'examen constitué de :

Louis-Martin ROUSSEAU, Ph. D., président

Martin TRÉPANIER, Ph. D., membre et directeur de recherche

André LANGEVIN, Ph. D., membre et codirecteur de recherche

Issmaïl EL HALLAOUI, Ph. D., membre

Jean-François CÔTÉ, Ph. D., membre externe

DÉDICACE

À mes parents, et à Dina.

REMERCIEMENTS

Je remercie très sincèrement mes directeurs de recherche André Langevin et Martin Trépanier pour leur soutien, leur disponibilité régulière, et leurs conseils. Et le plus important leur gentillesse. Honnêtement, ce fut un très grand plaisir de travailler avec vous.

J'aimerais aussi remercier les membres de mon jury pour avoir accepté d'évaluer mes travaux de thèse : Louis-Martin Rousseau, Issmail Elhallaoui et Jean-François Côté.

Un grand merci également au personnel de MAGI, SEP et AÉCSP pour leur support et leur sympathie, aussi à tous mes amis de Polytechnique, votre présence a rendu mon travail plus agréable.

J'exprime ma gratitude à tous les professeurs qui ont assuré ma formation à Polytechnique : Bernard Clément, Jean-Marc Frayret, Diane Riopel, Louis-Martin Rousseau, Charles Audet, Alain Hertz, Michel Gendreau, Vincent Pascal et Issmail Elhallaoui.

Merci aux services des travaux publics de la ville de Granby et de la ville de Saint-Jean-Sur-Richelieu pour les données et les informations qu'ils nous ont fournies à propos de leurs pratiques.

Je tiens à remercier mes parents, mes beaux-parents, et également mes frères.

Finalement, je tiens à adresser un remerciement spécial à mon épouse Dina pour son soutien, son encouragement, son affection et son amour, surtout que nous avons commencé cette aventure ensemble.

RÉSUMÉ

Cette thèse combine deux domaines de recherche différents appliqués au déneigement : la recherche opérationnelle et la science des données. La science des données a été utilisée pour développer un modèle de prédiction de quantité de sel et d'abrasif avec une méthodologie d'apprentissage machine; par la suite, ce modèle est pris en compte pour la confection des tournées de véhicules. La confection des tournées a été élaborée en utilisant des outils de la recherche opérationnelle, qui servent à optimiser les tournées en considérant plusieurs contraintes et en intégrant les données réelles.

La thèse est le fruit d'une collaboration avec deux villes québécoises, Granby et Saint-Jean-sur-Richelieu. Elle traite une application réelle en viabilité hivernale, qui est l'opération d'épandage. Cette opération est une activité nécessaire, dont le but est d'assurer une meilleure circulation routière. Cependant, cela se réalise avec un coût économique et environnemental important. Par conséquent, la réduction de ce coût devient une grande préoccupation. Cette thèse contribue significativement aux opérations d'épandage : premièrement, nous prédisons la quantité nécessaire de sel et d'abrasif à épandre afin d'éviter le surépandage; deuxièmement, nous optimisons les tournées des opérations d'épandage en considérant la variation de la quantité.

La première contribution de cette thèse consiste en un modèle de prédiction des quantités de sel et d'abrasif pour chaque segment de rue et pour chaque heure, en utilisant des algorithmes d'apprentissage machine. L'importance de cette contribution réside d'une part dans l'intégration des données géomatiques avec les données météo-routières, et d'autre part dans l'extraction des variables importantes (*feature engineering*) pour le modèle de prédiction. Plusieurs algorithmes d'apprentissage machine ont été évalués : (les forêts aléatoires, les arbres extrêmement aléatoires, les réseaux de neurones artificiels, *Adaboost*, *Gradient Boosting Machine* et *XGBoost*). Le modèle élaboré par *XGBoost* a réalisé une meilleure performance. Le modèle de prédiction permet non seulement de prédire les quantités de sel et d'abrasif nécessaires à épandre mais aussi, d'identifier les variables les plus importantes pour la prédiction. Cette information représente un outil de décision intéressant pour les gestionnaires.

L'identification des variables importantes pourrait améliorer les opérations de déneigement. D'après les résultats trouvés, le facteur humain (conducteur) influence significativement la quantité d'épandage; donc, le contrôle de ce facteur peut améliorer considérablement ces opérations.

La deuxième contribution introduit un nouveau problème dans la littérature : le problème de tournées de véhicules générales avec capacité dont la quantité de sel et d'abrasif dépend du temps. Le problème est basé sur l'hypothèse que le modèle de prédiction est capable de fournir la quantité d'épandage pour chaque segment et pour chaque heure avec une bonne précision. Le fait d'avoir cette information pour chaque heure et pour chaque segment de rue, introduit la notion du temps dépendant. Le nouveau problème est modélisé à l'aide d'une formulation mathématique sur le graphe original, ce qui présente un défi de modélisation. En effet, il est difficile d'associer des temps de début et de fin uniques à un arc ou à une arête. Une métaheuristique basée sur la stratégie de destruction et construction a été développée pour résoudre les grandes instances. La métaheuristique est inspirée de SISRs (*Slack Induction by String Removals*). Elle considère la demande dépendante du temps et la présence des arêtes par la méthode d'évaluation basée sur la programmation dynamique. De nouvelles instances ont été créées à partir des instances des problèmes de tournées de véhicules générales avec contrainte de capacité avec demande fixe. Elles ont été générées à partir de différents types de fonction dont la demande dépend du temps.

La troisième contribution propose une nouvelle approche, dans le but de présenter le niveau de priorité des rues (la hiérarchie de service) sous forme d'une fonction linéaire dépendante du temps. Le problème présenté dans cette contribution concerne des tournées de véhicules générales hiérarchiques avec contrainte de capacité sous l'incertitude de la demande. Lorsque les données collectées ne permettent pas de développer un bon modèle de prédiction, la notion de demande dépendante du temps n'est plus valide. L'approche robuste a démontré une grande réussite pour traiter et résoudre les problèmes avec incertitude. Une métaheuristique robuste a été proposée pour résoudre les deux cas réels de Granby et de Saint-Jean-sur-Richelieu. La métaheuristique a été validée par un modèle mathématique sur les petites instances générées à partir des cas réels. La simulation de Monte Carlo a été utilisée pour

évaluer les différentes solutions proposées. En outre, elle permet d'offrir aux gestionnaires un outil de décision pour comparer les différentes solutions robustes, et aussi pour comprendre le compromis entre le niveau de robustesse souhaité et d'autres mesures de performances (coût, risque, niveau de service).

ABSTRACT

This thesis combines two different fields applied to winter road maintenance : operational research and data science. Data science was used to develop a prediction model for the quantity of salt and abrasive with a machine learning methodology, later this model is considered for building vehicles routing. This route planning was developed using operational research which seeks to optimize routes by looking at several constraints and by integrating real data.

The thesis which is the fruit of a collaboration with two Canadian cities Granby and Saint-Jean-sur-Richelieu, deals with a real application in winter road maintenance which is the spreading operation. The spreading operation presents an activity necessary for winter road maintenance, in order to ensure better road traffic. However, this road safety comes with a significant economic and environmental cost, which creates a great concern in order to reduce the economic and environmental impact. This thesis contributes significantly in the spreading operations : firstly, predicting the necessary quantity of salt and abrasive to be spread in order to avoid over-spreading, secondly optimizing the spreading operations routes considering quantity variations.

The first contribution of this thesis is to develop a prediction model for the quantities of salt and abrasive using machine learning algorithms, for each street segment and for each hour. The importance of this contribution lies in the integration of geomatic data with weather-road data, and also the feature engineering. Several machine learning algorithms were evaluated (Random Forest, Extremely Random Trees, Artificial Neural Networks, Adaboost, Gradient Boosting Machine and XGBoost); ultimately XGBoost performed better. The prediction model not only predicts the amounts of salt and abrasive needed to spread, but also identifies the most important variables in the model. This information presents an interesting decision-making tool for managers. The identification of important variables could improve snow removal operations. According to the results, the human factor (driver) significantly influences the amount of spreading, so controlling this factor can significantly improve the spreading operations.

The second contribution introduces a new problem in the literature : the mixed capacitated general routing problem with time-dependent demand; the problem is based on the assumption that the prediction model is able to provide the amount of spreading for each segment and for each hour with good accuracy. Having this information for each hour and for each street segment introduces the concept of time dependency. The new problem was modeled using a mathematical formulation on the original graph, which presents a modeling challenge since it is difficult to associate a unique starting and ending time to an arc or edge. A meta-heuristic based on the destruction and construction strategy has been developed to solve large-scale instances. The meta-heuristic is inspired by SISRs considers time-dependent demand and the presence of edges by an evaluation method based on dynamic programming. New instances were created from the instances of the mixed capacitated general routing problem with fixed demand; the new instances were generated from different types of function where the demand varies with time.

The third contribution proposes a new approach to present the service hierarchy or the priority level of streets, as a time-dependent linear function. The problem addressed in this contribution concerns the hierarchical mixed capacitated general routing problems under demand uncertainty. When the collected data does not allow the development of a good prediction model, the concept of time-dependent demand is no longer valid. The robust approach has demonstrated great success in resolving and dealing with problems with uncertainty. A robust meta-heuristic was proposed to solve the two real cases Granby and Saint-Jean-sur-Richelieu, the meta-heuristic was validated by a mathematical model on small instances generated from the real cases. The Monte Carlo simulation was used, on the one hand, to evaluate the different solutions proposed, and, on the other hand, to offer managers a decision tool to compare the different robust solutions and also to understand the trade-off between the desired level of robustness, and other performance measures (cost, risk, level of service).

TABLE DES MATIÈRES

DÉDICACE	iii
REMERCIEMENTS	iv
RÉSUMÉ	v
ABSTRACT	viii
TABLE DES MATIÈRES	x
LISTE DES TABLEAUX	xiv
LISTE DES FIGURES	xv
LISTE DES SIGLES ET ABRÉVIATIONS	xvii
CHAPITRE 1 INTRODUCTION	1
1.1 Objectifs de recherche	3
1.2 Plan de la thèse	4
CHAPITRE 2 REVUE DE LITTÉRATURE	6
2.1 Le déneigement	6
2.1.1 L'épandage	6
2.1.2 Le déblaiement ou tassement	7
2.1.3 L'enlèvement de la neige	7
2.2 Modèle de prévision de fonte de la neige	8
2.2.1 Le temps de récupération des chaussées	8
2.2.2 Taux d'application de sel	8
2.2.3 Profondeurs de la couche de neige	9
2.2.4 Durabilité du sel	9
2.2.5 Volume de fonte de la neige	9

2.3	Apprentissage machine appliqué aux problèmes hivernaux	10
2.3.1	Réseaux de neurones artificiels	10
2.3.2	Réseaux de neurones convolutifs	11
2.3.3	K-plus proches voisins	11
2.4	Problème de tournées de véhicules pour les opérations d'épandage	12
2.4.1	Les travaux récents sur les problèmes de tournées de véhicules appli- quées aux opérations d'épandage	12
2.5	Problème de tournées de véhicules générales avec contrainte de capacité . . .	15
2.5.1	Les méthodes heuristiques pour MCGRP	15
2.5.2	Les méthodes exactes pour le MCGRP	16
2.6	Problème de tournées de véhicules dépendantes du temps	18
2.6.1	Problème de tournées de véhicules dépendantes du temps sur les nœuds	18
2.6.2	Problème de tournées de véhicules dépendantes du temps sur les arcs	18
2.7	Problème de tournées de véhicules robuste	21
2.7.1	Méthodes exactes pour le problème de tournées de véhicule robuste .	21
2.7.2	Méthodes heuristiques pour le problème de tournées de véhicules robuste	22
2.7.3	Problème de tournées de véhicules robuste sur les arcs	23
CHAPITRE 3	MÉTHODOLOGIE	25
3.1	La collecte et préparation de données	26
3.2	Modèle de prédiction	26
3.3	La modélisation dépendante du temps	27
3.4	La modélisation robuste	28
CHAPITRE 4	ARTICLE 1: STREET-SEGMENT-BASED SALT AND ABRASIVE PREDICTION FOR WINTER MAINTENANCE USING MACHINE LEARNING AND GIS	29
4.1	Abstract	29
4.2	Introduction	29
4.3	Literature review	30
4.3.1	Prediction models for winter maintenance	31

4.3.2	Machine learning applied to winter maintenance	32
4.4	Methodology	34
4.4.1	Data treatment	34
4.4.2	Feature engineering	36
4.4.3	Feature selection	39
4.4.4	Training and testing	40
4.4.5	Machine-learning algorithms	40
4.4.6	Model evaluation	42
4.5	Case study	43
4.5.1	Data description	45
4.5.2	RESULTS AND DISCUSSION	48
4.5.3	Cross-validation	49
4.5.4	XGBoost algorithm evaluation and variable importance	50
4.5.5	Error analysis	53
4.5.6	Policy implications and operational improvements	55
4.6	Conclusions	55
4.7	ACKNOWLEDGMENTS	56

CHAPITRE 5 ARTICLE 2: THE MIXED CAPACITATED GENERAL ROUTING

	PROBLEM WITH TIME-DEPENDENT DEMANDS	57
5.1	Abstract	57
5.2	Introduction	57
5.3	Mathematical formulation	60
5.3.1	Linearization of the demand constraints	65
5.4	Modified SISRs	67
5.4.1	Solution representation	67
5.4.2	Cost and Capacity Evaluation	68
5.4.3	Metaheuristic	70
5.5	Computational Experiments	75
5.5.1	Benchmark instances	75

5.5.2	Instances modification	75
5.5.3	Experimental results	77
5.6	Conclusion	82
CHAPITRE 6 ARTICLE 3: ROBUST OPTIMIZATION FOR THE HIERARCHICAL MIXED CAPACITATED GENERAL ROUTING PROBLEM APPLIED TO WIN- TER ROAD MAINTENANCE		84
6.1	Abstract	84
6.2	Introduction	84
6.3	Formal description and mathematical formulations	89
6.3.1	Formal description and graph transformation	89
6.3.2	Mathematical formulation	90
6.3.3	Linearization of time dependent cost	91
6.3.4	Robust formulation	92
6.4	Robust SISRs	95
6.4.1	Robust heuristic	95
6.4.2	Objective function	96
6.5	Computational experiments	97
6.5.1	Monte Carlo simulation	98
6.5.2	Real-life case studies	99
6.6	Conclusion	105
CHAPITRE 7 DISCUSSION GÉNÉRALE		106
CHAPITRE 8 CONCLUSION ET RECOMMANDATIONS		109
8.1	Synthèse des travaux	109
8.2	Limitations des solutions proposées	110
8.3	Améliorations futures	111
RÉFÉRENCES		112

LISTE DES TABLEAUX

Tableau 2.1: Caractéristiques des modèles de tournées des véhicules pour l'épandage . .	14
Tableau 2.2: Caractéristiques des modèles de tournées des véhicules dépendantes du temps	20
Tableau 2.3: Caractéristiques des modèles de tournées des véhicules robustes	24
Tableau 3.1: Caractéristiques des données utilisées dans la thèse	27
Table 4.1: Input variables description (N = numeric, CAT = categorical)	46
Table 5.1: Instance characteristics	75
Table 5.2: Time dependent demand functions	76
Table 5.3: Comparison of CPLEX results and metaheuristic results ($\alpha = 0.3$)	78
Table 5.4: Comparison of CPLEX results and metaheuristic results ($\alpha = 0.5$)	79
Table 5.5: Comparison of CPLEX results and metaheuristic results ($\alpha = 0.7$)	80
Table 6.1: Comparison of CPLEX results and metaheuristic results	98
Table 6.2: The results of robust SISRs on Granby	101
Table 6.3: The results of robust SISRs on SJSR	101

LISTE DES FIGURES

Figure 3.1: Méthodologie générale	25
Figure 3.2: L'approche apprentissage machine utilisée au chapitre 4	28
Figure 4.1: Diagram summarizing the methodology applied in this research.....	34
Figure 4.2: Map matching of GPS points	35
Figure 4.3: Granby road network	44
Figure 4.4: Granby weather records, 2014–2015.....	44
Figure 4.5: Average amount of salt and abrasive by truck (kg)	47
Figure 4.6: Average amount of salt and abrasive by type of segment (kg)	47
Figure 4.7: Pearson correlation indicates the relationship between the variables and the output	48
Figure 4.8: Feature ranking with the randomized Lasso method.....	49
Figure 4.9: R^2 of machine-learning algorithms.....	50
Figure 4.10: MSE of machine-learning algorithms.....	50
Figure 4.11: Training results with XGBoost.....	51
Figure 4.12: Test results with XGBoost	52
Figure 4.13: Importance of variables	53
Figure 4.14: Error ratio for each segment	54
Figure 5.1: Example of a time dependent demand function for a specific link (i, j)	65
Figure 5.2: Solution representation example.....	68
Figure 5.3: Time dependent demand presentation.....	70
Figure 5.4: Time dependent demand function	70
Figure 5.5: An example of the modified SISRs metaheuristic.....	73
Figure 5.6: Box plots of sensitivity analysis for each alpha and for each function	81
Figure 5.7: Box plots of sensitivity analysis for each function	82

Figure 6.1: Graph transformation	90
Figure 6.2: Cost functions.....	92
Figure 6.3: Granby, QC	102
Figure 6.4: Saint-Jean-sur-Richelieu, QC	102
Figure 6.5: Level service and risk for Granby	103
Figure 6.6: Cost and # of vehicles for Granby.....	103
Figure 6.7: Cost and unified cost for Granby.....	104
Figure 6.8: Price of robustness for Granby	104
Figure 6.9: Level service and risk for SJSR	104
Figure 6.10: Cost and # of vehicles for SJSR	104
Figure 6.11: Cost and unified cost for SJSR.....	104
Figure 6.12: Price of robustness for SJSR	104

LISTE DES SIGLES ET ABRÉVIATIONS

AILS	Recherche locale itérative adaptative (<i>Adaptive Iterative Local Search</i>)
ALNS	Recherche à voisinage adaptatif large (<i>Adaptive Large Neighborhood Search</i>)
ANN	Réseaux de neurones artificiels (<i>Artificial Neural Networks</i>)
ARIMA	Moyenne mobile autorégressive univariée et multivariée (<i>Multivariate Integrated Autoregressive Moving Average</i>)
ARIMAX	Moyenne mobile autorégressive avec des variables externes (<i>Autoregressive Moving Averages with External Variables</i>)
BPRT	Temps de récupération de la chaussée nue (<i>Bare Pavement Regain Time</i>)
CARP	Problème de tournées sur les arcs avec capacités (<i>Capacitated Arc Routing Problem</i>)
CNN	Réseaux de neurones convolutifs (<i>Convolutional Neural Networks</i>)
CPP	Problème de postier chinois (<i>Chinese Postman Problem</i>)
CVRP	Problème de tournées sur les nœuds avec capacités (<i>Capacitated Vehicle Routing Problem</i>)
CVRP	Problème de tournées de véhicules avec capacité (<i>Capacitated Vehicle Routing Problem</i>)
CVRTW	Problème de tournées de véhicules avec capacité fenêtres de temps (<i>Capacitated Vehicle Routing problem with Time Windows</i>)
GIS	(<i>Geographic Information System</i>)
GPS	(<i>Global positioning system</i>)
HVRP	Problème de tournées de véhicules hétérogènes (<i>Heterogeneous Vehicle Routing Problem</i>)
KNN	K-plus proches voisins (<i>K-Nearest Neighbors</i>)
MCGRP	Problème de tournées de véhicules générales (Mixed Capacitated General Routing Problem)

MIP	Programmation mixte en nombres entiers (<i>Mixed Integer Programming</i>)
MTIM	Modèle d'indice de température modifié (<i>Modified Temperature Index Model</i>)
PCARP	Problème de tournées de véhicules sur les arcs périodiques avec contrainte de capacité (<i>Periodic Capacitated Arc Routing Problem</i>)
PTVDT	Problèmes de tournées de véhicules dépendantes du temps
R.O.	Recherche opérationnelle
RPP	Problème du postier rural (<i>Rural Postman Problem</i>)
RSPPRC	Problème robuste de chemin le plus court avec des contraintes de ressources (<i>Robust Shortest Path Problem with Resource Constraints</i>)
RST	Température de la surface de la route (<i>Road Surface Temperature</i>)
RVRP	Problème de tournées de véhicules robuste (<i>Robust Vehicle Routing Problem</i>)
SJSR	Saint-Jean-sur-Richelieu
SIG	Système d'information géographique
SISR _s	Slack Induction by String Removals for Vehicle Routing Problems
TD-ARP	Problème de tournées de véhicules sur les arcs dépendant du temps (<i>Time Dependent Arc Routing Problem</i>)
TD-PARP	Problème de tournées de véhicules sur les arcs pour la collecte de prix dépendant du temps (<i>Time Dependent Prize-Collecting Arc Routing Problem</i>)
TD-RPP	Problème du postier rural dépendant du temps (<i>Time-Dependent Rural Postman Problem</i>)
TD-VRP	Problème de tournées de véhicules sur les nœuds dépendant du temps (<i>Time Dependent Vehicle Routing Problem</i>)
TIM	Modèle d'indice de température (<i>Temperature Index Model</i>)
UHGS	Recherche génétique hybride unifiée (<i>Unified Hybrid Genetic Search</i>)
VRPD	Problème de tournées de véhicules avec des délais (<i>Vehicle Routing Problem with Deadlines</i>)

VRPTW	Problème des tournées de véhicules avec des fenêtres de temps (<i>Vehicle Routing Problem with Time Windows</i>)
VRPTWMD	Problème de tournées de véhicules avec des fenêtres de temps et plusieurs livreurs (<i>Vehicle Routing Problem with Time Windows and Multiple Deliverymen</i>)

CHAPITRE 1 INTRODUCTION

La viabilité hivernale représente une activité importante et indispensable afin d'assurer un déplacement sécuritaire des citoyens sur les réseaux routiers. Par contre, elle reste une activité extrêmement coûteuse pour les municipalités. À titre d'exemple, le budget accordé à la ville d'Edmonton en 2019 pour les opérations de déneigement est de 65M\$ ([Ville d'Edmonton, 2019](#)). Calgary a budgété 40M\$ en 2019 ([Ville de Calgary, 2019](#)). Pour la ville de Québec, les opérations de déneigement exigent un budget annuel de plus de 43 M\$, afin de déneiger et déglacer 2 373 km de chaussées et 1 248 km de trottoirs ([Ville de Quebec, 2020](#)). La ville de Montréal a augmenté le budget de 2.2M\$ en 2020 ([Ville de Montréal, 2019](#)) par rapport au budget de 2019 d'une valeur de 166M\$ ([Ville de Montréal, 2020a](#)). Le coût total de ces opérations s'amplifie en considérant les coûts indirects comprenant les coûts de corrosion d'infrastructures et d'équipements, et les coûts de retard à cause des fermetures des voies de circulation. Ces derniers sont estimés à 3,4 milliards de dollars par année aux États-Unis ([Nixon, 2009](#)). Les opérations de déneigement n'ont pas seulement un impact économique. Il y a aussi l'enjeu environnemental causé par le sel de voirie sur les nappes aquifères, sur le sol, sur la flore, sur la faune aquatique et terrestre ([MTQ, 2020](#)). Les budgets dépensés poussent les municipalités à utiliser des technologies qui peuvent réduire les coûts associés, comme les systèmes d'information météorologique routière et les systèmes de localisation automatisés de suivi des véhicules ([Fu et al., 2009](#)). Les dispositifs de suivi de la télémétrie sont indispensables dans la gestion de flotte. Ils collectent en temps réel de grandes quantités de données, qui peuvent ensuite être exploitées pour permettre une meilleure utilisation des ressources ([Quirion-Blais et al., 2015](#)). Par contre, ils semblent être peu exploités actuellement.

Cette thèse traite particulièrement des opérations d'épandage qui font partie des opérations de déneigement. D'abord, les opérations d'épandage sont modélisées sur un graphe mixte. En effet, il est souvent suffisant d'épandre dans une seule direction de la route. Mais, dans le cas des opérations de déblaiement de la neige, les deux voies de la route doivent être servies (graphe orienté). Une autre grande différence par rapport aux opérations de déblaiement est la contrainte de capacité : chaque segment de rue est associé à une quantité de sel et

d’abrasif qui dépend des plusieurs paramètres (longueur des rues, nombre de voies, météo ...). D’autres contraintes sont considérées aussi pour les opérations d’épandage et sont semblables aux opérations de déblaiement : équilibrage des routes, hiérarchie de réseaux. Les opérations d’épandage ont des impacts environnementaux négatifs ; de plus, elles sont coûteuses. Au Québec, 1.5 million de tonnes de sel sont épandues annuellement, ce qui représente un coût direct variant de 704 millions à 2.2 milliards de dollars (MTQ, 2020). En outre, il y a le coût de routage qui incluent les frais de carburant, camions, conducteurs, etc. Donc, l’optimisation de ces opérations devient une nécessité dans le but de réduire leur impact environnemental et économique.

Généralement, les opérations d’épandage en R.O. s’intègrent dans les problèmes de tournées de véhicules sur les arcs avec contrainte de capacité (CARP : *Capacitated Arc Routing Problem*) (Corberan et Laporte, 2014). Cependant, il est plus approprié de modéliser certains endroits par des nœuds au lieu des arcs, comme : les intersections, les arrêts de bus, les entrées d’écoles et d’hôpitaux, etc. Pour cette raison, les opérations d’épandage seront modélisées par un problème de tournées de véhicules générales sur les arcs, les arêtes et les nœuds (MCGRP : *Mixed Capacitated General Routing Problem*).

Dans la revue de littérature, les recherches précédentes n’ont pas considéré la variation de la quantité de sel et d’abrasif en fonction de temps. D’une part, cette variation n’a jamais été prise en compte dans la modélisation en recherche opérationnelle. D’autre part, les chercheurs ont ignoré que cette quantité dépend non seulement des variables météorologiques (température, vent, température de chaussée, épaisseur de neige, rayonnement solaire), mais aussi des caractéristiques géométriques du réseau routier (pente, intersection, trafic).

Cette variation de la quantité de sel et d’abrasif peut être quantifiée en développant des modèles théoriques complexes, qui prennent en compte tous les facteurs dépendants basés sur des lois de chimie et de physique ; ou en développant un modèle de prédiction basé sur les données historiques provenant des données de télémétrie et des données météorologiques.

Un modèle de prédiction de la quantité d’épandage peut servir à prédire les quantités de sel et d’abrasif pour chaque heure et pour chaque segment de rue. Ce modèle pourrait être utilisé pour construire des tournées de véhicules pour les opérations d’épandage avec des quantités

dépendantes du temps. Par contre, pour élaborer un bon modèle de prédiction, il faut avoir des quantités énormes de données enregistrées sur différentes périodes hivernales, ce qui n'est pas évident. Dans le cas de manque d'une large base de données, la variation de la quantité de sel et d'abrasif peut être intégrée dans le problème de confection des tournées de véhicules par deux types de modélisation :

- Modélisation basée sur l'optimisation stochastique ;
- Modélisation basée sur l'optimisation robuste.

L'optimisation robuste est plus rapide et pratique par rapport à l'optimisation stochastique. Malgré qu'elle soit répandue dans la résolution de grandes variétés de problèmes avec incertitude, l'optimisation stochastique a deux grandes lacunes : premièrement, elle suppose que la distribution de probabilité est connue avec précision, ce qui est rarement le cas en pratique ; deuxièmement, les problèmes de taille réaliste sont souvent intraitables en terme du temps du résolution ([Subramanyam *et al.*, 2020](#)).

Dans le cadre de cette thèse, nous avons collaboré avec deux villes québécoises : Granby et Saint-Jean-sur-Richelieu. Nous avons basé notre approche sur des données massives réelles provenant des capteurs des camions d'épandage, des sites météorologiques et des topologies des villes.

1.1 Objectifs de recherche

L'objectif principal de cette thèse est d'améliorer les opérations d'épandage en viabilité hivernale en utilisant des données de télémétrie enregistrées par des capteurs dans les camions d'épandage, et des informations météorologiques et géomatiques. Nous visons à proposer aux gestionnaires des outils d'aide à la décision pour les guider à bien planifier les opérations d'épandage. Pour ce faire, nous proposons l'élaboration des modèles de prédiction des quantités d'épandage en fonction de plusieurs variables, et des algorithmes pour la confection des tournées d'épandage. Le travail de cette thèse est divisé en trois objectifs :

1. Le premier objectif consiste à développer un modèle de prédiction des quantités de

sel et d'abrasif en fonction des données météorologiques, des données géométriques du réseau routier, et des données de télémétrie. Ce modèle de prédiction peut être ensuite utilisé pour optimiser les tournées d'épandage car il prédit la quantité nécessaire pour chaque segment et pour chaque heure.

2. Le deuxième objectif consiste à développer un programme mathématique qui modélise le problème de tournées de véhicules pour les opérations d'épandage dont la quantité dépend du temps. Nous supposons ici que le modèle de prédiction développé en 1 est satisfaisant et précis. Nous développons également une métaheuristique qui sert à résoudre les grandes instances de ce problème.
3. Le troisième objectif consiste à développer un modèle mathématique robuste pour le problème de tournées de véhicules pour les opérations d'épandage dont la quantité est incertaine. Nous assumons que les données ne permettent pas d'élaborer un bon modèle de prédiction. Une métaheuristique robuste est développée et appliquée sur les données réelles de la ville de Granby et la ville de Saint-Jean-sur-Richelieu.

1.2 Plan de la thèse

Cette thèse est composée de six chapitres. Le chapitre 2 est dédié à un état de l'art des modèles de prédiction pour la viabilité hivernale, et des principaux travaux qui traitent des tournées d'épandage et des problèmes de tournées de véhicules. Les chapitres 3, 4 et 5 présentent respectivement les articles scientifiques reliés au premier, deuxième et troisième objectif. Le premier article s'intitule "Street-segment-based salt and abrasive prediction for winter maintenance using machine learning and GIS". Il a été publié dans la revue *Transactions in GIS* ([Ahabchane et al., 2019](#)). Le deuxième article s'intitule "The Mixed Capacitated General Routing Problem with Time-Dependent Demands". Il a été soumis pour publication dans la revue *Networks*. Le troisième article s'intitule "Robust optimization for the hierarchical mixed capacitated general routing problem applied to winter road maintenance" soumis pour publication dans la revue *Computers & Industrial Engineering*.

Le chapitre 6 présente la discussion générale sur les travaux réalisés. Le dernier chapitre est

une conclusion générale qui présente une synthèse des travaux que nous avons réalisés, et propose quelques perspectives de recherches futures.

CHAPITRE 2 REVUE DE LITTÉRATURE

Dans ce chapitre, nous présentons une revue sur différents sujets, puisque la thèse combine deux domaines différents. Les trois premières sections traitent des opérations de déneigement, des modèles de prévision appliqués en viabilité hivernale, et de l'application de méthodes d'apprentissage machine aux problèmes hivernaux en général. La quatrième section présente une revue sur les problèmes de tournées de véhicules pour les opérations d'épandage. La cinquième section traite les problèmes de tournées de véhicules générales avec contrainte de capacité (MCGRP). Les deux dernières sections présentent les problèmes des tournées de véhicules avec la dépendance du temps, et les problèmes des tournées de véhicules robustes.

2.1 Le déneigement

Le déneigement peut être défini ([Rousseau, 2019](#)) comme : «L'ensemble des opérations incluant le déblaiement de la neige tombée sur la chaussée des rues pavées et non pavées et sur toutes autres voies publiques affectées à la circulation incluant les trottoirs et les passages de piétons ou autres endroits définis par la Ville. Il signifie la fourniture et l'épandage des abrasifs et de fondant à glace ainsi que le déglacage, le dégagement des puisards, le tassage de la neige dans les rues, les ronds-points et les culs-de-sac, le dégagement des triangles de visibilité, l'enlèvement de la neige et tous autres travaux connexes à l'entretien des chemins d'hiver.»

Le déneigement est une activité primordiale pendant l'hiver pour assurer un déplacement sécuritaire des voitures sur les routes, et des piétons sur les trottoirs. Il est composé de trois opérations principales : l'épandage, le déblaiement et l'enlèvement de la neige.

2.1.1 L'épandage

L'opération d'épandage consiste à épandre des fondants et/ou abrasifs, sur la chaussée et sur les trottoirs. L'épandage débute avec les précipitations (ou avant pour un épandage préventif), et se fait aussi une fois que l'opération d'enlèvement de la neige est complétée. La politique

change d'une ville à l'autre. Par exemple la [Ville de Laval \(2018\)](#) définit quatre situations d'épandage : verglas et accumulation inférieure à 1,5 cm ; accumulation entre 1,5 et 10 cm ; accumulation de plus de 10 cm ; et après l'enlèvement de la neige. Les produits d'épandage varient aussi selon les conditions météorologiques, à titre d'exemple la [Ville de Mont-royal \(2020\)](#) épand du sel si la température est entre 0 et -10°C, un mélange de 50% de sel et 50% de sable si la température est entre -10 et -15°C, et si la température inférieure à -15°C un mélange composé de 90% de sable et 10% de sel.

2.1.2 Le déblaiement ou tassement

L'opération de déblaiement consiste à tasser ou déplacer la neige afin de libérer la chaussée des rues, des trottoirs et de toutes autres voies publiques. Comme l'opération d'épandage, la politique de déblaiement change d'une ville à l'autre. Le déblaiement débute lorsque l'accumulation au sol atteint 1,5 cm comme à la [Ville de Laval \(2018\)](#), ou lorsqu'il y a une accumulation de 5 cm comme Longueuil, Montréal et Québec ([Ville de Longueuil, 2020](#); [Ville de Montréal, 2020b](#); [Ville de Quebec, 2009](#)).

2.1.3 L'enlèvement de la neige

L'enlèvement de la neige consiste à retirer la neige accumulée en bordure des routes ou sur la chaussée en la soufflant dans les camions ou sur les terrains riverains.

Les voies de circulation des villes sont classifiées par catégorie afin de prioriser ces opérations ([Ville de Montréal, 2020b](#)) :

- Priorité 1 : artères principales, artères secondaires, rues très étroites, circuits d'autobus prioritaires et voies réservées, entrées d'hôpitaux.
- Priorité 2 : les rues collectrices, rues commerciales locales, circuits d'autobus réguliers, présence d'un débarcadère scolaire.
- Priorité 3 : rues résidentielles, rues locales, rues de secteurs résidentiels, rues de secteurs industriels, rues à faible débit de circulation.

Dans cette thèse, nous examinons les problèmes d'épandage.

2.2 Modèle de prévision de fonte de la neige

Plusieurs études ([Blackburn *et al.*, 2004](#); [Akin et Shi, 2010](#); [Druschel, 2012](#); [Hossain *et al.*, 2016](#)) ont étudié l'effet de différents facteurs influençant les performances de fonte de la neige : température de chaussée, trafic, épaisseur de neige, rayonnement solaire, type de produit chimique, types de chaussée, état du vent et taux d'application. Dans cette section, nous présentons les études pertinentes liées au modèle de prédiction pour la viabilité hivernale. Les objectifs du modèle de prédiction étaient de prendre en compte le taux d'application, l'épaisseur de la couche de neige, la durabilité des fondants, le temps de récupération des chaussées et le volume de fonte de la neige.

2.2.1 Le temps de récupération des chaussées

[Hossain *et al.* \(2014\)](#) ont utilisé une régression linéaire multiple pour prédire la performance de fonte de la neige et pour identifier les facteurs de contribution, sous différents taux d'application et différentes conditions météorologiques. Ils ont utilisé le temps de récupération de la chaussée nue (BPRT : *Bare pavement regain time*) comme indicateur de performance de fonte de la neige, ce qui présente le temps entre l'application de sel et le moment où le 80% du pavé est dégagé. La température de la chaussée, les taux d'application de sel et la quantité de neige sont les variables explicatives utilisées dans le modèle. Les résultats ont montré l'importance des variables, même si le modèle de prédiction n'était pas efficace avec $R\text{-carré} = 0.39$. Les tests ont été effectués sur un stationnement à l'Université de Waterloo ; cela signifie que les analyses n'ont pas examiné d'autres variables telles que le volume de trafic et les caractéristiques des rues (type de rue, longueur, virages, pentes ...).

2.2.2 Taux d'application de sel

[Hossain *et al.* \(2015\)](#) ont élaboré un modèle physique pour prédire le taux d'application optimal pour les parcs de stationnement et les trottoirs. Ils ont utilisé un modèle théorique basé sur la thermodynamique, la concentration et la température eutectique, la cinétique des taux de réaction de premier ordre, le modèle de fonte de la neige, la concentration de sel requise. Le modèle a été étalonné en utilisant les données des essais sur le terrain, mais les

résultats ont montré une mauvaise précision d'estimation.

2.2.3 Profondeurs de la couche de neige

Feng et Fu (2015) ont introduit des approches de modélisation de séries temporelles telles que la moyenne mobile autorégressive univariée et multivariée (ARIMA : *Multivariate integrated autoregressive moving average*) et la moyenne mobile autorégressive avec des variables externes (ARIMAX : *Autoregressive moving averages with external variables*). Le modèle a été élaboré afin de prédire les mesures de la surface de la route (température de la surface et profondeurs de la couche de neige). Le modèle de prédiction a utilisé des données historiques. Le trafic, les conditions météorologiques et les opérations d'épandage et de déneigement ont été considérés comme des variables explicatives. Le modèle de prédiction était bon, avec un R-carré entre 0.85 et 0.97.

2.2.4 Durabilité du sel

Blomqvist *et al.* (2011) ont présenté une combinaison d'un modèle empirique développé par l'Institut national suédois de recherche sur les routes et les transports, avec les données de l'autorité routière au Danemark. Le modèle empirique est modélisé sur une fonction mathématique qui prédit la durabilité d'une dose de sel donnée en fonction des conditions de circulation et de l'humidité de la surface de la route. Les résultats montrent que le modèle de prédiction était assez bon avec un R-carré de 0.89. Dans la même direction, İkiz et Galip (2016) ont estimé que la concentration de sel diminuait après sa propagation ; ils ont utilisé aussi une fonction mathématique dépend du trafic et du temps, et cette étude a été réalisée en premier lieu sur le terrain pour découvrir les caractéristiques d'influence et, afin de corriger les erreurs commises lors des essais sur le terrain, ils ont utilisé des tests en laboratoire. Le modèle de prédiction a fourni de bons résultats avec un R-carré élevé (entre 0.82 et 0.99).

2.2.5 Volume de fonte de la neige

Trenouth *et al.* (2015) ont introduit le modèle d'indice de température modifié (MTIM : *Modified temperature index model*) pour prédire le volume de fonte de la neige. MTIM utilise

deux variables météorologiques des information sur les opérations de déneigement, ce qui est une amélioration du modèle d'indice de température (TIM : *temperature index model*), qui ignore les opérations de déneigement. Les auteurs ont utilisé trois sites pour étalonner et valider le modèle. Le MTIM peut être utilisé pour optimiser le taux d'application du sel pour certaines conditions météorologiques. Le modèle de prédiction était bon avec un R-carré de 0.99.

2.3 Apprentissage machine appliqué aux problèmes hivernaux

L'apprentissage machine a été utilisé dans divers domaines, notamment en viabilité hivernale. Ici on présente les différents algorithmes d'apprentissage machine qui ont été utilisés dans ce contexte.

2.3.1 Réseaux de neurones artificiels

[Shi et Akin \(2012\)](#) ont analysé les performances et la corrosivité des solutions de déglacant au chlorure à l'aide des réseaux de neurones artificiels (ANN : *Artificial Neural Networks*). Les ANN ont été utilisés pour explorer la corrélation entre la corrosion électrochimique et gravimétrique (R-carré = 0.84), entre la composition du déglacant et la corrosivité du déglacant (R-carré = 0.90), et entre la composition du déglacant et les performances du déglacant (R-carré = 0.98). Ce travail a été développé par [Shi et al. \(2013\)](#), qui ont étudié les données de performance de référence des dégivreurs au chlorure typiques utilisés sur les autoroutes. Pour cette nouvelle étude, ils ont développé trois modèles basés sur les ANN : le premier modèle a été établi pour corréler les données de corrosion électrochimique avec celles de la méthode d'essai gravimétrique ; deux modèles supplémentaires ont été utilisés pour quantifier les corrélations entre la composition du déglacant (type, concentrations en chlorure et inhibiteur, pH et conductivité électrique) et leur corrosivité et performance, respectivement. Ces modèles peuvent être utilisés pour améliorer la conception de dégivreurs plus puissants et moins corrosifs. [Ye et al. \(2009\)](#) ont utilisé des ANN pour évaluer les effets des informations météorologiques sur les coûts d'entretien hivernal. L'analyse de sensibilité montre que les kilomètres de voies entretenus par les unités d'entretien ont l'impact le plus important sur

les coûts d'entretien hivernal ; les résultats montrent que l'utilisation et la précision accrue des informations météorologiques peuvent réduire les coûts. Le modèle de prédiction a un $R\text{-carré} = 0.93$. [Ye et al. \(2014\)](#) ont utilisé les ANN pour modéliser la relation entre la sécurité routière en termes de nombre d'accidents de voiture en hiver et d'autres caractéristiques telles que l'utilisation de produits chimiques, le volume de trafic, la longueur et les conditions météorologiques. Une analyse de sensibilité a été menée pour identifier les caractéristiques importantes et étudier leurs effets. Les résultats ont révélé que les produits chimiques d'hiver amélioreraient la sécurité routière et que les avantages l'emportaient sur les coûts. Le modèle de prédiction a un $R\text{-carré}$ entre 0.62 et 0.75.

2.3.2 Réseaux de neurones convolutifs

L'apprentissage profond a été utilisé pour la première fois dans l'entretien des routes en hiver par [Pan et al. \(2018\)](#). Le modèle d'apprentissage profond est basé sur des réseaux de neurones convolutifs (CNN : *Convolutional Neural Networks*), afin de classer les conditions de surface de la route sur la base d'images provenant de cellulaires. Les résultats montrent que le modèle proposé surpasse les ANN, les machines à vecteurs de support et les arbres de classification aléatoires avec une précision supérieure à 78%.

2.3.3 K-plus proches voisins

D'autres algorithmes, différents des ANN, ont également été utilisés. [Yun et al. \(2018\)](#) ont utilisé les K-plus proches voisins (KNN : *K-Nearest Neighbors*) pondérés pour prédire la température de la surface de la route (RST : *Road Surface Temperature*), qui est considérée comme cruciale pour développer des stratégies d'entretien hivernal en matière de sécurité routière ; cette étude évalue la faisabilité de prédire la RST sur la base d'informations climatiques (c'est-à-dire humidité et température de l'air) et de la température ambiante qui peuvent être collectées via des véhicules sondes. Les résultats montrent que les performances globales étaient satisfaisantes.

2.4 Problème de tournées de véhicules pour les opérations d'épandage

Les problèmes de déneigement ont attiré une grande attention des chercheurs en recherche opérationnelle, surtout que les problèmes de viabilité hivernale sont liés généralement aux problèmes des tournées de véhicules. [Campbell et Langevin \(1995a,c, 2000\)](#) ont présenté des travaux sur différents problèmes de déneigement et sur les principales applications réussies aux États-Unis, au Canada et au Royaume-Uni. Après plusieurs travaux dans ce domaine, [Perrier *et al.* \(2006a,b, 2007a,b\)](#) ont élaboré une revue de littérature pertinente répartie en quatre articles : la conception des systèmes d'épandage et de déblaiement, la conception des systèmes de chargement et d'élimination de la neige, les tournées de véhicules et la localisation des dépôts pour l'épandage ainsi que les tournées de véhicules et la localisation des dépôts pour le déneigement. [Campbell *et al.* \(2014a\)](#) ont présenté les dernières recherches sur les opérations de déblaiement en recherche opérationnelle. Dans cette thèse, nous nous intéressons aux opérations d'épandage, qui sont modélisées généralement en recherche opérationnelle par le problème de tournées sur les arcs avec capacités (CARP). [Eglese *et al.* \(2014\)](#) et [Perrier *et al.* \(2012\)](#) ont présenté des revues de littérature sur les nouveaux modèles d'optimisations et les méthodes de solutions utilisées pour les opérations d'épandage. Puisque ces revues étaient exhaustives, une récente revue sur l'épandage (le Tableau 2.1 présente l'extension de la revue ([Perrier *et al.*, 2007a](#))) est présentée.

2.4.1 Les travaux récents sur les problèmes de tournées de véhicules appliquées aux opérations d'épandage

[Hajibabai *et al.* \(2014\)](#) ont combiné pour la première fois les opérations d'épandage et déblaiement dans le but de minimiser la plus grande tournée. Ils ont proposé un modèle mathématique et une heuristique basée sur la recherche locale. Une version dynamique de ce problème a été proposé par [Hajibabai et Ouyang \(2016\)](#) avec une demande incertaine et perturbation des services ; un modèle de gestion de flotte dynamique stochastique est développé pour affecter les camions disponibles pour couvrir la demande incertaine. L'objectif est de minimiser simultanément le coût de déplacement et du repositionnement des camions, ainsi que de maximiser le niveau de service. [Kinable *et al.* \(2016\)](#) ont considéré le problème de

déblaiement et d'épandage simultanément sous des contraintes d'utilisation et du réapprovisionnement des ressources (sel, carburant) et des restrictions pour les véhicules. Les auteurs ont proposé un modèle mathématique en nombre entier, un modèle de programmation par contraintes et une heuristique constructive.

Quirion-Blais *et al.* (2017) ont étudié aussi le problème de déblaiement et d'épandage combinés ou certains camions sont équipés pour déneiger, certains pour épandre et d'autres peuvent réaliser les deux tâches simultanément. Pour ce problème, ils ont proposé un algorithme de recherche adaptative de grand voisinage (ALNS : *Adaptive Large Neighborhood Search*) afin de minimiser la plus grande tournée, en considérant la hiérarchie du réseau, des restrictions sur les virages autorisés et les vitesses d'opérations. Lukman *et al.* (2018) ont proposé une heuristique pour un modèle d'optimisation en temps réel en prenant compte les priorités, les événements inattendus, l'utilisation des ressources et les impacts environnementaux ; de plus, ils ont étudié des scénarios avec l'ajout des dépôts de recharge de sel dans la zone de service. Sullivan *et al.* (2019) ont présenté une heuristique itérative basée sur l'approche partition en premier, routage en second utilisée pour résoudre les problèmes combinés. Ils ont développé une nouvelle mesure de priorité pour chaque rue du réseau et pour chaque jour. L'heuristique utilisée sert à minimiser le nombre total d'heures et de minimiser le temps total nécessaire pour desservir les rues les plus critiques du réseau routier.

Tableau 2.1 – Caractéristiques des modèles de tournées des véhicules pour l'épandage

Auteurs	Objectif	Contraintes	Méthode de résolution
Soyster (1974)		Hierarchie Durée maximale des tournées	Méthode constructive parallèle
Cook et Alprin (1976)	Minimiser le temps d'épandage	Capacité Équilibrage de charge	Méthode constructive parallèle
Ungerer (1989)	Minimiser la distance totale	Capacité Longueur maximale des tournées	Méthode constructive parallèle
Evans (1990) Evans et Weant (1990)	Minimiser la distance totale	Capacité Durée maximale d'épandage Durée maximale des tournées	Méthode constructive séquentielle
Li et Eglese (1996)	Minimiser les coûts des véhicules et les coûts de transport	Hierarchie Capacités Dépôts multiples	Méthode constructive séquentielle
Benson <i>et al.</i> (1998)	Minimiser la violation des durées, et le retour à vide	Capacité Vitesse de service Durée des opérations	Recherche taboue
Qiao (2002)	Minimiser le retour à vide	Capacité Niveau de service	Recherche taboue
Haghani et Qiao (2002)	Minimiser le retour à vide	Hierarchie Capacité Durée maximale des tournées Niveau de service	Heuristique à 4 étapes
Toobaie et Haghani (2004)	Minimiser la taille de la flotte,	Capacité Équilibrage de charge Dépôts multiples Fréquences du service	Algorithme génétique
Handa <i>et al.</i> (2005)	Minimiser la distance totale	Capacité	Algorithme génétique
Omer (2007)	Minimiser la distance totale	Capacité Nombre maximal de véhicules Fréquences du service Durée maximale des tournées	Recuit simulé
Tagmouti <i>et al.</i> (2007)	Minimiser les coûts de voyage, et les coûts de service dépendant de temps	Capacité Nombre maximal de véhicules Service dépendant de temps Durée maximale d'épandage	Génération de colonnes
Tagmouti <i>et al.</i> (2010)	Minimiser les coûts de voyage, et les coûts de service dépendant de temps	Capacité Service dépendant du temps	Recherche taboue
Tagmouti <i>et al.</i> (2011)	Minimiser les coûts de voyage, et les coûts de service dépendant de temps	Capacité Service dépendant du temps	Recherche de voisinage variable
Hajibabai <i>et al.</i> (2014)	Minimiser la plus grande tournées	Capacité, Hiérarchie	Recherche locale
Kinable <i>et al.</i> (2016)	Minimiser la plus grande tournée	Capacité (sel, carburant) Véhicule hétérogène Multidépôts	Modèle mathématique (MIP) Programmation par contraintes Heuristique constructive
Quirion-Blais <i>et al.</i> (2017)	Minimiser la plus grande tournée	Capacité, Hiérarchie Restrictions sur les virages Vitesses d'opérations	Recherche adaptative de grand voisinage
Lukman <i>et al.</i> (2018)	Minimiser la distance totale	Hiérarchie, capacité, Événements inattendus	Heuristique basée sur l'approximation du plus cours chemin
Sullivan <i>et al.</i> (2019)	Minimiser le nombre total d'heures, et le temps pour desservir les rues les plus critiques	Capacité, Niveau de service dynamique	Heuristique itérative combine le partitionnement allocation et tournées de véhicules

2.5 Problème de tournées de véhicules générales avec contrainte de capacité

Le problème de tournées de véhicules avec capacité est un problème combinatoire dont le but est de satisfaire la demande des clients une seule fois, sans violer la contrainte de capacité des véhicules, tout en minimisant le coût total du déplacement en partant et en revenant au dépôt. Les travaux sur ce problème sont divisés en trois catégories :

- Le problème de tournées de véhicule sur les nœuds avec capacité (CVRP) introduit par [Dantzig et Ramser \(1959\)](#) où les nœuds représentent des clients (ex : livraison des biens (marchandises, colis, journaux), tournées des infirmières). Pour plus de détails voir le livre par [Toth et Vigo \(2014\)](#).
- Le problème de tournées de véhicule sur les arcs avec capacité (CARP) introduit par [Golden et Wong \(1981\)](#) où les arcs et les arêtes représentent les clients (ex : collection de déchets, surveillance des rues, balayage des rues). Pour plus de détails voir le livre par [Corberan et Laporte \(2014\)](#).
- Le problème de tournées de véhicules générales mixte avec capacité (MCGRP) introduit [Pandi et Muralidharan \(1995\)](#) où les tournées servent les arcs, les nœuds et les arrêtes, (ex : collection de déchets, tournées de bus scolaires).

Cette thèse traite en particulier le problème de tournées de véhicules générales avec capacité. Bien que le MCGRP soit récent, il y a plusieurs chercheurs qui se sont intéressés à ce problème, soit avec des méthodes heuristiques, ou bien des méthodes exactes. Ici, nous présentons les différentes approches de résolution.

2.5.1 Les méthodes heuristiques pour MCGRP

[Pandi et Muralidharan \(1995\)](#) ont présenté une heuristique basée sur la création d'un tour géant qui est ensuite divisé en petites routes pour chaque véhicule (routage en premier, partition en second). [César et al. \(2002\)](#) ont amélioré les résultats des premiers en utilisant une approche différente basée sur la confection des tournées après l'affectation des secteurs (partition en premier, routage en second). [Prins et Bouchenoua \(2005b\)](#) ont développé l'algorithme

mémétique avec cinq opérateurs de recherche locale pour la phase d'éducation ; l'heuristique de Split du tour géant a été utilisée pour la solution initiale ; les auteurs ont créé un nouvel ensemble d'instances CBMix inspirées par des cas réels. [Kokubugata et al. \(2007\)](#) ont proposé la métaheuristique recuit simulé basée sur trois opérateurs : l'échange de tâches entre deux routes, le déplacement de tâches vers une autre route et l'échange de tâches dans la même route. Cette métaheuristique a trouvé dix meilleures solutions sur CBMix. [Bosco et al. \(2014\)](#) ont utilisé une métaheuristique basée sur la recherche de voisinage à partir d'une solution initiale générée avec un algorithme constructif ; de plus un algorithme de branch-and-cut est invoqué à la fin de la phase d'intensification. La métaheuristique a obtenu 4 meilleures solutions sur Cbmix par rapport à ([Kokubugata et al., 2007](#)). [Mandal et al. \(2015\)](#) ont proposé aussi un algorithme mémétique, mais pour un nouveau problème le MCGRP biobjectif où l'objectif est de minimiser le coût total de voyage et équilibrer la charge de travail tout en minimisant la différence de coût entre la tournée la plus longue et la tournée la plus courte. Récemment, [Dell'Amico et al. \(2016\)](#) présentent une nouvelle métaheuristique hybride AILS recherche locale itérative adaptative (AILS : *Adaptive Iterative Local Search*) dédiée aux instances de grande taille. AILS utilise les mécanismes de deux métaheuristiques classiques : la recherche locale itérée et la recherche de large voisinage adaptatif. De nouveaux opérateurs de recherche locale à large voisinage ont été développés spécialement pour AILS. Les expériences sur tous les problèmes étalons existants de MCGRP ont montré d'excellentes performances. Dans la même année, [Vidal \(2017\)](#) a étudié une famille de voisinage étendu pour les problèmes des tournées sur les arcs, qui impliquent des délocalisations et des échanges de services avec des choix de modes optimaux combinés sur les routes en utilisant la programmation dynamique. L'algorithme recherche génétique hybride unifiée (UHGS : *Unified Hybrid Genetic Search*) exécuté sur les 5 instances de MCGRP a montré les meilleures performances jusqu'à maintenant.

2.5.2 Les méthodes exactes pour le MCGRP

Jusqu'à 2012, il n'y avait pas de méthode exacte pour le MCGRP. La première formulation a été proposée par [Bosco et al. \(2013\)](#) et résolue avec l'algorithme branch-and-cut en développant des inégalités valides du problème de CARP. L'algorithme est testé sur des nouvelles

instances MGGDB et MGVAL (dérivées des instances existantes GDB et MVAL). [Irnich et al. \(2015\)](#) ont proposé une nouvelle méthode de branch-and-cut différente de [Bosco et al. \(2013\)](#) avec de meilleurs résultats. Les procédures de bornes supérieures ont été discutées par [Hasle et al. \(2012\)](#), qui a obtenu des résultats intéressants en utilisant le solveur commercial Spider. [Bach et al. \(2013\)](#) ont proposé la première procédure de borne inférieure pour le MCGRP basée sur [Wøhlk \(2006\)](#). Ils ont développé également deux nouveaux problèmes étalons : 1) BHW basé sur 20 instances CARP bien connues dans la littérature, 2) DI-NEARP se compose de 24 instances définies à partir de cas réels de la livraison de journaux. [Gaze \(2013\)](#) ont proposé l'algorithme branch-and-price inspiré par la formulation de [Letchford et Oukil \(2009\)](#). L'algorithme a été testé sur les 5 instances connues de MCGRP (MGGDB, MGVAL, BHW, Di-NEARP, CBMix), 14 nouvelles solutions optimales ont été trouvées. [Bach et al. \(2016\)](#) ont présenté l'algorithme branch-and-cut-and-price inspiré par [Fukasawa et al. \(2006\)](#). L'idée principale est de combiner deux modèles mathématiques différents pour le problème, un modèle pour la génération de coupes et l'autre pour la génération de colonnes. Le but est de présenter un modèle robuste comme l'approche de ([Fukasawa et al., 2006](#)) où les coupes peuvent être ajoutées dans la configuration de branch-and-price. Récemment, [Ciancio et al. \(2018\)](#) ont proposé aussi l'algorithme branch-and-cut-and-price pour le MCGRP avec fenêtres de temps après une transformation du graphe. Ils ont présenté de nouvelles instances dérivées à partir des instances provenant de CARP avec fenêtres de temps et de MCGRP.

À notre meilleure connaissance, jusqu'à maintenant aucun article scientifique n'a traité les problèmes de déneigement avec le MCGRP. Dans cette thèse nous présentons le problème d'épandage comme un problème de MCGRP car nous considérons que les intersections, les arrêts de bus et les virages correspondent à des nœuds, les rues à sens unique correspondent à des arcs, les rues larges à double sens, dont l'épandage doit être effectué des deux côtés correspondent à deux arcs opposés, et les rues étroites à double sens dont l'épandage peut être effectué en une seule direction correspondent tous à des arêtes.

2.6 Problème de tournées de véhicules dépendantes du temps

Les problèmes de tournées de véhicules dépendantes du temps (PTVDT) ont reçu récemment plus d'attention d'une part grâce à l'intérêt croissant des chercheurs pour l'utilisation de données réelles dans les problèmes académiques, et d'autre part grâce à des progrès récents des techniques de traitement des données. Le tableau 2.2 résume les travaux sur les problèmes PTVDT, on lit de gauche à droite : les auteurs, les noms des problèmes, la variante de la dépendance du temps, la méthode de solution et au type de problème.

2.6.1 Problème de tournées de véhicules dépendantes du temps sur les nœuds

L'importance de la notion de dépendance du temps se manifeste dans son application dans différents domaines : planification des tournées sur les réseaux routiers, planification de déplacements dans les réseaux de transport en commun, ainsi que dans certaines applications robotiques et militaires (Gendreau *et al.*, 2015). En particulier, les problèmes de tournées de véhicules sur les nœuds dépendant du temps (TD-VRP : *Time Dependent Vehicle Routing Problem*) ont été largement étudiés dans la littérature avec différentes variantes de dépendance du temps : le temps de service (Malandraki et Daskin, 1992; Ichoua *et al.*, 2003; Donati *et al.*, 2008), les fenêtres de temps (Dabia *et al.*, 2013; Jung et Haghani, 2001; Balseiro *et al.*, 2011), la demande (Victoria *et al.*, 2016). Pour plus de détails, nous renvoyons le lecteur à Mourão et Pinto (2017) et Gendreau *et al.* (2015).

2.6.2 Problème de tournées de véhicules dépendantes du temps sur les arcs

Contrairement au problème TD-VRP, le problème de tournées sur les arcs (TD-ARP : *Time Dependent Arc Routing Problem*) a été abordé dans très peu d'études et sous la forme de plusieurs variantes de dépendance du temps (temps de service, coût, option zigzag). Le temps de service était la variante la plus abordée de TD-ARP dans la littérature. Sun *et al.* (2011), Tan et Sun (2011) et Sun *et al.* (2015) ont présenté différentes approches de solution pour le problème du facteur chinois avec un temps de voyage dépendant du temps. Black *et al.* (2013) a introduit un nouveau problème nommé le problème de tournées de véhicules sur les arcs pour la collecte de prix dépendant du temps (TD-PARP : *Time Dependent Prize-Collecting Arc*

Routing Problem). Ce problème décrit la situation dans laquelle un gestionnaire de transport doit choisir entre un certain nombre de ramassages et de livraisons complètes de camions sur un réseau routier avec un temps de trajet dépendant du temps. [Kayacı Çodur et Yilmaz \(2018\)](#) ont utilisé l'algorithme génétique et recuit simulé hybride pour le problème hiérarchique du facteur chinois avec temps de trajet dépendant du temps. [Tagmouti et al. \(2007\)](#) et [Tagmouti et al. \(2010\)](#) ont présenté une variante de coût dépendant du temps pour un problème de tournées de véhicules sur les arcs avec contrainte de capacité. Le problème était motivé par l'application de l'épandage hivernal où le moment de l'intervention est crucial. Ils ont considéré un coût de service sur chaque arc comme une fonction linéaire par morceaux. Si le service d'un arc est trop tôt ou trop tard, le coût en temps et en matériel d'épandage augmente. Cette fonction inclut également un intervalle de temps optimal où le coût du service est minimale. [Tagmouti et al. \(2011\)](#) ont traité une variante dynamique du même problème. L'aspect dynamique modifie ces intervalles de temps de service optimaux en fonction des mises à jour de la météo. [Nossack et al. \(2017\)](#) ont combiné la dépendance du temps dans un problème de tournées de véhicule sur les arcs avec une option en zigzag, ce qui signifie de desservir les deux côtés de la rue en un seul passage. Le zigzag est autorisé lorsque la rue est étroite ou lorsque la circulation est faible pendant certaines périodes de la journée. Dans d'autres cas, la rue doit être desservie par deux passages. Des applications réelles de ce problème se posent dans la collecte des ordures et la livraison des journaux. Récemment [Calogiuri et al. \(2019\)](#) ont introduit une procédure basée sur la solution du problème orienté du postier rural (RPP : *Rural Postman Problem*) pour calculer les bornes inférieures et supérieures de la solution optimale du problème du postier rural dépendant du temps (TD-RPP : *Time-Dependent Rural Postman Problem*) sous une loi générale de variation de vitesse.

L'approche courante, pour modéliser un problème de tournées de véhicules sur les arcs avec des contraintes qui dépendent du temps, consiste soit à modifier le graphe, soit à transformer le problème de tournées de véhicule sur les arcs en un problème de tournées de véhicule sur les nœuds ([Aminu et Eglese, 2006](#); [Mullaseril, 1997](#); [Wøhlk, 2005](#); [Vansteenwegen et al., 2010](#)). Sans la modification de graphe ou la transformation de nœud, il est difficile d'associer une heure de début et de fin unique à un arc ou une arrête ([Mullaseril, 1997](#)). Sauf dans [Lystlund et Wøhlk \(2011\)](#), [Wang et Wen \(2002\)](#), [Sun et al. \(2015\)](#) et [Kayacı Çodur et Yilmaz \(2018\)](#),

ils ont utilisé le graphe originale.

Tableau 2.2 – Caractéristiques des modèles de tournées des véhicules dépendantes du temps

Auteurs	Nom du problème	La dépendance du temps	Méthode de résolution	Type de tournées
Malandraki et Daskin (1992)	VRP dépendant du temps	Le temps du trajet	Heuristiques et branch-and-cut	VRP
Jung et Haghani (2001)	Algorithme génétique pour VRP dépendant du temps	Les fenêtres du temps	Algorithme génétique	VRP
Ichoua et al. (2003)	Répartition des véhicules avec temps du trajet dépendant du temps	Le temps du trajet	Recherche taboue	VRP
Tagmouti et al. (2007)	CARP avec coût de service dépendant du temps	Le coût	Génération de colonnes	ARP
Donati et al. (2008)	VRP dépendant du temps	Le temps du trajet	Système de colonies multi-fourmis	VRP
Tagmouti et al. (2010)	CARP avec cout de service dépendant du temps	Le coût	Descente à voisinage variable	ARP
Tagmouti et al. (2011)	Dynamique CARP avec cout de service dépendant du temps	Le coût	descente à voisinage variable	ARP
Balseiro et al. (2011)	Algorithme de colonies de fourmis pour le TDVRPTW	Les fenêtres du temps	Colonies de fourmis	VRP
Sun et al. (2011)	CPP dépendant du temps	Le temps du trajet	branch-and-Bound	ARP
Tan et Sun (2011)	RPP dépendant du temps	Le temps du trajet	branch-and-cut	ARP
Dabia et al. (2013)	branch-and-price pour le TDVRPTW	Les fenêtres du temps	Génération de colonnes	VRP
Black et al. (2013)	Collection des prix pour ARP	Le temps du trajet	Recherche de voisinage variable	ARP
Sun et al. (2015)	CPP dépendant du temps	Le temps du trajet	branch-and-cut	ARP
Victoria et al. (2016)	CVRP avec demande dépendant du temps appliquée à la logistique humanitaire	La demande	Génération de colonnes	VRP
Nossack et al. (2017)	WRRP avec une option de zigzag dépendante du temps	L'option de zigzag	branch-and-cut	ARP
Kayacı Çodur et Yılmaz (2018)	HCPP dépendant du temps	Le temps du trajet	Algorithme génétique avec recuit simulé	ARP
Calogiuri et al. (2019)	RPP dépendant du temps	Le temps du trajet	branch-and-Bound	ARP

2.7 Problème de tournées de véhicules robuste

Le problème de tournées de véhicules robuste (RVRP : *Robust Vehicle Routing Problem*) a été introduit par [Sungur et al. \(2008\)](#) qui a montré la pertinence d'utiliser l'approche robuste. Pour gérer l'incertitude de la demande, ils ont utilisé trois types d'ensembles bornés (coque convexe, boîte et ellipsoïdale). Une application réelle pour l'optimisation robuste a été introduite plus tard dans [Sungur et al. \(2010\)](#) : le problème de livraison de courrier, une variante du problème des tournées de véhicules avec de fenêtres de temps (VRPTW : *Vehicle Routing Problem with Time Window*) où les temps de service sont incertains. [Ordóñez \(2010\)](#) a prolongé les travaux précédents de [Sungur et al. \(2008\)](#) en considérant plusieurs sources d'incertitude (coûts, demande, temps, clients) et la corrélation entre les coefficients incertains. Pour cela ils ont présenté différents modèles robustes.

2.7.1 Méthodes exactes pour le problème de tournées de véhicule robuste

Plusieurs méthodes exactes ont été appliquées pour aborder le RVRP. L'approche de génération de colonnes a été utilisée pour la première fois pour une optimisation robuste par [Lee et al. \(2012\)](#) du problème de tournées de véhicule avec des délais (VRPD : *Vehicle Routing Problem with Deadlines*). L'incertitude de la demande et du temps de trajet a été encapsulée dans le sous-problème appelé le problème robuste de chemin le plus court avec des contraintes de ressources (RSPPRC : *Robust Shortest Path Problem with Resource Constraints*). L'approche du plan de coupe a été introduite pour le VRPTW par [Agra et al. \(2013\)](#) avec une incertitude de temps de déplacement qui était une extension des travaux précédents [Agra et al. \(2012\)](#). L'approche de branch-and-cut a été introduite par [Sungur et al. \(2008\)](#) et [Noorizadegan et al. \(2012\)](#) et améliorée en ajoutant des inégalités de capacité robustes par [Gounaris et al. \(2013\)](#), pour le problème de tournées de véhicules avec capacité (CVRP : *Capacitated Vehicle Routing Problem*) sous incertitude de la demande. [Lu et Gzara \(2019\)](#) ont proposé l'algorithme de branch-and-price-and-cut pour le CVRTW (CVRTW : *Capacitated Vehicle Routing problem with Time Window*) sous incertitude de la demande. Ils ont également proposé une nouvelle procédure de séparation pour résoudre un problème robuste de sac à dos (*bin packing*). Toutes ces études sont basées sur une formulation de dualisation

pour obtenir le modèle robuste correspondant. [Jaillet *et al.* \(2016\)](#) et [Adulyasak et Jaillet \(2016\)](#) ont proposé un nouveau cadre d’optimisation robuste basé sur quelques informations des distributions de probabilité des variables aléatoires décrites par un intervalle et une valeur moyenne. De plus, ils ont proposé un nouveau critère, appelé l’indice de violation des exigences, qui évalue le risque de violation d’une solution. Récemment, [Munari *et al.* \(2019\)](#) ont développé une nouvelle formulation d’optimisation compacte robuste basée sur une programmation dynamique, au lieu d’utiliser un schéma de dualisation.

2.7.2 Méthodes heuristiques pour le problème de tournées de véhicules robuste

Les approches heuristiques ont également joué un rôle considérable dans le RVRP, [Moghadam *et al.* \(2012\)](#) ont utilisé l’optimisation par essaims de particules (Particle Swarm Optimization) pour le CVRP avec une incertitude de demande et les résultats sont comparés à ceux obtenus par [Sungur *et al.* \(2008\)](#). Une métaheuristique d’optimisation des colonies de fourmis (*ant colony optimization*) a été appliquée au CVRP avec des coûts de voyage incertains [Toklu *et al.* \(2013a,b\)](#). [Cao *et al.* \(2014\)](#) ont utilisé l’algorithme d’évolution différentielle pour le problème de tournées de véhicules ouvertes avec demandes incertaines, et quatre stratégies robustes ont été proposées pour gérer la demande incertaine. [Gounaris *et al.* \(2016\)](#) ont étendu leurs travaux antérieurs en utilisant la métaheuristique de programmation de mémoire adaptative (*adaptive memory programming*) au lieu de l’algorithme exact développé précédemment. [Solano-Charris *et al.* \(2015\)](#) ont utilisé une métaheuristique basée sur la recherche locale avec une approche différente pour gérer l’incertitude de façon différente de l’ensemble borné ou de l’ensemble budgétaire. Ils ont proposé un ensemble de scénarios discrets pour le CVRP avec des coûts d’arc incertains en utilisant le critère lexicographique min–max. Le critère min–max sur le coût du déplacement a également été proposé par [Bederina et Hifi \(2018\)](#) en utilisant un algorithme évolutif multiobjectif hybride. [Braaten *et al.* \(2017\)](#) a proposé une variante de recherche adaptative de grand voisinage basée sur la programmation par optimisation pour le VRPTW avec temps de parcours incertains. [De La Vega *et al.* \(2019\)](#) a étendu l’heuristique de Solomon ([Solomon, 1987](#)) pour résoudre le problème de tournées des véhicules avec fenêtres de temps et plusieurs livreurs (VRPTWMD : *Vehicle Routing Problem with Time Window and Multiple Deliverymen*) avec demande incertaine. [Hu *et al.* \(2018\)](#) ont

conçu un algorithme en deux étapes basées sur une heuristique de recherche à voisinage variable adaptative modifiée pour le VRPTW sous une incertitude de demande et de temps de trajet basé sur des ensembles d'incertitude dépendant de l'itinéraire. Récemment [Subramanyam et al. \(2020\)](#) ont développé une recherche locale robuste pour le problème classique de tournées des véhicules hétérogènes (HVRP : *Heterogeneous Vehicle Routing Problem*) avec une incertitude de la demande. La recherche locale robuste a été intégrée dans deux méta-heuristiques (recherche locale itérée et programmation de mémoire adaptative) et testée sur cinq classes populaires d'ensembles d'incertitude (ensembles budgétaires, modèles factoriels, ensemble ellipsoïdes, ensembles à contrainte de cardinalité, ensembles discrets). La qualité des solutions métaheuristiques est quantifiée à l'aide d'un algorithme de branch-and-cut qui fournit des bornes inférieures sur la valeur optimale.

2.7.3 Problème de tournées de véhicules robuste sur les arcs

Le problème de tournées de véhicules robuste sur les arcs n'a été abordé que dans deux articles : [Chen et al. \(2016\)](#) a présenté le problème de tournées de véhicules sur les arcs avec une incertitude de temps de service de surveillance quotidienne du réseaux routière et en comparant l'approche d'optimisation robuste à la méthode de programmation par contrainte de chance en utilisant un algorithme de branch-and-cut . Un problème de tournées de véhicules sur les arcs périodique avec contrainte de capacité (PCARP : *Periodic Capacitated Arc Routing Problem*) avec une incertitude de demande a été proposé par [Tirkolaei et al. \(2018\)](#) pour la gestion des déchets, et résolu en utilisant un algorithme amélioré de recuit simulé hybride.

Tableau 2.3 – Caractéristiques des modèles de tournées des véhicules robustes

Auteurs	Domaine d'application	Type de problème	Type d'incertitude	Méthode de résolution	Formulation de l'incertitude
Sungur et al. (2008)	–	CVRP	Demande	Branch-and-cut	Ensemble borné avec dualisation
Sungur et al. (2010)	Livraison du courrier	VRPTW	Temps de service	Recherche taboue	Ensemble borné
Ordonez (2010)	–	CVRP	Couts Demande Temps de trajet Clients	–	Ensemble borné avec dualisation
Lee et al. (2012)	–	CVRPD	Demande Temps de trajet	Branch-and-price	Ensemble budgétaire avec dualisation
Noorizadegan et al. (2012)	–	CVRP hétérogène	Demande	Branch-and-cut	Ensemble borné avec dualisation
Moghaddam et al. (2012)	–	CVRP	Demande	Optimisation des essaims de particules	Ensemble d'intervalle
Agra et al. (2012)	Transport maritime	VRPTW	Temps de trajet	Branch-and-bound	Ensemble borné avec dualisation
Agra et al. (2013)	Transport maritime	VRPTW	Temps de trajet	Branch-and-cut	Ensemble borné avec dualisation
Gounaris et al. (2013)	–	CVRP	Demande	Branch-and-cut	Ensemble borné avec dualisation
Toklu et al. (2013a,b)	–	CVRP	Temps de trajet	Optimisation des colonies de fourmis	Ensemble budgétaire
Cao et al. (2014)	Livraison des biens	VRP ouverts	Demande	Algorithme d'évolution différentielle	Ensemble borné
Solano-Charris et al. (2015)	–	CVRP	Couts de trajet	Métaheuristique basée sur la recherche locale	Scénarios discrets
Gounaris et al. (2016)	–	CVRP	Demande	Programmation de mémoire adaptative	Ensemble budgétaire
Chen et al. (2016)	Entretien des routes	ARP	Temps de service	Branch-and-cut	Ensemble budgétaire avec dualisation
Adulyasak et Jaillet (2016)	–	VRP avec délais	Temps de trajet	Branch-and-cut	un intervalle et une valeur moyenne avec dualisation
De La Vega et al. (2019)	Ramassage ou livraison des biens	VRPTWMD	Demande	Heuristique constructive	Ensemble budgétaire
Tirkolaee et al. (2018)	Gestion des déchets	PCARP	Demande	Algorithme amélioré de recuit simulé hybride	Ensemble budgétaire
Bederina et Hifi (2018)	–	VRP	Couts de trajet	Algorithme évolutif multiobjectif hybride	Scénarios discrets
Hu et al. (2018)	–	VRPTW	Demande Temps de trajet	Recherche de voisinage à variable adaptative	Ensembles d'incertitude dépendant de l'itinéraire
Lu et Gzara (2019)	Livraison de carburant	CVRTW	Demande	Branch-and-price-and-cut	Ensemble budgétaire avec dualisation
Munari et al. (2019)	–	CVRTW	Demande Temps de trajet	Branch-price-and-cut	Ensemble budgétaire basée sur une programmation dynamique
Subramanyam et al. (2020)	–	HVRP	Demande	Recherche locale robuste	Ensembles budgétaires Modèles factoriels Ensembles ellipsoïdes Ensembles de cardinalité Ensembles discrets

CHAPITRE 3 MÉTHODOLOGIE

Ce chapitre présente la méthodologie utilisée dans cette thèse. Puisque cette thèse traite une application réelle, l'utilisation des données était nécessaire pour mener cette recherche. La première étape a été la collecte et la préparation des données brutes provenant de différentes sources. La deuxième étape a consisté à développer un modèle de prédiction de quantités de sel et d'abrasif en se basant sur une approche d'apprentissage machine. La troisième étape (modélisation dépendante du temps) et la quatrième étape (modélisation robuste) ont consisté à optimiser les opérations d'épandage selon les résultats du modèle de prédiction. La figure 3.1 résume la méthodologie de cette thèse.

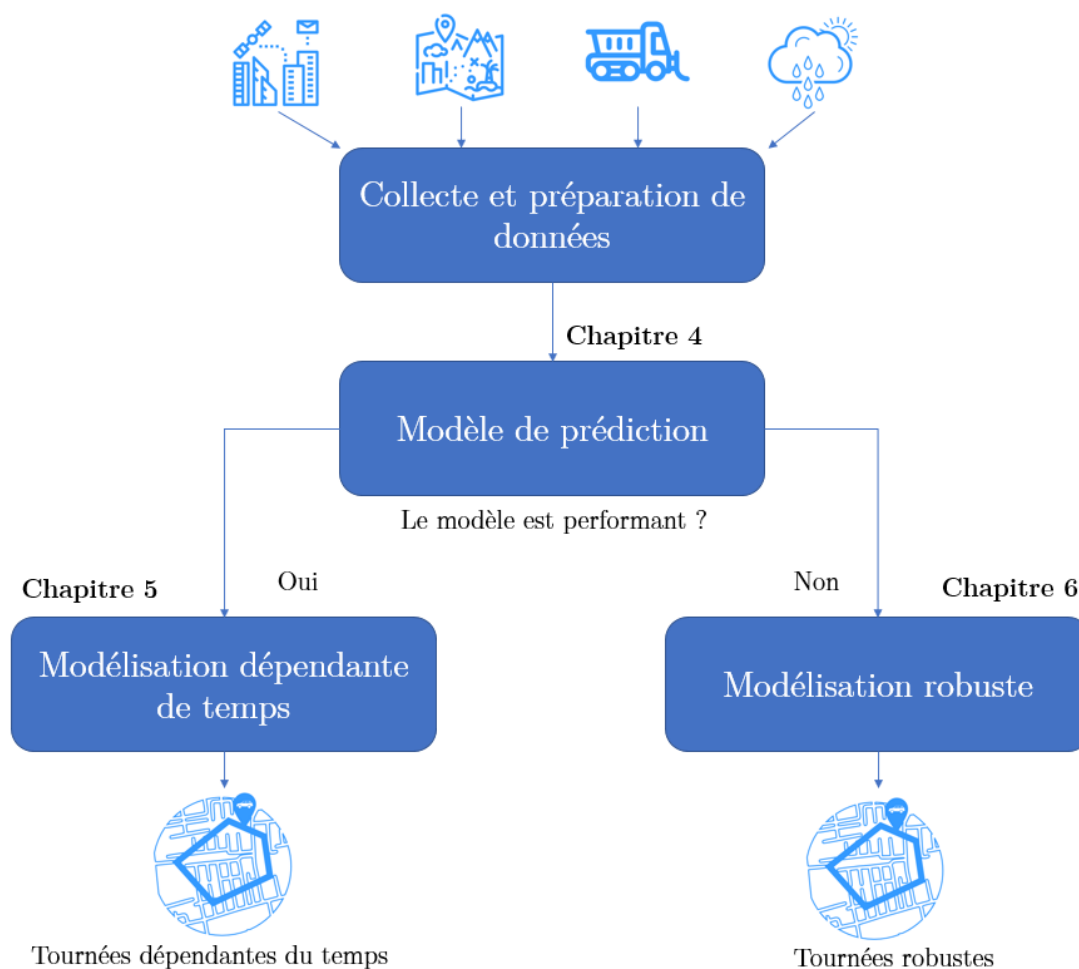


Figure 3.1 Méthodologie générale

3.1 La collecte et préparation de données

Nous avons collecté 9 ensembles de données de différentes sources : données de télémétrie, météorologiques, géomatiques; le tableau 3.1 présente les informations pour chaque ensemble de données. Ces données ont été importées dans une base de données PostgreSQL (<http://postgresql.org/>), et assignées à chaque segment de rue en utilisant un logiciel SIG (*Système d'information géographique*) et les coordonnées géographiques. Deux autres outils sont utilisés : PostGIS (<http://postgis.net/>), une extension de PostgreSQL qui permettent la manipulation d'informations géographiques sous forme géométrique, et QGIS (<http://qgis.net/>) pour la visualisation et l'analyse géographique. La préparation des données a permis de transformer les réseaux routiers en des réseaux mathématiques en utilisant Pgrouting (<http://pgrouting.org/>) dans le but d'appliquer les algorithmes d'optimisation. Cette transformation était compliquée due aux erreurs de topologie (absence d'intersection, des arcs pendants, des pseudo nœuds, des îlots isolés...). De nombreuses éditions de données ont donc été effectuées.

3.2 Modèle de prédiction

Le développement de modèle de prédiction est basé sur une approche d'apprentissage machine présenté à la figure 3.2. La démarche commence par l'extraction des variables (*feature engineering*) qui consiste à rechercher de nouvelles variables pour améliorer le modèle. La deuxième étape consiste à sélectionner des variables afin de choisir les plus pertinentes. La technique Lasso aléatoire a été utilisée pour sélectionner ces variables les plus influentes. L'étape suivante est la phase d'entraînement et de test; cette phase utilise la validation croisée et le réglage hyperparamétrique sur l'ensemble d'entraînement (80% de données) pour comparer différents algorithmes d'apprentissage automatique (les forêts aléatoires, les arbres extrêmement aléatoires, les réseaux de neurones artificiels, *Adaboost*, *Gradient Boosting Machine* et *XGBoost*). Le réglage de l'hyperparamètre a été fait à l'aide de la méthode de recherche aléatoire (Bergstra et Bengio, 2012). Les données de test (20% de données) sont utilisées uniquement pour l'analyse des erreurs et l'évaluation des performances, puis le meilleur modèle est choisi pour une étude plus approfondie. Le langage Python est utilisé avec plusieurs bi-

Tableau 3.1 – Caractéristiques des données utilisées dans la thèse

Jeu de données	Source	Type de données	# de lignes	# de champs
Données camions Granby	Capteurs des Camions d'épandage de Granby	Données de télémétrie	29 millions	24
Données camions SJSR	Capteurs des camions d'épandage de SJSR	Données de télémétrie	9 millions	26
Données météorologiques Granby-1	www.climat.meteo.gc.ca	Données météorologiques horaires	4344	12
Données météorologiques Granby-2	Ministère du Développement durable, de l'Environnement et de la Lutte contre les changements climatiques	Données météorologiques horaires	4344	12
Données météorologiques Granby-3	www.meteoblue.com	Données météorologiques simulation horaires	4344	15
Données météorologiques SJSR	www.climat.meteo.gc.ca	Données météorologiques horaires	7249	8
Données météorologiques SJSR-2	Campbell Scientific	Données météorologiques 15 minutes	29116	7
Réseau routier Granby	La ville de Granby	Couche géographique	3153	19
Réseau routier SJSR	La ville de SJSR	Couche géographique	6057	8
Modèle numérique d'élévation	Gouvernement du Canada	DEM : Modèle numérique d'élévation	–	–

bibliothèques (Sklern, XGBoost, Theano, Keras). Enfin, dans l'étape d'évaluation du modèle, le modèle est validé itérativement. S'il n'est pas satisfaisant, un retour à l'étape d'extraction des variables est fait. Pour plus de détails, se référer à l'article du chapitre 4.

3.3 La modélisation dépendante du temps

Le modèle de prédiction décrit en 3.2 prédit la quantité à épandre pour chaque heure. Cette information horaire nous permet de proposer un modèle avec une demande dépendante du temps. Une formulation mathématique est proposée, codée avec Julia ([Dunning et al., 2017](#)) et résolue avec CPLEX 12.9 (www.cplex.com) pour les petites instances. La métaheuristique SISR modifiée est développée en C++ pour résoudre les grande instances. La métaheuristique a été inspirée de [Christiaens et Vanden Berghe \(2020\)](#). Toutefois, certaines modifications importantes ont été réalisées pour considérer la demande dépendante du temps, et aussi pour considérer l'évaluation des coûts et de la capacité en utilisant la programmation dynamique.

Pour plus de détails, se référer à l'article du chapitre 5.

3.4 La modélisation robuste

La modélisation dépendante du temps est basée sur l'hypothèse que le modèle de prédiction est fiable, c'est-à-dire qu'on peut prédire exactement les quantités à épandre. Dans le cas contraire, l'optimisation robuste s'avère la plus appropriée pour résoudre les problèmes avec incertitude. Nous avons modélisé cet incertitude avec une formulation robuste basée sur la dualisation (Ben-Tal et Nemirovski, 1998; Bertsimas et Sim, 2003), codée avec Julia et résolue avec CPLEX 12.9, pour résoudre les petites instances. La métaheuristique SISRs robuste qui considère la demande incertaine est développée en C++ pour résoudre les cas réels. Pour plus de détails, se référer à l'article du chapitre 6.

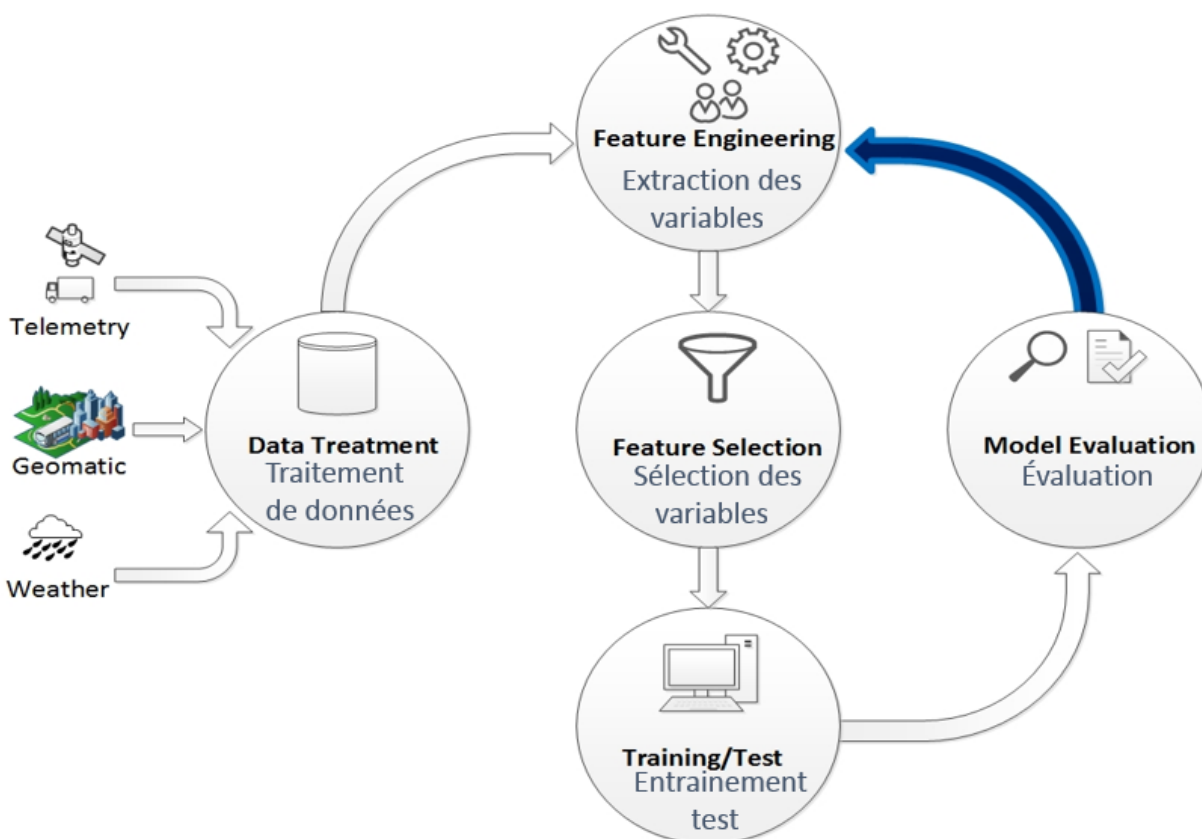


Figure 3.2 L'approche apprentissage machine utilisée au chapitre 4

CHAPITRE 4 ARTICLE 1: STREET-SEGMENT-BASED SALT AND ABRASIVE PREDICTION FOR WINTER MAINTENANCE USING MACHINE LEARNING AND GIS

C. Ahabchane, M. Trépanier et A. Langevin

Cet article a été publié dans [Transactions in GIS](#), (2019), 23(1), p. 48-69.

4.1 Abstract

Salting operations are essential but expensive in northern winters. We aim to predict the quantity of salt and abrasive needed for a specific road segment for each hour. An estimate of the quantity allows managers to better manage the loads in the trucks and to propose optimal vehicle routes based on the forecast, which will allow them to optimize costs. This article uses machine-learning techniques based on truck telemetry data, weather conditions, and segment attributes. Geographic information systems (GIS) allow us to exploit the street-network characteristics, which were ignored by previous prediction models. The results show that the XGBoost method performs better than other techniques ($R^2 = 0.83$).

4.2 Introduction

In northern regions, one of the most important components of winter road maintenance is the spreading of salt and abrasives, which we refer to as salting. These operations are carried out using appropriately equipped trucks, and they are extremely costly. In 2013, the Quebec transportation ministry estimated its spending at C\$260 million. In the same year, C\$786 million was spent by all the cities in the province ([Normandin, 2015](#)), and the city of Montreal alone spent C\$159 million ([Ville de Montréal, 2016](#)).

Given the cost, municipalities use technologies that can reduce the associated costs, such as road weather information systems and automated vehicle-tracking systems ([Fu et al., 2009](#)). Telemetry tracking devices are indispensable in fleet management. They collect in real time vast quantities of data, which can then be exploited to allow a better use of resources

(Quirion-Blais *et al.*, 2017)

This study considers the short-term prediction of the quantities of salt and abrasives to spread. The prediction is based on the geometric characteristics of the street grid, previous winter maintenance telemetry data, and weather data. The model is intended to be used just before or during a precipitation episode. We pay careful attention to the data processing and the extraction of the variables: “Much of the actual effort in deploying machine learning algorithms goes into the design of feature extraction, preprocessing and data transformations” (Bengio *et al.*, 2013).

The prediction process has two phases. In the first phase, machine-learning tools are applied to select the most significant variables using the randomized Lasso method (stability selection). In the second phase, several prediction models are tested: random forests, AdaBoost, extremely randomized trees, the gradient boosting machine, XGBoost, and artificial neural networks (ANNs).

Section 4.3 presents a review of relevant work on prediction models for winter road maintenance. In Section 4.4 the methodology of this research is explained. The article first processes the collected telemetry, weather, and geomatics data. Then feature engineering is performed (i.e. new variables to improve the prediction model are developed). In the next step, feature selection is used to reduce the number of variables. The training and testing step finds the best machine-learning algorithm, which is studied in detail in the model evaluation phase. Section 4.5 presents a case study for the city of Granby, Canada. Finally, Section 4.6 provides the results and a discussion.

4.3 Literature review

This section discusses prediction models for winter road maintenance. First the models, the targets or outputs used, and the results are presented. Then the application of machine learning in these problems is reviewed; it has not been widely used.

4.3.1 Prediction models for winter maintenance

Blackburn *et al.* (2004); Akin et Shi (2010); Druschel (2012); Hossain *et al.* (2016) have investigated the factors influencing snow melting: pavement temperature, traffic, snow thickness, solar radiation, type of chemical spread, pavement type, wind conditions, roadway shade, and application rate. Different prediction models use different performance indicators.

- **Bare pavement regain time (BPRT):** Hossain *et al.* (2014) used linear regression to predict the snow-melting performance of road salt and to identify the contributing factors, under different application rates and weather conditions. For the performance indicator they used BPRT, which is the time between salt application and the point where 80% of the pavement is bare. Pavement temperature, salt application rates, and the amount of snow are found to be important factors. The results showed the significance of the variables, but the prediction model was not efficient ($R^2 = 0.39$). The tests were conducted in a university parking lot and did not consider variables such as real traffic and street characteristics (type, length, turns, slopes, etc.).
- **Application rate:** Hossain *et al.* (2015) developed a physical model to predict the optimal application rate for parking lots and sidewalks. They used a theoretical model based on thermodynamics, chemistry, and a snow-melting model. The model was calibrated using data from field tests, but the results had poor estimation accuracy.
- **Contaminant layer depths:** Feng et Fu (2015) introduced time-series modeling approaches such as univariate and multivariate integrated autoregressive moving average (ARIMA) and autoregressive moving averages with external variables (ARIMAX) to predict the road surface conditions: surface temperature and contaminant (snow, ice, water) layer depths. The prediction model relies on previous winter operations and historical data. The variables are traffic volume, weather information, plowing record, and salting intervals. The model was good: R^2 between 0.85 and 0.97.
- **Salt durability:** Blomqvist *et al.* (2011) combined an empirical model developed by the Swedish National Road and Transport Research Institute with data from the Danish road authority. The model is based on a mathematical function that predicts

the durability of a given salt dose as a function of the traffic conditions and the moisture of the road surface. The model was fairly good: $R^2 = 0.89$.

Similarly, [İkiz et Galip \(2016\)](#) estimated the decrease in the salt concentration over time. They also used a mathematical function based on traffic and time. The study was first conducted in the field to determine the influencing factors, and then laboratory tests were carried out to correct errors made during the field tests. The model has R^2 between 0.82 and 0.99.

- **Snowmelt volume:** [Trenouth et al. \(2015\)](#) introduced the modified temperature index model (MTIM) to predict the volume of snowmelt. MTIM uses two meteorological characteristics (air and base temperature) as well as plowing and salting information. It is a modification of the temperature index model (TIM); the latter ignores the plowing and salting operations, which makes it inconvenient for urban applications (Semádeni-Davies, 2000). The authors calibrated and validated the model at three sites, showing that MTIM can be used to optimize salting rates for given weather conditions. The model was good: $R^2 = 0.99$.

4.3.2 Machine learning applied to winter maintenance

Machine learning has been used in various fields, including winter road maintenance. [Shi et Akin \(2012\)](#) analyzed the performance and corrosivity of chloride deicer solutions using ANNs. ANNs have been used to explore the correlation between electrochemical and gravimetric corrosion ($R^2 = 0.84$), between deicer composition and deicer corrosivity ($R^2 = 0.90$), and between deicer composition and deicer performance ($R^2 = 0.98$). This work was further developed by [Shi et al. \(2013\)](#), who investigated the baseline performance data of typical chloride deicers used on highways. For this new study, they developed three models based on ANNs: the first model was established to correlate the electrochemical corrosion data with those from the gravimetric test method; two additional models were used to quantify the correlations between the deicer composition (type, chloride and inhibitor concentrations, pH, and electrical conductivity) and their corrosivity and performance, respectively. These models can be used to improve the design of more powerful and less corrosive deicers.

Ye *et al.* (2009) used ANNs to evaluate the effects of weather information on winter maintenance costs. The sensitivity analysis shows that the lane miles of roadways maintained by maintenance units have the most important impact on winter maintenance costs; the results show that the increased use and accuracy of weather information can reduce costs. The prediction model has $R^2 = 0.93$. Ye *et al.* (2014) used ANNs to model the relationship between road safety in terms of the number of winter car accidents and other features such as chemical usage, traffic volume, length, and weather conditions. A sensitivity analysis was conducted to identify significant features and investigate their effects. The results revealed that winter chemicals improved road safety, and the benefits outweighed the costs. The prediction model has R^2 between 0.62 and 0.75.

Other algorithms, different from ANNs, have also been used. Yun *et al.* (2018) used the weighted KNN (K-nearest neighbors) to predict the road surface temperature, which is considered crucial to develop winter maintenance strategies in traffic safety; this study evaluates the feasibility of predicting RST based on climatological information (i.e. humidity and air temperature) and ambient temperature that can be collected via probe vehicles. The results show that overall performances were satisfactory based on the AAE (average absolute error) measurement. Deep learning was used for the first time in winter road maintenance by Pan *et al.* (2018). The deep-learning model is based on convolutional neural networks (CNNs), in order to classify road surface conditions based on images from smartphones. The results show that the proposed model outperforms ANNs, support vector machines, and random forests with an accuracy exceeding 78%.

This research is innovative in three respects. Firstly, to the best of our knowledge, this study is the first to use GIS features in order to predict salt and abrasives. In fact, previous work in the literature established their studies in a laboratory or on at most three segments. However, this study takes into consideration almost the whole city, because we believe that each segment has its own characteristics which influence the spreading quantities. Secondly, this study uses telemetry data in the model because human factors influence the prediction: speed, driver behavior, number of passages. Finally, the proposed model predicts the quantities for each hour and for each segment; this has never been proposed in the literature.

4.4 Methodology

The article uses the methodology presented in Figure 4.1. First the GPS points (telemetry) are allocated and the data is cleaned in the data treatment step. The feature engineering step looks for new variables to improve the model, and in the feature selection step chooses the most relevant variables. The training and testing step investigates various machine-learning algorithms and selects the best model. Finally, in the model evaluation step the model is iteratively validated. If it is not satisfactory, the blue arrow is followed and returns to the feature engineering step. The five steps are explained in detail in the following subsections.

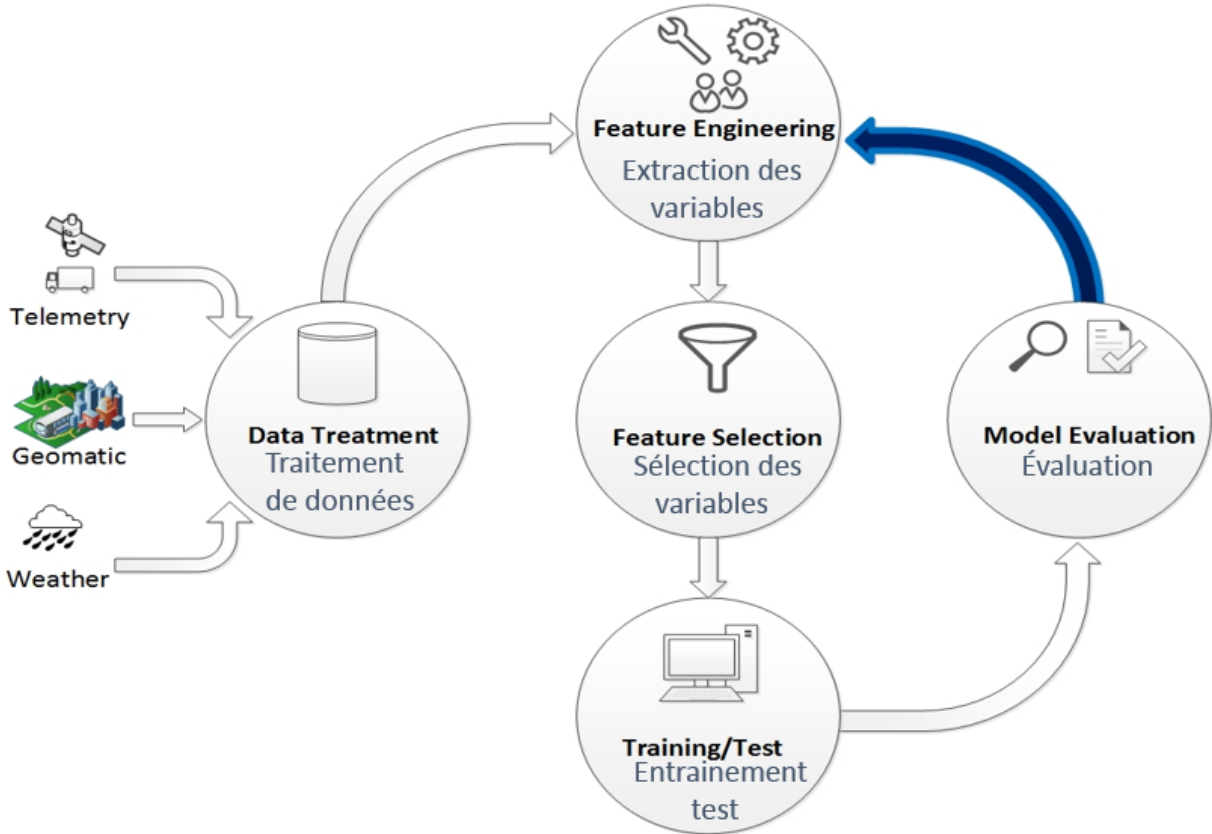


Figure 4.1 Diagram summarizing the methodology applied in this research

4.4.1 Data treatment

Three types of data are collected: geomatics for the city (street length, number of lanes, street segment type), telemetry data from the truck sensors (date and time, truck speed,

GPS traces, salting quantity), and weather data (air temperature, snow and rain amounts, wind speed, dew point) from the Ministry of Sustainable Development, the Environment and the Fight against Climate Change as well as the Meteorological Service of Canada website (<https://climat.meteo.gc.ca>). This data is detailed in Section 4.5, where the case study is presented.

Map matching

Before data cleaning, the streets were split into segments, following the street network segmentation provided by the city of Granby (Figure 4.2). Street segments are bounded by each intersection, sharp turn (name changes), and in long stretches streets are sometimes cut at some points like bridges. Then the GPS traces with high frequency (0.5 s between GPS traces) are assigned to the appropriate street segments using map matching. Each GPS trace is associated with the nearest segment within a radius of 20 m; the direction is also considered at intersections and some divided roads. This provides sufficient precision, so we do not need path-based matching. We then compute hourly statistics for each street segment.

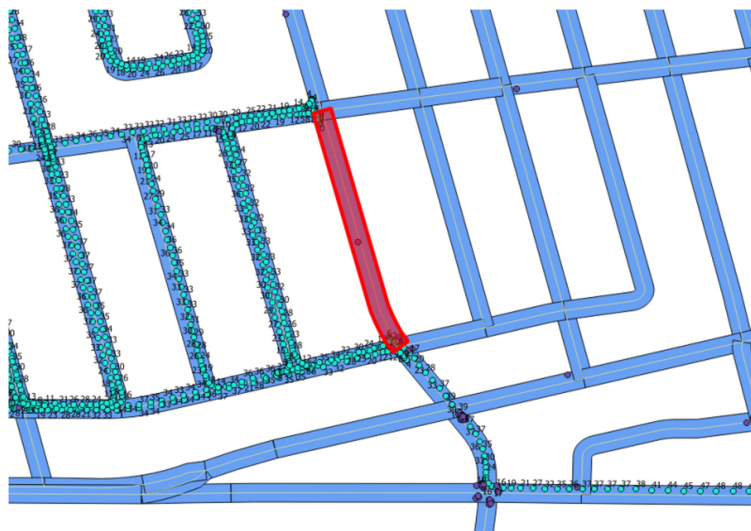


Figure 4.2 Map matching of GPS points

Data cleaning

Data cleaning tries to remove outliers using the following rules:

- The salting ratio (salting quantity by street length) must be greater than 50 kg/km (the minimum required by the Quebec Transportation Ministry). This ensures that the study considers only the salting periods.
- The number of passages (nbp) (i.e. the number of times a truck traverses a street during a given period) must be greater than 0.8 (for more details, see Section 4.4.2). Map matching is approximate: some GPS points are assigned to the wrong street segment or have an incorrect coordinate. In Figure 4.2, the segment in red does not have enough GPS points to identify a passage, so it is eliminated from the modeling step.
- All consecutive GPS traces with zero speed for more than 2 min are eliminated. This criterion was introduced to eliminate long stops, idling, and parking.

4.4.2 Feature engineering

Feature engineering is the important task of finding the relevant features for the model. It consumes the majority of the effort in machine-learning projects: the use of intuition and creativity turns out to be more important than the machine-learning techniques (Domingos, 2012). “Feature engineering is a way to take advantage of human intelligence and prior knowledge to compensate for the weakness of current learning algorithms” (Bengio *et al.*, 2013). This phase consists of identifying new weather, geomatics, and telemetry features from the data collected and GIS manipulation.

Weather

Weather is an important factor in our model. In addition to known weather data (temperature, precipitation, pressure, humidity, dew point, solar radiation, wind speed), we looked for other important meteorological features, using our intuition and talking to experts.

- (a) Precipitation accumulation during the last 24 (36) hours:

The amount of salting depends on the amount of accumulated precipitation. Historical

data analysis has shown that snowfalls can last for more than 24 hours, so the snowfall accumulation per hour in the last 24 and 36 hours is added to this model.

(b) Cumulative effect of temperature over 24 hours:

Snow removal experts consulted mentioned that the temperature accumulation plays a significant role in salting operations. A feature has been defined which expresses the accumulated temperature per hour for the last 24 hours by summing the hourly temperature of the last 24 hours.

Geomatics

Previous prediction models ignored the street-network characteristics. The road and street characteristics found in the data are used (segment type, speed limits, segment length) and new features that may influence salting, including turn intensity, slope intensity, and elevation are added. We use a digital elevation model (DEM) with 30 x 30 m spatial resolution and GIS to extract these features.

There are several methods for measuring turn or curvature (Weisstein, 2017). This article developed a simple method to quantify turn intensity. The basic idea is to classify each segment according to the intensity of its turns. Each segment is divided into sub-segments j based on angle variations, and we calculate the sum of the angle gradient by dividing by the sum of the lengths multiplied by a binary variable δ_j to eliminate sub-segments with the same angle. This indicator TI_i for segment i is calculated via:

$$TI_i = \frac{\sum_0^{n_i-1} |\theta_{i,j+1} - \theta_{ij}|}{\sum_0^{n_i} \delta_j d_{ij}} \quad (4.1)$$

where: n_i = the number of subsegments in segment i ;

θ_{ij} = the angle of subsegment j of segment i ;

d_{ij} = the length of subsegment j of segment i ; and

$\delta_j = 1$ if $\theta_{i,j+1} \neq \theta_{ij}$; 0 otherwise.

Our case study has a varied topography and steep slopes, so slope measurement is an important feature in the geomatics. Each segment is divided into sub-segments according to its length in such a way that the sub-segment number increases with the segment length in order to have more precision (for example a segment of 100 m has 5 parts, and a segment of 1,000 m has 20 parts). The slope intensity is measured using a similar formula, based on the elevation of the subsegment center:

$$TI_i = \frac{\sum_0^{n_i-1} |E_{i;j+1} - E_{ij}|}{\sum_0^{n_i} \delta_j d_{ij}} \quad (4.2)$$

where: n_i = the number of subsegments in segment i ;

E_{ij} = the elevation of subsegment j of segment i ;

d_{ij} = the length of subsegment j of segment i ; and

$\delta_j = 1$ if $E_{i;j+1} \neq E_{ij}$; 0 otherwise.

The mean elevation of the center of these parts is used to calculate the height or elevation of the segment and considered as an additional feature.

Telemetry

On-board truck systems generate useful information. [Ahabchane et al. \(2015\)](#) developed performance indicators:

- (a) Salting quantity and frequency for the last 24 and 36 hours:

Some streets are salted several times daily by different trucks. The quantity and frequency of the salting over the last 24 and 36 hours are considered.

- (b) Number of passages:

This indicator represents the number of passages by all trucks, for each street segment (e.g. two passages for one round trip). The calculation is made using the average speed:

$$nbp_i = \frac{\bar{V}_i T_i}{D_i} \quad (4.3)$$

where:

$\bar{V}_i$ = the average speed of segment i ;

T_i = the time spent on segment i ;

D_i = the length of segment i . For details, see (Ahabchane *et al.*, 2015).

(c) Total number of passages:

The number of applications over the winter by street segment is introduced as an indicator of the importance of a street: important streets are salted more often.

(d) Traffic speed:

When vehicles travel quickly, they generate heat by friction between the tires and the pavement (Trenouth *et al.*, 2015). The annual average daily traffic (AADT) is often used to measure traffic volumes, but it is not available in our case study. However, average speed is a reliable indicator of traffic volume (Jie *et al.*, 2011); Tong *et al.* (2009) used travel time and average speed for highway traffic pattern recognition and congestion analysis. For each segment, the average speed for each interval of each day is calculated.

4.4.3 Feature selection

Feature selection is an automated technique for selecting the most informative variables. It is useful because (Guyon *et al.*, 2006):

- it reduces the amount of data, which will reduce the computational time and the training time;
- it improves model performance by removing misleading data and redundant variables; and
- it facilitates visualization and understanding of the model.

There are many feature-selection algorithms. The randomized Lasso is used, also called stability selection (Meinshausen *et al.*, 2010), which overcomes the limitations of the Lasso method (Wang *et al.*, 2011).

4.4.4 Training and testing

Before starting the feature selection, the data set was randomly divided into training data (80%) and testing data (20%). Various machine-learning algorithms are trained using the cross-validation method on the training data. The test data are used only for error analysis and performance evaluation, and then the best model is chosen for further study. The Python language with several libraries is used (Sklern, XGBoost, Theano, Keras).

4.4.5 Machine-learning algorithms

This subsection presents a brief description of the machine-learning techniques used.

(a) Random forest:

The random forest was introduced by [Breiman \(2001\)](#). It is a set of decision trees, and each tree is constructed from a randomly selected training-data subset (bootstrapping). Each tree is trained on a randomly selected subset of features. The average of the classifiers or regressors obtained for each tree is the prediction model ([Breiman, 1996](#)).

(b) Extremely randomized trees:

Extremely randomized trees were developed by [Geurts *et al.* \(2006\)](#). They are a collection of methods that resemble random forests. However, random forests choose the split that gives the best information gain, whereas extremely randomized trees choose the split randomly; this generally reduces the variance of the model and slightly increases the bias.

(c) AdaBoost:

AdaBoost is an algorithm developed by [Freund et Schapire \(1997\)](#) that constructs a strong classifier $f(x)$ from a linear combination of T weak classifiers: $f(x) = \sum_{t=1}^T \alpha_t h_t(x)$, where α_t represents the weight of weak classifier h_t . The algorithm initially sets the weights $\alpha = \frac{1}{T}$ for any t . The first iteration trains a weak classifier on the data, then in each subsequent iteration the next weak classifier is trained on weighted data to reduce

the misclassified examples. Finally, all the classifiers are combined via:

$$f(x) = \sum_{t=1}^T \alpha_t h_t(x) \quad (4.4)$$

where $\alpha_t = \frac{1}{2} \ln\left(\frac{1-e_t}{e_t}\right)$ and e_t = the error function of classifier h_t .

(d) Gradient-boosting machine:

The gradient-boosting machine is a powerful technique developed by [Friedman \(2001\)](#). The learning process is based on the construction of new weak learners (regression or classification tree). At each iteration, gradient descent is used to add a new weak learner to minimize the loss function (squared error for regression and logarithmic loss for classification); this modifies the parameters of the tree ([Natekin et Knoll, 2013](#)).

(e) XGBoost:

XGBoost was developed by [Chen et Guestrin \(2016\)](#). It uses the principle of the gradient-boosting machine but is better and faster. The main difference is the use of regularization terms to avoid over-learning.

(f) ANNs:

ANNs are among the best machine-learning algorithms; the first neural-network concept was developed by [Rosenblatt \(1958\)](#). The algorithm is based on gradient descent by back-propagation. ANNs are formed of several layers; each input is a nonlinear function of the output linear combinations of the previous layer. Gradient descent optimizes the weights of these linear combinations to minimize the prediction error. There are several types of ANNs, including multi-layer perceptrons (MLPs), recurrent neural networks, convolutional neural networks, and Boltzman machines. We used the MLP.

Cross-validation

Cross-validation is a statistical method that evaluates learning algorithms by partitioning the data into two subsets: a training set to train the model and a test set to evaluate the model [Refaeilzadeh et al. \(2009\)](#). We used k -fold cross-validation, where the data is partitioned into

k subsets of equal size. Each subset in turn is considered as the test set and the remaining subsets as the training set. The cross-validation process is repeated k times.

4.4.6 Model evaluation

The model is evaluated using metric functions (R^2 and the mean square error, MSE). If the model is unsatisfactory, the process returns to the feature engineering step. If the model is acceptable, the important variables are determined, and the model will be used to predict the amount of salt for a new data set.

Performance evaluation

Two evaluation criteria or metric functions are used:

$$R^2 = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (4.5)$$

$$MSE = \frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{n} \quad (4.6)$$

where:

y_i = the observed value;

$\hat{y}_i$ = the predicted value;

$\bar{y}$ = the mean of the observed values.

Hyperparameter tuning

Hyperparameters are parameters that are fixed before the training of the model; they are not optimized during the training. For example, in ANNs the numbers of neurons in the input layer and the type of activation function are hyperparameters, and the weights of the neurons are ordinary parameters.

Hyperparameter tuning or optimization aims to find all the hyperparameters that maximize the performance of the model. Hyperparameter tuning is used in both training/testing with the cross-validation method to determine the best model phase and with the selected

algorithm in the model evaluation step.

Variable importance

Variable importance, or feature importance, is mainly known as “sensitivity analysis” in the literature (Wei *et al.*, 2015). It is a method used to determine which features are correlated with the output, and which features contribute to the output variability Hamby (1994). The sensitivity of a feature is the effect that a variation of that feature will have on the output. A higher sensitivity of a feature implies a greater variation of the output and vice-versa (King *et Perera*, 2011).

4.5 Case study

This section begins by presenting our case study and explaining the use of the information system to collect data and to assign GPS traces to street segments. Then the data collected and the data developed during the feature engineering are presented. sectionCase study:
City of Granby, Quebec

This research was conducted in collaboration with the city of Granby in Quebec, Canada (90,000 inhabitants). The network has 661 km of roads, corresponding to 3,280 nodes and 2,250 arcs in the mathematical network. Figure 4.3 shows the Granby road network.

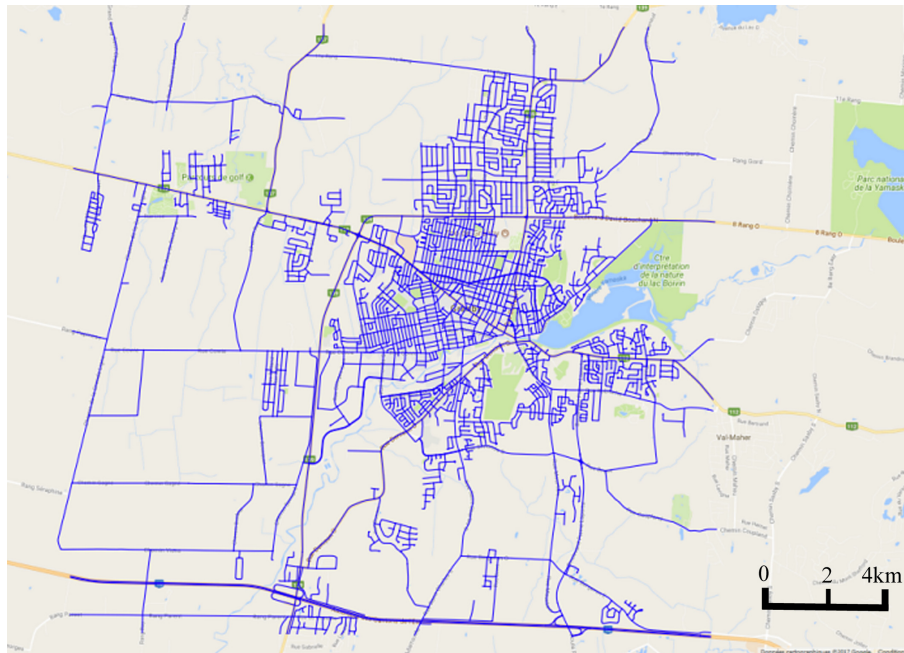


Figure 4.3 Granby road network

The data used was collected by the on-board systems in Granby's 12 salt trucks. These data are recorded each 0.5 s by the various sensors on board. Each record shows the date and time, the geographical position (longitude, latitude), the salting quantity, the speed, and the direction. The data covers the 2014–2015 winter and has 29.1 million records from 12 vehicles. The season began in November 2014 and ended in April 2015 (118 mm of rainfall, 216 cm of snowfall); see Figure 4.4.

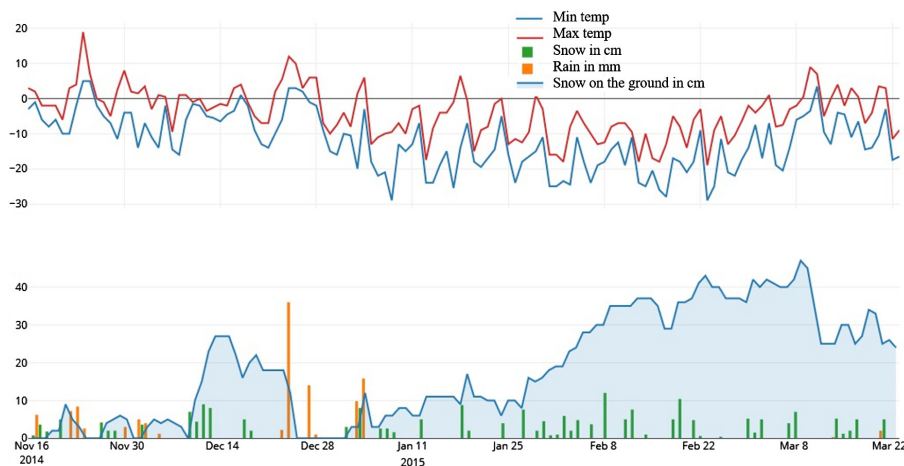


Figure 4.4 Granby weather records, 2014–2015

Each record is mapped to Granby’s territory using its geographic coordinates and GIS. Two more tools are used: PostGIS, an extension of PostgreSQL that allows the manipulation of geographic information as geometries, and QGIS for geographical visualization and analysis.

4.5.1 Data description

This subsection presents a descriptive analysis of the data set and illustrates graphically its impact on the output. The data is first cleaned and aggregated by hour and by truck for each street segment. This results in 155,056 records (road segments x hours of operation) with 46 features and the output variable (salting quantity). Table 1 presents the input data. The model is intended to be used just before or during precipitation episodes, so some variables are based on data from ongoing operations.

Categorical variables

In Figures 4.5 and 4.6, boxplots are presented for the categorical variables that appear to significantly influence the average quantity of salt and abrasive (truck identity and segment type). The boxplots indicate the effect of each variable setting. The salting quantity depends on the segment type and on the vehicle, since the trucks do not serve the same sectors. This confirms our choice of a machine-learning approach.

Table 4.1 – Input variables description (N = numeric, CAT = categorical)

Variable	Type	Description	Min	Max	Average
WEATHER					
Temp	C	hourly air temperature (oC)	-30	3	5
Dewpoint	C	hourly dew point	-34.5	2.6	7.5
HumReal	C	hourly humidity (%)	33	98	83.5
DirWind	C	hourly wind direction (deg)	0	360	190
VitWind	C	hourly wind speed (km/h)	0	28	10
Pressure	C	hourly pressure (kPa)	98	103	100
TotalPre	C	total hourly precipitation (mm)	0	3.5	0.5
Rain	C	hourly rainfall (mm)	0	3.2	0.1
Snow	C	hourly snowfall (cm)	0	3.5	0.32
Totalcloud	C	total cloud cover as percentage of maximum cloud cover	0	100	90
Highcloud	C	percentage of clouds at 18 km at the equator	0	100	28
Mediumcloud	C	percentage of clouds at 10 km at the equator	0	100	52
Lowcloud	C	percentage of clouds at 5 km at the equator	0	100	90
Radiation	C	solar radiation (W)	0	545	51
Accumusnow	C	snow accumulation over 24 h (cm)	0	21	4.6
Accumurain	C	rain accumulation over 24 h (mm)	0	14.8	1.1
Accumusnow36	C	snow accumulation over 36 h (cm)	0	24	6
Accumusnow48	C	snow accumulation over 48 h (cm)	0	27	7
Accumutemp	C	temperature accumulation over 24 h	-552	58	120
GEOMATICS					
Typesegment	CAT	type of street segment (artery, collector, local, etc.): 11 categories			
Typetruck	B	set if trucks are prohibited			
Speedlimit	CAT	speed limit (25, 50, 70, etc.): 5 categories			
Direction	CAT	direction of travel			
Sidewalk	CAT	presence of sidewalk			
Lanenummer	C	number of road segments	1	4	2.05
Width	C	segment width (m)	5.88	24	10.52
Length	C	length of street segment (m)	5.05	3912	174.02
Coating	CAT	type of coating (asphalt, gravel, etc.)			
Busline	B	set if bus line present on segment			
Slope	C	slope intensity	0	0.11	0.015
Turn	C	turn intensity	0	1.17	0.15
Elevation	C	segment center height (m)	68	177.5	115
TELEMETRY					
Totalnbpspread	C	total number of passages	300	3815	805
Traffic	C	hourly average speed for each day and each segment	4	93	29
Vehicleno	CAT	truck identity			
Nbp	C	estimated number of passages	0.8	12	1.74
Quantspread24	C	salting quantity over 24 h (kg)	0	1990	94
Nbspread24	C	number of times segment was salted over 24 h	0	106	9.66
Quantspread36	C	salting quantity over 36 h (kg)	0	2290	115
Nbspread36	C	number of times segment was salted over 36 h	0	135	12
TypeAbrasive	CAT	type of salts and abrasives used: 4 categories			
Day	CAT	day of week			
Hour	CAT	time of day			
Month	CAT	month of year			
Output	C	salting quantity (kg)	1	3170	48

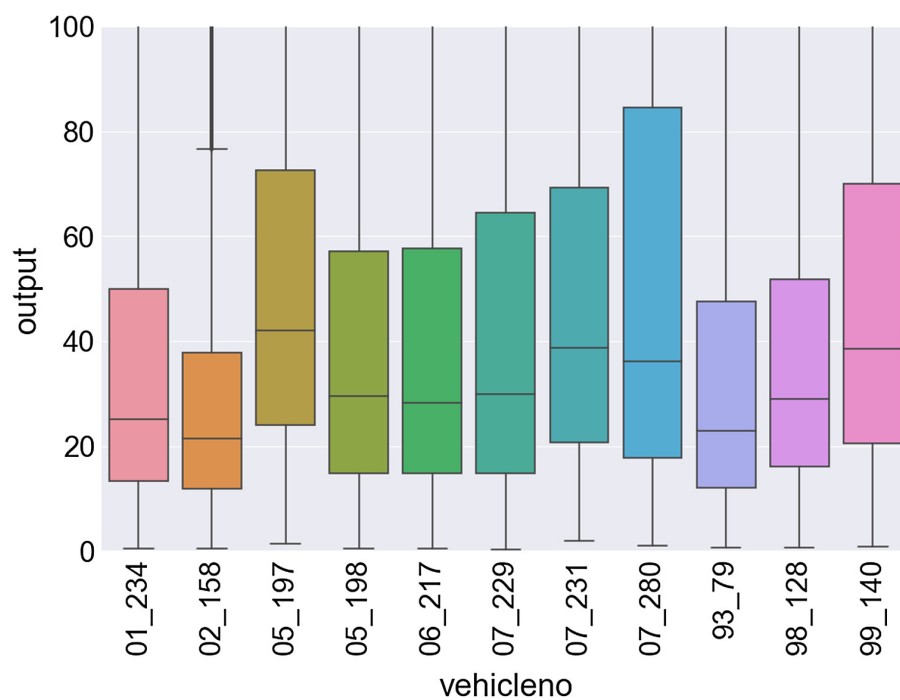


Figure 4.5 Average amount of salt and abrasive by truck (kg)

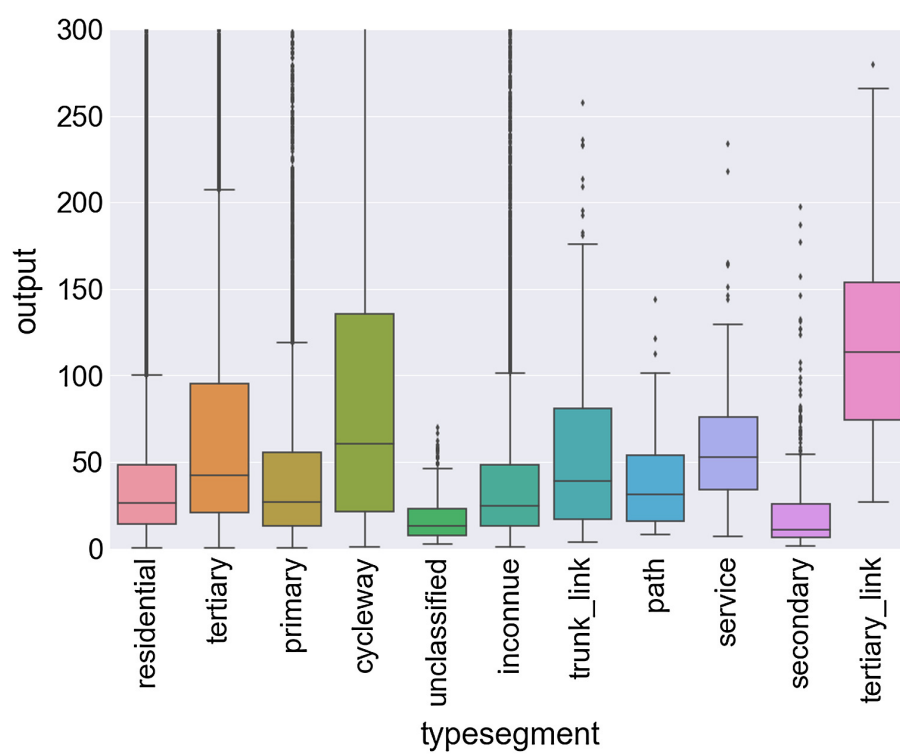


Figure 4.6 Average amount of salt and abrasive by type of segment (kg)

Continuous variables

For a preliminary analysis of the continuous variables, Pearson correlation is used to evaluate the relationship between these variables and the output (Figure 4.7). The variables most correlated with the output (salt and abrasive quantities) are salting quantity over 36 hours and length.

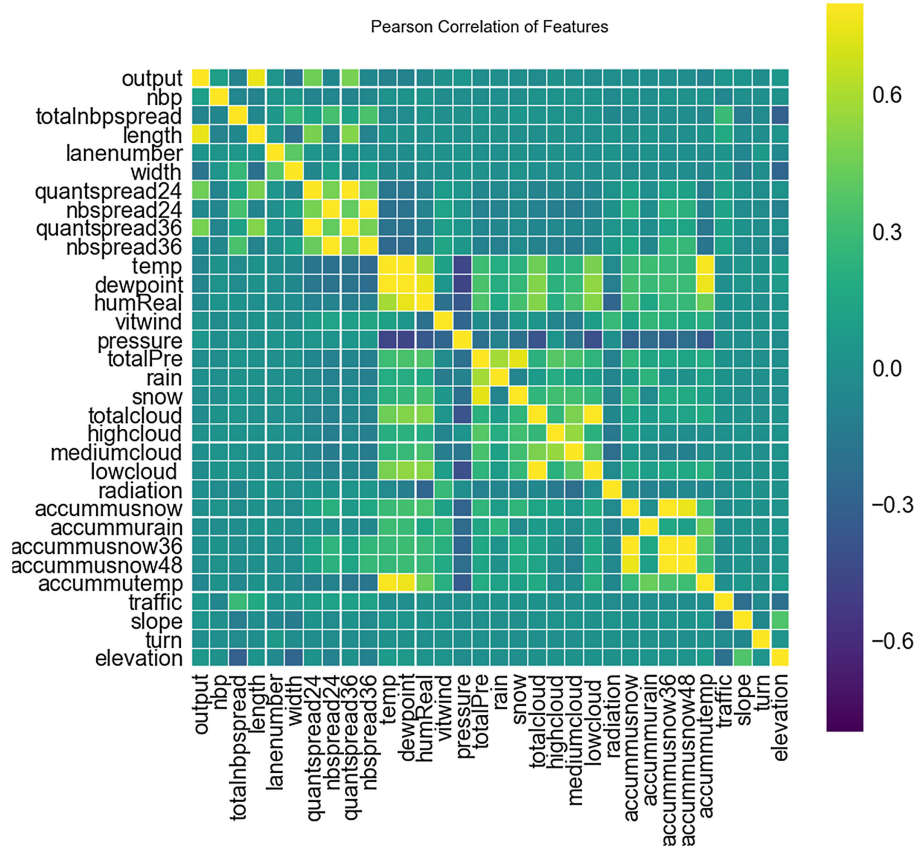


Figure 4.7 Pearson correlation indicates the relationship between the variables and the output

4.5.2 RESULTS AND DISCUSSION

The results of the feature selection are presented in this section; 35 of the 46 features are retrained, which significantly reduces the computational time. We then discuss the results of the training and testing, showing that XGBoost outperforms the other algorithms. Finally, XGBoost is studied in detail and we present an error analysis. 5.1 Feature selection

Figure 4.8 shows the feature ranking with the randomized Lasso method on a scale from 0

to 1. The nine variables removed are wind direction, rain accumulation, rainfall, snowfall, presence of bus line, presence of sidewalk, truck prohibition indicator, type of coating, and direction of travel.

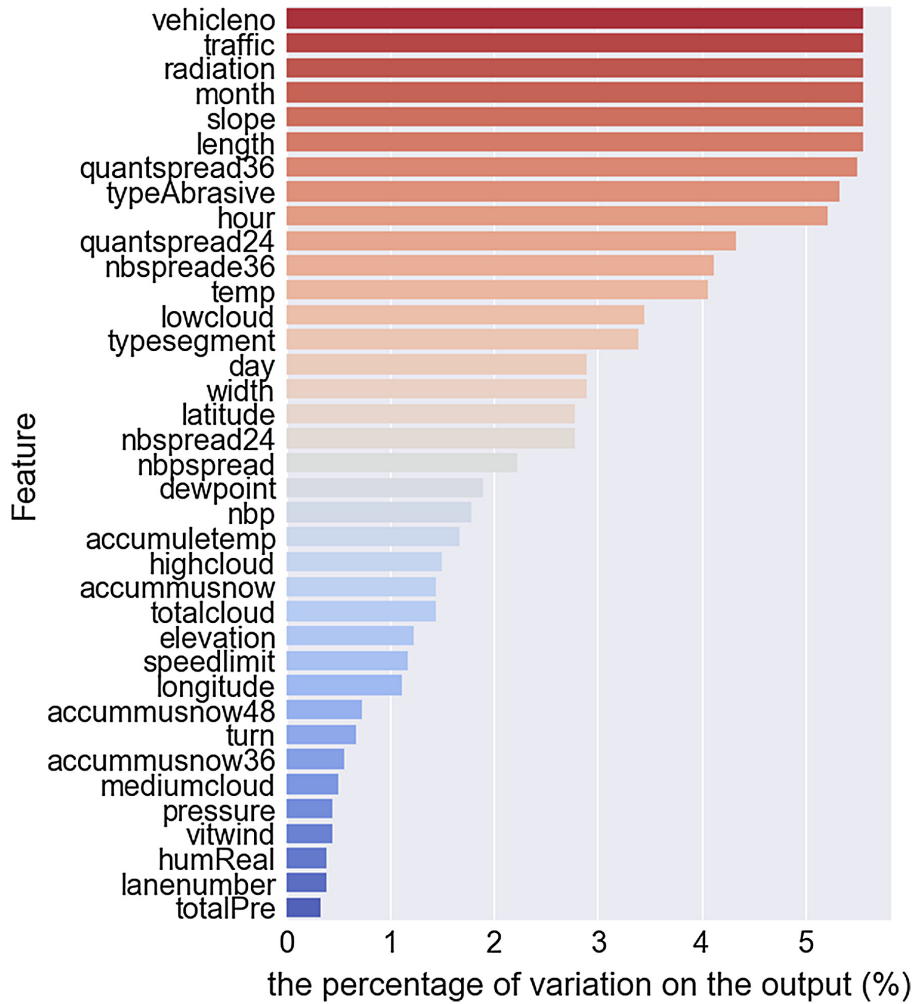


Figure 4.8 Feature ranking with the randomized Lasso method

4.5.3 Cross-validation

Tenfold cross-validation is used with hyperparameter tuning only on the training set (80% of our data) to compare different machine-learning algorithms. The hyperparameter tuning used the random search method developed by Bergstra and Bengio (2012). R2 and MSE are used as the metric functions (Figures 4.9 and 4.10). Linear regression was tested to show its poor performance.

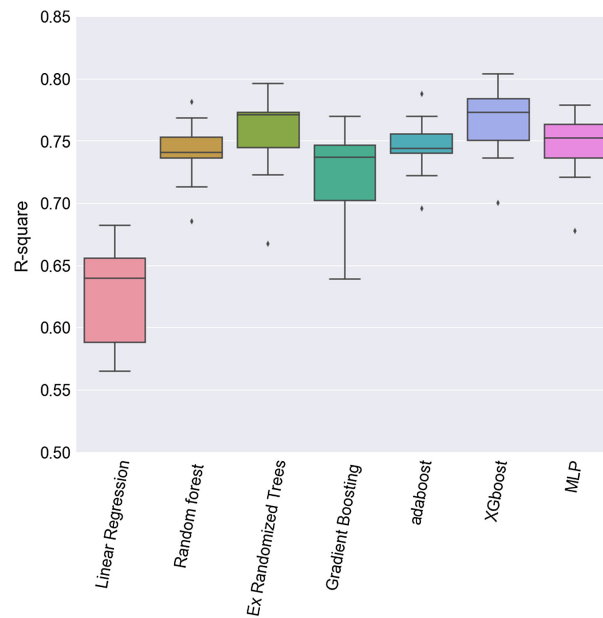


Figure 4.9 R^2 of machine-learning algorithms

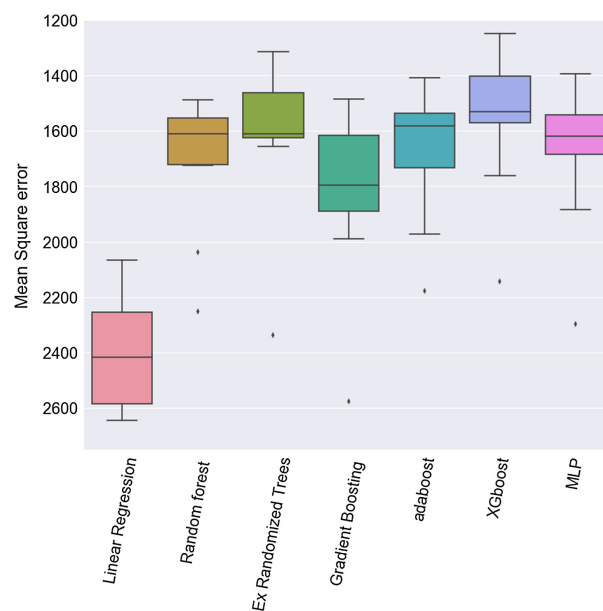


Figure 4.10 MSE of machine-learning algorithms

4.5.4 XGBoost algorithm evaluation and variable importance

The results show that XGBoost outperforms all the other techniques, even MLP. Therefore, this study continues with XGBoost. The training set (Figure 4.11) is used to train the model

with this method, while the testing set (Figure 4.12) is used to evaluate the model (R^2 train = 0.92, R^2 test = 0.83; MSE train = 876.54, MSE test = 1,256.93).

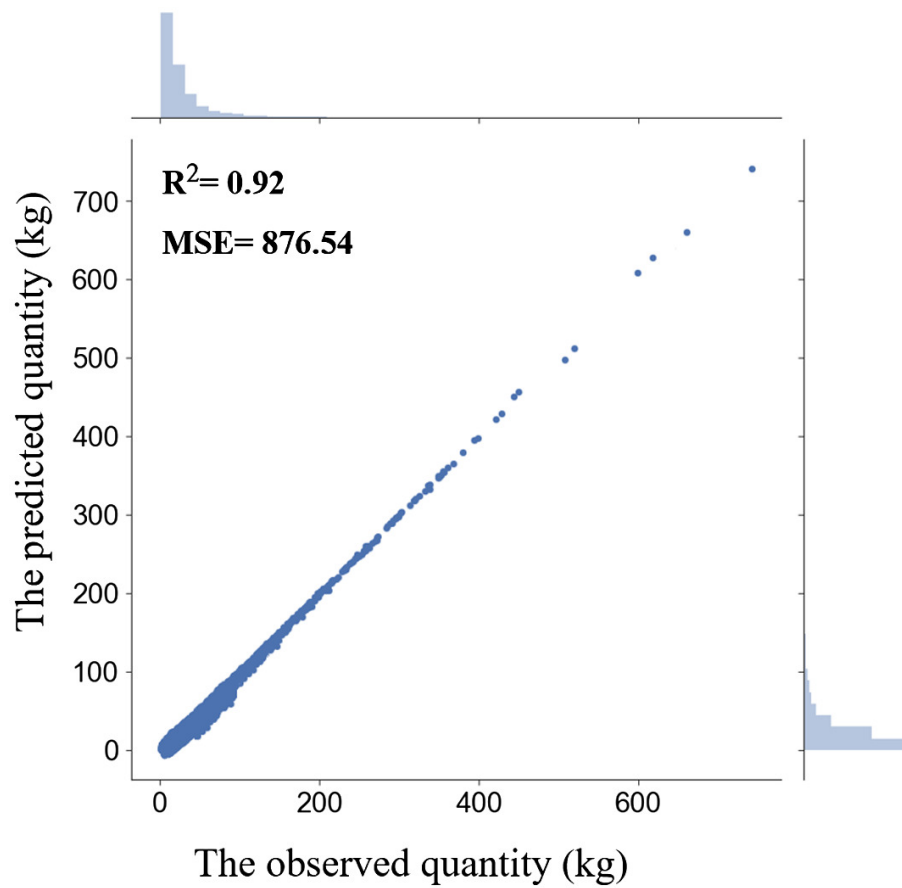


Figure 4.11 Training results with XGBoost

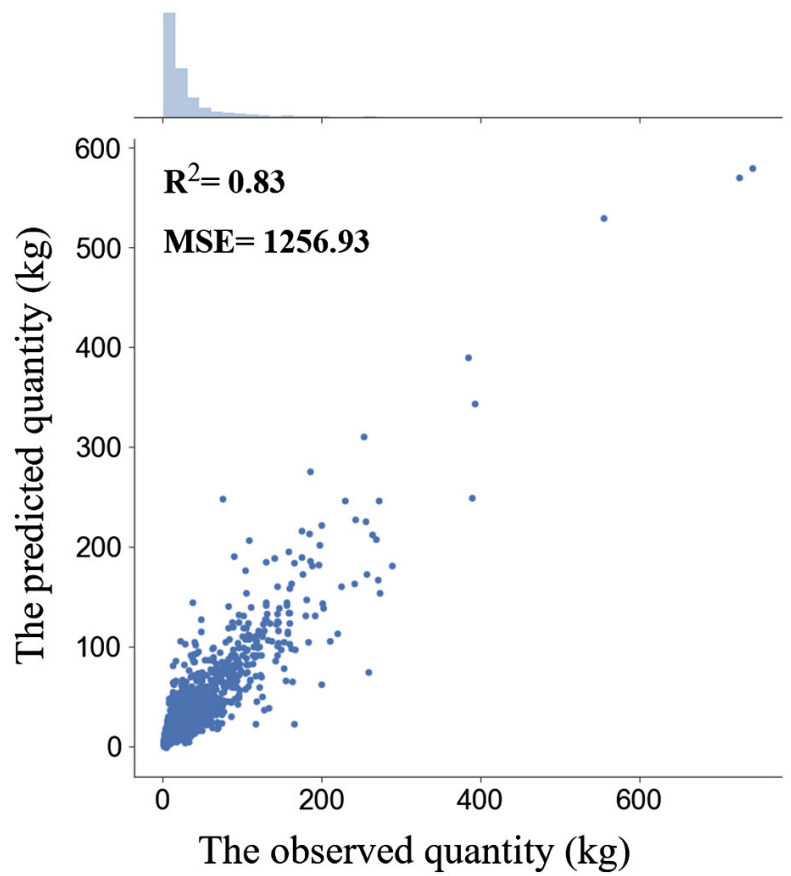


Figure 4.12 Test results with XGBoost

Figure 4.13 presents the most important variables; the variables with the greatest influence are: estimated number of passages, segment length, traffic, previous salting quantity, total number of passages, truck identity, width, pressure, temperature, and elevation.

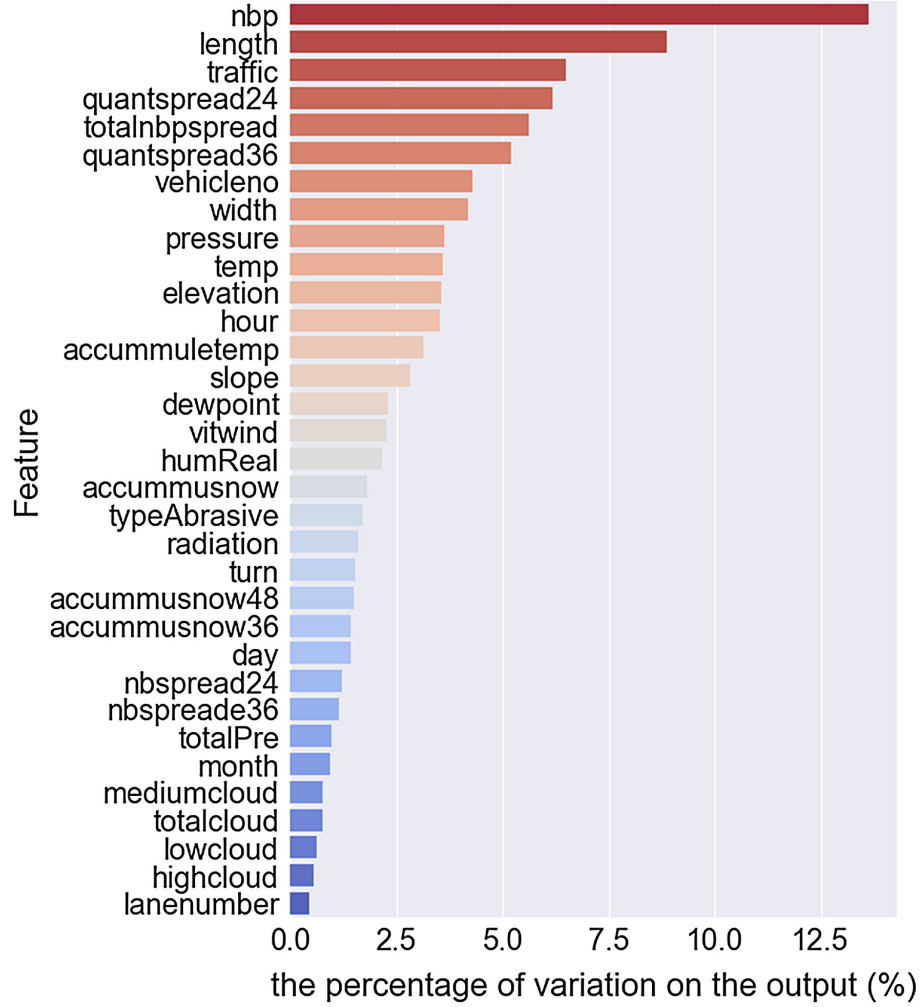


Figure 4.13 Importance of variables

4.5.5 Error analysis

The absolute error ratio is calculated with respect to the segment length via:

$$errorratio_i = \frac{\sum_{k=0}^{n_i} |\hat{y}_{ik} - y_{ik}|}{n_i D_i} \quad (4.7)$$

where:

y_{ik} = the observed value k of segment i ;

$\hat{y}_{ik}$ = predicted value k of segment i ;

n_i = the number of observed values of segment i ;

4.5.6 Policy implications and operational improvements

These results were found interesting by the snow removal managers. Firstly, the variable importance (Figure 4.13) shows that the human factor is among the most influencing factors that could be controlled to improve the salting operations: truck identity (trucks are usually assigned to the same drivers), number of passages (assigned by the managers), previous salting quantity (previous operations). Secondly, the distribution of errors on the map (Figure 4.14) allows the manager to locate segments with high prediction errors and subsequently apply different policies, standardizing driver behavior or even rebuilding vehicle routes for salting operations. Finally, the prediction model can be used to optimize vehicle routes because one can predict the amount needed for each segment and for each hour. This can help to reduce the cost of fuel by limiting truck weights.

4.6 Conclusions

Our goal was to predict the quantity of salt and abrasives to use for each street segment using GIS. We first cleaned the data and then used feature extraction to determine the potentially important variables. We applied feature selection to remove the variables with no significant influence. We then evaluated several machine-learning algorithms.

XGBoost outperformed all the other techniques ($R^2 = 0.83$). It provides a good prediction model, even without plowing operation data. The most important variables are: number of passes, segment length, traffic, previous salting quantity, number of passages, truck identity, width, pressure, temperature, and elevation. The proposed method could be applied to other regions if similar data, with high frequency, is available (telemetry, geomatics, weather). With available data, this prediction model can also be used for different applications of salt (anti-icing or pre-wetting), and for various winter road maintenance practices such as the increased use of anti-icing, direct liquid applications, or slurry applications

One limitation of our approach is that the prediction assumes that the operators are able to ensure the right level of service without salting too much (or not enough). Oversalting has a negative environmental impact. It is difficult to find a balance between clean pavements and

preserving the environment. However, we believe that the operators' experience is a good proxy for the ideal salting quantities, and that data provided over long periods of time will counterbalance the effect.

Another limitation is that the model does not predict the salt and abrasive quantities separately. Our data set has 29 million points, but it covers just one season and does not have sufficient numbers of precipitation episodes for the separate treatment of salt and abrasives. However, we believe that, with sufficient data and a longer time period, it could be used successfully for separate prediction.

Plans for future work include:

- The estimation of the total salt and abrasive quantity for each truck to optimize fuel costs by limiting truck weight.
- The development of vehicle routes for salting operations, because under different weather conditions, the salting quantities will change for each street segment, which implies a change in the routes.

4.7 ACKNOWLEDGMENTS

The authors are grateful for the important contributions of Christian Bérubé from the city of Granby, Serge Hamel from nXstream, Marco Dubé from Form-Éval, and François Poulin from Isaac Instruments. This project was funded by the Nature and Technology Research Fund of Quebec.

CHAPITRE 5 ARTICLE 2: THE MIXED CAPACITATED GENERAL ROUTING PROBLEM WITH TIME-DEPENDENT DEMANDS

C. Ahabchane, A. Langevin et M. Trépanier

Cet article a été soumis pour publication dans *Networks*.

5.1 Abstract

The Mixed Capacitated General Routing Problem (MCGRP) is defined over a mixed graph, for which some nodes, arcs, and edges must be serviced. The problem consists of determining a set of routes of minimum cost that satisfy the demand. Some problems like salt spreading have a time-dependent demand which was ignored in the previous studies. This variation of demand is due to the weather or traffic conditions. This study presents a mixed integer programming model without graph transformation to nodes routing. We use CPLEX to solve small instances and we adapt a Slack Induction by String Removals metaheuristic for large instances to this problem. The proposed model and metaheuristic were tested on problems derived from a set of classical instances of the MCGRP with some modifications to include time-dependent demands.

5.2 Introduction

This paper introduces a new problem motivated by road winter maintenance when the demand of salt or abrasive on street segments depends on time. The mixed capacitated general routing problem with time-dependent demands (MCGRP-TDD) is the problem of finding a set of routes with minimum total cost, starting and ending at the depot and satisfying the time-dependent demand of all required edges, arcs, and nodes without violating the vehicle capacity. The real-life application underlying this study is the salt and abrasives spreading in winter road maintenance because the quantity required on each component of street network (intersection, street segment) varies according to the weather conditions and the traffic that depend on the time. [Ahabchane *et al.* \(2019\)](#) used machine learning techniques to predict the

quantity of salt and abrasive needed for a specific road segment for each hour, based on truck telemetry data, weather conditions, and segment attributes. They also pointed out that this prediction model could be used to optimize the routing problem, since the quantity depends on time. Having a good prediction model to the demand on each period can thus be used to generate a time dependent demand on each component of the street network.

Recently time-dependency has received more attention due to the growing interest of researchers for the use of real data in academic problems and the recent advances in data-processing techniques. Hence its application arises in different fields: route planning on road networks, travel planning in public transit networks, and also in some robotic and military applications ([Gendreau *et al.*, 2015](#)). In particular, time-dependent vehicle routing problems have been widely studied in the literature with different variants of time dependency: service time (e.g., [Malandraki et Daskin \(1992\)](#), [Ichoua *et al.* \(2003\)](#), [Donati *et al.* \(2008\)](#)), time windows (e.g., [Dabia *et al.* \(2013\)](#), [Jung et Haghani \(2001\)](#), [Balseiro *et al.* \(2011\)](#)), demand (e.g., [Victoria *et al.* \(2016\)](#)). For more details we refer the reader to [Mourão et Pinto \(2017\)](#) and [Gendreau *et al.* \(2015\)](#).

Unlike the time-dependent vehicle routing problem, the arc routing problem (TD-ARP) has been addressed in very few studies and in the form of several variants of time dependency (service time, cost, zigzag option). Service time was the most frequent variant of TD-ARP addressed in the literature. [Sun *et al.* \(2011\)](#), [Tan et Sun \(2011\)](#) and [Sun *et al.* \(2015\)](#) presented different solution approaches for the Chinese postman problem with time dependent travel time. [Black *et al.* \(2013\)](#) introduced a new problem named Time-Dependent Prize-Collecting Arc Routing Problem (TD-PARP). This problem handles the situation where a transport manager has to choose between a number of full truck load pick-ups and deliveries on a road network with time-dependent travel time. [Kayacı Çodur et Yılmaz \(2018\)](#) used genetic algorithm and hybrid simulated annealing for the Hierarchical Chinese Postman Problem with Time Dependent Travel Time. [Tagmouti *et al.* \(2007\)](#) and [Tagmouti *et al.* \(2010\)](#) presented a time-dependent cost variant for a capacitated arc routing problem. The problem was motivated by the winter spreading application where the timing of the intervention is crucial.

They considered a service cost on each arc as a piece-wise linear function. If the service of an arc is too early or too late, the cost in time and de-icing material increases. This function also includes an optimal time interval where the service cost is minimal. [Tagmouti et al. \(2011\)](#) addressed a dynamic variant of the same problem. The dynamic aspect changes these optimal service time intervals due to weather report updates.

[Nossack et al. \(2017\)](#) combined time dependencies with arc routing problem with a zigzag option, which means serving both sides of the street in a single pass. The zigzagging is allowed when the street is narrow or traffic is light during some periods of day. In other cases, the street must be served by two passes. Real life applications of this problem arise in trash collection and newspaper delivery. The common approach for modeling arc routing problem with time variants is to modify the graph or transform an arc routing problem to node routing problem (e.g., [Aminu et Eglese \(2006\)](#), [Mullaseril \(1997\)](#), [Wøhlk \(2005\)](#), and [Vansteenwegen et al. \(2010\)](#)). Because without a graph modification or node transformation it is hard to associate a unique starting and ending time to an arc or edge ([Mullaseril, 1997](#)). Except in [Lystlund et Wøhlk \(2011\)](#), [Wang et Wen \(2002\)](#), [Sun et al. \(2015\)](#) and [Kayacı Çodur et Yilmaz \(2018\)](#) they used the original graph.

This study presents a mixed integer programming formulation without graph modification, and a modified Slack Induction by String removals (SISRs) metaheuristic inspired by [Christiaens et Vanden Berghe \(2020\)](#), in order to solve medium- and large-scale problems derived from a set of classical instances of the MCGRP with some modifications adapted for this problem. To the best of our knowledge, time-dependent demand has not been addressed in MCGRP or arc routing problems.

The paper is organized as follows. In the next section, a mathematical model is formulated. In the following section [5.4](#), we present a description of the SISRs metaheuristic developed for the problem with a detailed explanation of the cost and capacity evaluation. The computational experiments section [5.5](#) presents a description of the new instances generated to evaluate the performance of the modified SISRs, followed by a presentation and discussion

of the results obtained. The final section 5.6 concludes with limitations and perspectives.

5.3 Mathematical formulation

The TDD-MCGRP can be defined on a mixed graph $G(V, A, E)$, where V is the set of vertices or nodes, A is the set of arcs, E is the set of edges. It should be mentioned that in the real life application the nodes correspond to intersections and the arcs to the street segments, whereas the edges correspond to narrow street segments when they can be served on both sides only in one passage. Vertex 0 represents the depot and holds a homogeneous fleet of vehicles with capacity Q , each link $(i, j) \in A \cup E$ has a cost c_{ij} , additionally each required link $(i, j) \in A_R \cup E_R$ (A_R and E_R represent the required arcs and edges respectively) has a time dependent demand given by a function $\mathcal{F}(\cdot)$ according to the serving time TD_{ij}^k of link (i, j) by vehicle k , and each vertex $i \in V_R$ (V_R represents the required vertices) has also a time dependent demand given by the same function $\mathcal{F}(\cdot)$ according to the serving time TD_i^k of node i by vehicle k . $\mathcal{F}(\cdot)$ is a piecewise linear function defined on $|H|$ intervals or periods, each period h has a lower and upper bound MIN_{ij}^h, MAX_{ij}^h (MIN_i^h, MAX_i^h for vertex i) respectively, and the value of this function at period h is q_{ij}^h on link (i, j) and q_i^h on vertex i . We use the concept of steps introduced by [Lystlund et Wøhlk \(2011\)](#) in order to make the utilization of time with arc routing problem feasible. For the adjacent arcs (i, j) and (j, k) on a route, when the arc (i, j) is traversed at step s from i to j , then arc (j, k) is traversed at step $s + 1$.

We propose an integer linear programming formulation inspired by [Lystlund et Wøhlk \(2011\)](#). This formulation contains non-linear constraints (5.6) which will be linearized after in Section 5.3.1.

Variables :

x_{ij}^k	binary variable equal to 1 if and only if (i, j) is serviced by vehicle k from vertex i to vertex j
y_{ij}^k	integer variable representing the number of deadheading from vertex i to vertex j by vehicle k
β_{ij}^{sk}	binary variable equal to 1 if and only if (i, j) is visited by vehicle k at step s
w_{ij}^{sk}	binary variable equal to 1 if and only if (i, j) is serviced by vehicle k at step s
z_i^{sk}	binary variable equal to 1 if and only if node i is serviced by vehicle k at step s
T_i^{ks}	non-negative variable representing the visiting time at node i by vehicle k at step s
T_{ij}^{ks}	non-negative variable representing the visiting time at link (i, j) by vehicle k at step s
TD_i^k	non-negative variable representing the serving time at node i by vehicle k
TD_{ij}^k	non-negative variable representing the serving time at link (i, j) by vehicle k
φ_{ij}^{hk}	binary variable equal to 1 if (i, j) is served by vehicle k at period h
φ_i^{hk}	binary variable equal to 1 if i is served by vehicle k at period h
$\tilde{q}_i^k$	non-negative variable representing the quantity delivered at node i by vehicle k
$\tilde{q}_{ij}^k$	non-negative variable representing the quantity delivered at link (i, j) by vehicle k

$$\text{Min } Z = \sum_{k \in K} \sum_{(i,j) \in E_R \cup A_R} c_{ij}(x_{ij}^k + x_{ji}^k) + \sum_{k \in K} \sum_{(i,j) \in E \cup A} c_{ij}(y_{ij}^k + y_{ji}^k) \quad (5.1)$$

Subject to :

$$\sum_{k \in K} (x_{ij}^k + x_{ji}^k) = 1, \quad \forall (i, j) \in E_R \quad (5.2)$$

$$\sum_{k \in K} x_{ij}^k = 1, \quad \forall (i, j) \in A_R \quad (5.3)$$

$$\sum_{s \in S} \sum_{k \in K} z_i^{ks} = 1, \quad \forall i \in V_R \quad (5.4)$$

$$\sum_{(i,j) \in A_R \cup E_R} x_{ij}^k + \sum_{(i,j) \in A \cup E} y_{ij}^k = \sum_{(j,i) \in A_R \cup E_R} x_{ji}^k + \sum_{(j,i) \in A \cup E} y_{ji}^k, \quad \forall k \in K, i \in V \quad (5.5)$$

$$\sum_{(i,j) \in E_R} \mathcal{F}(TD_{ij}^k) + \sum_{(i,j) \in A_R} \mathcal{F}(TD_{ij}^k) + \sum_{i \in V_R} \mathcal{F}(TD_i^k) \leq Q, \quad \forall k \in K \quad (5.6)$$

$$\sum_{(i,j) \in E} \beta_{ij}^{ks} \leq 1, \quad \forall k \in K, s \in S \quad (5.7)$$

$$\sum_{j \in V} \beta_{ji}^{ks} - \sum_{j \in V} \beta_{ij}^{k,s+1} = 0, \quad \forall k \in K, s \in S, i \in V \quad (5.8)$$

$$\sum_{l \in V} \beta_{0l}^{k1} \geq \beta_{ij}^{ks}, \quad \forall k \in K, s \in S \setminus \{1\}, (i, j) \in E \cup A \quad (5.9)$$

$$\sum_{j \in V} \beta_{ij}^{ks} \geq z_i^{ks}, \quad \forall k \in K, s \in S, i \in V_R \quad (5.10)$$

$$\beta_{ij}^{ks} \geq w_{ij}^{ks}, \quad \forall k \in K, s \in S, (i, j) \in E_R \cup A_R \quad (5.11)$$

$$x_{ij}^k = \sum_{s \in S} w_{ij}^{ks}, \quad \forall k \in K, (i, j) \in E_R \cup A_R \quad (5.12)$$

$$y_{ij}^k + x_{ij}^k = \sum_{s \in S} \beta_{ij}^{ks}, \quad \forall k \in K, (i, j) \in E \cup A \quad (5.13)$$

$$T_i^{ks} \leq T_{ij}^{ks} + M(1 - \beta_{ij}^{ks}), \quad \forall k \in K, s \in S, (i, j) \in E \cup A \quad (5.14)$$

$$T_{ij}^{ks} \leq T_j^{k,s+1} - t_{ij} + M(1 - \beta_{ij}^{ks}), \quad \forall k \in K, s \in S, (i, j) \in E \cup A \quad (5.15)$$

$$T_{ij}^{ks} \geq T_j^{k,s+1} - t_{ij} - M(1 - \beta_{ij}^{ks}), \quad \forall k \in K, s \in S, (i, j) \in E \cup A \quad (5.16)$$

$$TD_{ij}^k \geq T_{ij}^{ks} - M(1 - w_{ij}^{ks}), \quad \forall k \in K, s \in S, (i, j) \in E_R \cup A_R \quad (5.17)$$

$$TD_{ij}^k \leq T_{ij}^{k,s} + M(1 - w_{ij}^{ks}), \quad \forall k \in K, s \in S, (i, j) \in E_R \cup A_R \quad (5.18)$$

$$TD_i^k \geq T_i^{ks} - M(1 - z_i^{ks}), \quad \forall k \in K, s \in S, i \in V_R \quad (5.19)$$

$$TD_i^k \leq T_i^{ks} + M(1 - z_i^{ks}), \quad \forall k \in K, s \in S, i \in V_R \quad (5.20)$$

$$T_0^{k,1} = 0, \quad \forall k \in K \quad (5.21)$$

$$T_{ij}^{ks} \leq T_i^{ks} + M(1 - \beta_{ij}^{ks}), \quad \forall k \in K, s \in S, (i, j) \in E \cup A \quad (5.22)$$

$$y_{ij}^k, \beta_{ij}^{ks} \in \{0, 1\}, \quad \forall k \in K, s \in S, (i, j) \in E \quad (5.23)$$

$$x_{ij}^k \in \{0, 1\}, \quad \forall k \in K, (i, j) \in E_R \cup A_R \quad (5.24)$$

$$z_i^{ks} \in \{0, 1\}, \quad \forall k \in K, s \in S, i \in V_R \quad (5.25)$$

$$T_i^{ks} \in \mathcal{R}^+, \quad \forall k \in K, i \in V, s \in S. \quad (5.26)$$

$$T_{ij}^{ks} \in \mathcal{R}^+, \quad \forall k \in K, s \in S, (i, j) \in E \cup A, \quad (5.27)$$

$$TD_{ij}^k \in \mathcal{R}^+, \quad \forall k \in K, (i, j) \in E_R \cup A_R \quad (5.28)$$

$$TD_i^k \in \mathcal{R}^+, \quad \forall k \in K, i \in V_R. \quad (5.29)$$

The objective function (5.1) minimizes the total cost. Constraints (5.2) - (5.4) ensure that all required elements (node, arc, edge) are served. Constraints (5.5) impose the flow conservation, (5.6) model the time dependent demand limitations imposed by the capacity Q of each vehicle, where $\mathcal{F}(\cdot)$ is a piece-wise linear function. (5.7) implies that each vehicle has at most one task at each step. (5.8) presents the connectivity constraint, (5.9) imply that the first step starts from the depot. (5.10) links the passage by node i at step s with the service of node i at step s . Also (5.11) links the passage on link (i, j) at step s with the service of the

required link (i,j) at step s by the same vehicle. Constraints (5.12) ensure that a required link is served at only one step. Constraints (5.13) guarantee that the number of steps for each vehicle is equal to the number of traversed links. Constraints (5.14)-(5.16) define the visiting time on link (i,j) and node i . Constraints (5.17)-(5.20) define the servicing time on link (i,j) and node i . Constraints (5.21) determine the starting time and finally the constraints (5.22) prohibit the waiting time. Constraints (5.23)-(5.29) define the variables.

5.3.1 Linearization of the demand constraints

In this study a stepwise linear function is used, because in the real life application with the available data, we have a prediction model for each hour (Ahabchane *et al.*, 2019). The demand constraints (5.6) being non-linear, we introduce new binary variables φ_{ij}^{hk} , φ_i^{hk} in order to replace these constraints by the linear constraints (5.31)-(5.39). MIN_{ij}^h and MAX_{ij}^h represent respectively the minimum time value and the maximum time value of period h . Figure 5.1 illustrates this linearization with an example. If the serving time $TD_{ij} = 13$, it means that the link (i,j) is served at period $h = 3$ with $q_{ij}^3 = 55$ kg, $MIN_{ij}^3 = 9$ and $MAX_{ij}^3 = 21$.

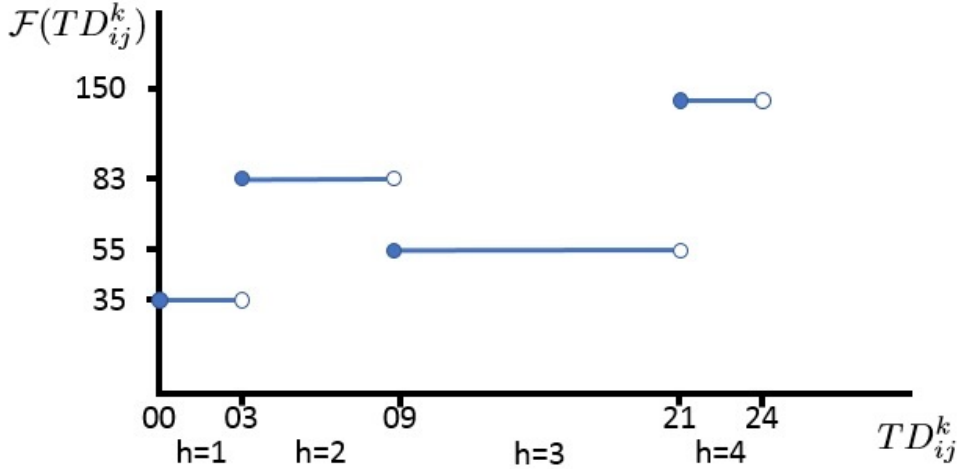


Figure 5.1 Example of a time dependent demand function for a specific link (i,j)

$$\sum_{(i,j) \in E_R} \tilde{q}_{ij}^k + \sum_{(i,j) \in A_R} \tilde{q}_{ij}^k + \sum_{i \in V_R} \tilde{q}_i^k \leq Q, \quad \forall k \in K \quad (5.31)$$

$$TD_{ij}^k \leq MAX^h + M(1 - \varphi_{ij}^{hk}) \quad \forall h \in H, (i, j) \in E_R \cup A_R \quad (5.32)$$

$$TD_{ij}^k \geq MIN^h - M(1 - \varphi_{ij}^{hk}) \quad \forall h \in H, (i, j) \in E_R \cup A_R \quad (5.33)$$

$$\sum_{h \in H} \varphi_{ij}^{hk} = \sum_{s \in S} w_{ij}^{ks}, \quad \forall k \in K, (i, j) \in E_R \cup A_R \quad (5.34)$$

$$\tilde{q}_{ij}^k = q_{ij}^h \sum_{h \in H} \varphi_{ij}^{hk} \quad \forall k \in K, (i, j) \in E_R \cup A_R \quad (5.35)$$

$$TD_i^k \leq MAX^h + M(1 - \varphi_i^{hk}) \quad \forall h \in H, i \in V_R \quad (5.36)$$

$$TD_i^k \geq MIN^h - M(1 - \varphi_i^{hk}) \quad \forall h \in H, i \in V_R \quad (5.37)$$

$$\sum_{h \in H} \varphi_i^{hk} = \sum_{s \in S} z_i^{ks}, \quad \forall k \in K, i \in V_R \quad (5.38)$$

$$\tilde{q}_i^k = q_i^h \sum_{h \in H} \varphi_i^{hk} \quad \forall k \in K, i \in V_R \quad (5.39)$$

Constraints (5.31) are a linear version of (5.6). Constraints (5.32)-(5.33) and (5.36)-(5.37) ensure that the serving time is in the right period. Constraints (5.34) and (5.38) imply that each required element is served at one unique period. Constraints (5.35) and (5.39) determine

the quantity spread on the period chosen.

5.4 Modified SISRs

The SISRs (Slack Induction by String Removals) metaheuristic has been developed by [Christiaens et Vanden Berghe \(2020\)](#). It is based on Ruin and Recreate (R^- and R^+) Strategies, but unlike other R^- and R^+ methods (for example, [Ropke et Pisinger 2006](#)) SISRs uses a single R^- (adjacent string removal) and a single R^+ (greedy insertion with blinks). This new metaheuristic obtained competitive results and new best results on some VRP various problems, especially for large-scale instances. We have modified the SISRs metaheuristic to adapt it to the TDD-MCGRP. This section highlights the solution representation and points out how the cost and the capacity of routes are evaluated. Finally the SISRs metaheuristic that we developed is explained.

5.4.1 Solution representation

The particularity of an arc routing problem compared with the node routing problem is the presence of two possible directions to serve an edge. That is why it is important to define a suitable solution representation in order to reduce computing time during moves evaluation. This metaheuristic uses the [Vidal \(2017\)](#) solution representation and the optimal direction choice can be found optimally using dynamic programming as in [Beullens *et al.* \(2003\)](#). From the original graph, only the required elements are considered and each required edge is transformed into two nodes, one for each direction. The shortest path between these required elements is found using the Floyd Warshall algorithm. Figure 5.2 illustrates the solution representation with an example of transformation from the original graph (Figure 5.2a), where the green links presents the required links, e_i for edges and a_i for arcs, the distance is unitary between each node. Figure 5.2b presents the graph corresponding to a possible sequence for a solution.

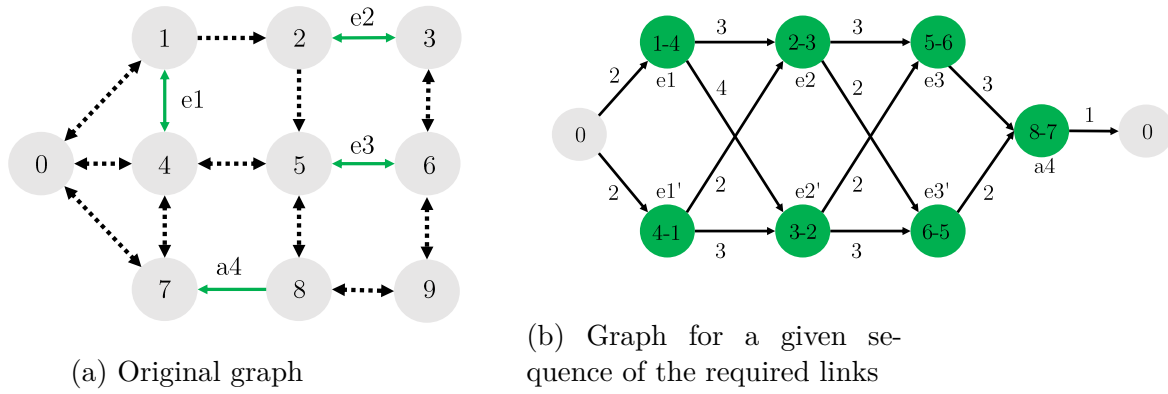


Figure 5.2 Solution representation example

5.4.2 Cost and Capacity Evaluation

The solution cost for given route presented as a set of required elements is calculated using dynamic programming which determine the optimal direction choice for each required edge. Therefore, for a given sequence σ^k , two costs are defined for each required element i (i' presents the other direction if i is an edge):

1. $Cost(\sigma_i^k)$ the front cost from the starting depot to the element i using Equation 5.40, for each element i the $Cost(\sigma_i^k)$ is equal to the minimum between the previous element cost $Cost(\sigma_{i-1}^k)$ plus the shortest path from $i-1$ to i and the previous element cost of the other direction $Cost(\sigma_{i'-1}^k)$ plus the shortest path from $i'-1$ to i .
2. $Cost^-(\sigma_i^k)$ the back cost from the required element i to the ending depot using Equation 5.41, for each element i the $Cost(\sigma_i^k)$ is equal to the minimum between the next element cost $Cost(\sigma_{i+1}^k)$ plus the shortest path from i to $i+1$ and the next element cost of the other direction $Cost(\sigma_{i'+1}^k)$ plus the shortest path from i to $i+1$.

These front and back costs $Cost(\sigma_i^k)$, $Cost^-(\sigma_i^k)$ are not used only to evaluate the solution cost of any element i by Equation 5.42, but are used also for *the move and capacity evaluation*:

- *The move evaluation* is needed to evaluate moves during the metaheuristic and local search procedure. For example, if we want to evaluate the insertion of element l on

a route k between two elements i and j , the cost of the new solution can be found using Equation 5.43, whereas $Cost(\sigma_l^k)$, $Cost^-(\sigma_l^k)$ can be found with Equations 5.40 and 5.41 where $l-1$ is i and $l+1$ is j . The idea of this example of insertion move remains the same for the other moves.

- In order to evaluate the total demand of a set of required elements, we need to evaluate the demand every time when a move is applied on each sequence, starting from the element affected by the move until the last one. In fact the problem is a time dependent demand, which means that demand can take different values depending on the time arrival to the element, and when a move is performed before a given element it will affect the demand for all the following elements. Since the demand is a linear function $\mathcal{F}(\cdot)$ in terms of time (the time will be represented by the cost), the demand is calculated for each element using Equation 5.44. The idea behind the Equation 5.44 is to determine the optimal direction for a given element and its front cost : $Cost(\sigma_i^k)$, which referring to the arrival time. In some special case when an element has the same cost for both of the direction choices it is necessary to track the previous direction choice.

In order to simplify the calculation we duplicate as well the nodes and arcs on the solution representations (in Figure 5.3 arc (8, 7) is duplicated). Figures 5.3 and 5.4 illustrate the cost and demand calculations for the graph of figure 5.2b. The triplets on Figure 5.3 represents $(Cost(\sigma_i^k), Cost^-(\sigma_i^k), Demand(\sigma_i^k))$. Figure 5.4 presents the time dependent demand function on each required element for 10 periods.

$$Cost(\sigma_i^k) = Min\{Cost(\sigma_{i-1}^k) + cost(i-1, i), Cost(\sigma_{i-1'}^k) + cost(i-1', i')\} \quad (5.40)$$

$$Cost^-(\sigma_i^k) = Min\{Cost^-(\sigma_{i+1}^k) + cost(i, i+1), Cost^-(\sigma_{i+1'}^k) + cost(i', i+1')\} \quad (5.41)$$

$$Cost(\sigma_0^k, \sigma_1^k, \dots, \sigma_n^k, \sigma_0^k) = Min\{Cost^-(\sigma_i^k) + Cost(\sigma_i^k), Cost^-(\sigma_{i'}^k) + Cost(\sigma_{i'}^k)\} \quad \forall i \quad (5.42)$$

the minimum demand of all required elements divided by the capacity of vehicle (Equation 5.45), and all unserved clients are added to the set A (set of unserved clients). Unlike in the original SISRs where the initial solution contains the same number of vehicles as the number of clients. We don't use an algorithm to reduce the number of vehicles as in the original SISRs. Our modified SISRs starts with a minimum number of vehicles and after 5% of iterations if the set U is still not empty a new vehicle is added. Since the SISRs metaheuristic is based on Simulated Annealing, the algorithm starts with an initial temperature $\mathcal{T}_0$ (line 2); after each iteration this temperature is updated (line 42). The modified SISRs contains Ruin (Section 5.4.3) and Recreate (Section 5.4.3) phases as well as the classic SISRs, but it also includes a Local Search phase (Section 5.4.3).

$$N_{vehicles} = \frac{\sum_{i=1}^{|V_R \cup E_R \cup A_R|} \min_{\forall h \in H} d_i^h}{Q} \quad (5.45)$$

where d_i^h presents the demand of element i at period h .

Ruin Phase

Slack induction by string removal means that when a string (set of adjacent required elements on the same route) is removed, it introduces space or capacity slack (for the time-dependent demand problem it is not always true, for some cases removing an element may increase the total demand of the vehicle). A similar idea was used before with the SMART operator by [Rousseau et al. \(2002\)](#). The ruin phase is based on 3 premises : 1) Remove a sufficient number of elements, 2) Remove adjacent elements, 3) Remove strings of elements from different routes. The number of strings to remove k_s and the number of elements in this string l_t are defined by Equations 5.46 and 5.47, where l_s^{max} is the minimum between the average number of routes $\overline{|t \in T|}$ and L^{max} the maximum number of elements to remove in a string (Equation 5.48), and $\bar{c}$ presents the average number of elements to remove. L^{max} and $\bar{c}$ were fixed at 10 as in the classical SISRs. The ruin phase algorithm starts with updating l_s^{max} , k_s , l_t ; a random element is chosen from set T as a seed c_t^{seed} ; from this seed we choose the k_s nearest elements c_t (the client itself as the first one) from different routes t . For each selected route t a l_t

nearest elements to c_t are selected to be removed and added to A with a probability of 99%.

$$k_s = \mathcal{U}(1, \frac{4\bar{c}}{1 + l_s^{max}}) \quad (5.46)$$

$$l_t = \mathcal{U}(1, \text{Min}\{|t|, l_s^{max}\} + 1) \quad (5.47)$$

$$l_s^{max} = \text{Min}\{L^{max}, \overline{|t \in T|}\} \quad (5.48)$$

Recreate Phase

The recreate phase starts with sorting the removed elements randomly, or by decreasing demand (the largest demand first) or by distance from the depot (closest from the depot and also the farthest) with different probabilities: 4/10 for the demand and for the random sorting, 1/10 for the distance sorting. Each element from A is evaluated on each position on each route with a probability of $1 - \beta$ ($\beta = 0.01$). The position chosen is the position with the lowest insertion cost that respects the capacity, otherwise a new route is created for this element.

Local search

A local search inspired by the education phase of the genetic algorithm proposed by [Vidal \(2017\)](#) is performed on the best solution found after the recreate phase. The local search operators used are 2-opt, 2-opt*, relocate and swap moves with possible reversal ([Laporte et al., 2014](#)). First improvement and granular local search ([Toth et Vigo, 2003](#)) policies are used as well.

Figure 5.5 illustrates an example of how the SISRs metaheuristic works. The example starts with a first solution where $k_s = 3$, the red element represents the seed c_1^{seed} and the blue ones

represent c_2 and c_3 . The cardinality of the removed string is determined for each element $l_1 = 4$, $l_2 = 5$ and $l_3 = 3$. After the ruin phase the algorithm starts the insertion at the recreate phase. Finally, a local search procedure (relocate, 2 swap) is added.

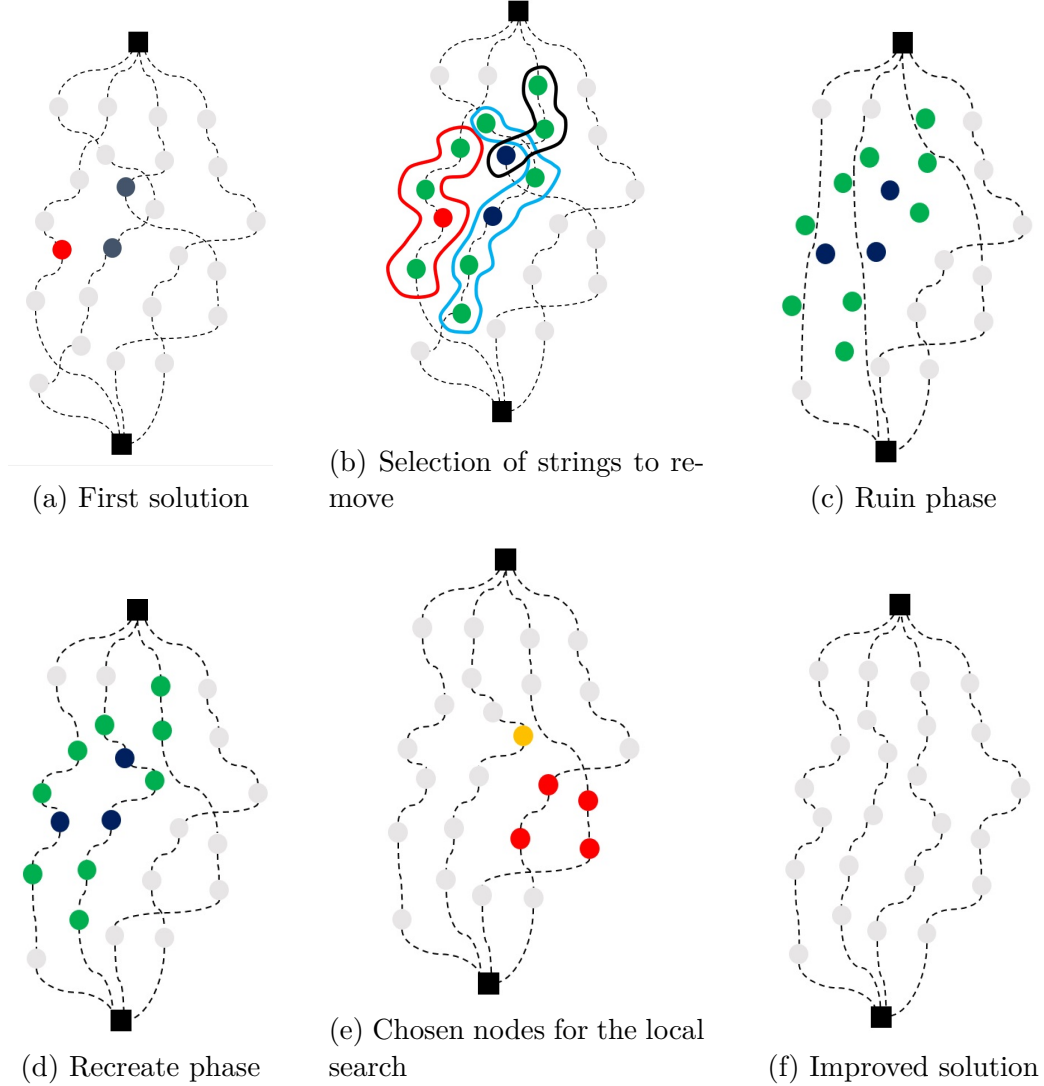


Figure 5.5 An example of the modified SISRs metaheuristic.

Algorithm 1: The Modified SISRs metaheuristic

Input: Initial solution $s = (T; U)$ where T is the set of routes and U the set of Unserved required elements

```

1  $s^{best} \leftarrow s$ 
2  $\mathcal{T} \leftarrow \mathcal{T}_0$ 
3 for  $Itr$  iterations do
4   Ruin Phase :
5     Update :  $l_s^{max}, k_s, l_t$ 
6     choose a random element  $c_t^* \notin U$  //  $adj(c_t^*)$ : adjacency list of element  $c_t^*$ ,
7        $R$ : list of removed route
8     for  $c_t \in adj(c_t^*)$  and  $|R| \leq k_s$  do
9       if  $c_t \notin U$  and  $t \notin R$  then
10         Update :  $l_s^{max}, l_t$   $R \leftarrow R \cup t$ 
11          $n \leftarrow 1$ 
12         /* Section 5.4.3 */
13         for  $c \in adj(c_t)$  and  $n \leq l_t$  do
14           if  $c \notin U$  and  $c \in Route(c_t)$  then
15             if  $U(0, 1) \leq 0.99$  then
16                $U \leftarrow U \cup c$ 
17              $n \leftarrow n + 1$ 
18   Recreate Phase :
19     sort( $U$ )
20     for  $c \in U$  do
21       Insertion( $r$  : route,  $p$  : position,  $insertionCost$  : cost)  $\leftarrow (\emptyset, \emptyset, M)$ 
22       for  $t \in T$  do
23         for  $m$  in  $|t|$  //  $m$  presents the position in route  $t$  do
24           if  $U(0, 1) \leq 1 - \beta$  then
25             if  $Demand(t \cup \{c\}) \leq Q$  and  $cost(t \cup \{c\}) \leq insertionCost$ 
26               then
27                 Insertion  $\leftarrow (t, m, cost(t \cup \{c\}))$ 
28                 /* Section 5.4.3 */
29             if Insertion =  $(\emptyset, \emptyset, M)$  then
30               add new route  $t'$ ,  $t' \leftarrow t \cup \{c\}$ 
31             else
32               insert  $c$  in route  $r$ , at position  $p$ 
33              $U \leftarrow U \setminus \{c\}$ 
34   Local search :
35     for a random local search operator  $\phi$  in  $\{Opt, Opt^*, Relocate, Swap\}$  do
36       for  $t_i \in T$  do
37         for  $t_j \in T$  do
38           if  $cost(\phi(t_i)) + cost(\phi(t_j)) \leq cost(t_i) + cost(t_j)$  then
39             if  $Demand(\phi(t_i)) \leq Q$  and  $Demand(\phi(t_j)) \leq Q$  then
40                $t_i \leftarrow \phi(t_i), t_j \leftarrow \phi(t_j)$ 
41               /* Section 5.4.3 */
42             Break
43    $s^* \leftarrow (T, U)$  //  $s^*$  presents the best solution found by the 3 phases
44   if  $Cost(s^*) \leq Cost(s) - \mathcal{T} \cdot \ln(U(0, 1))$  then
45      $s \leftarrow s^*$ 
46   if  $Cost(s^*) \leq Cost(s^{best})$  then
47      $s^{best} \leftarrow s^*$ 
48    $\mathcal{T} \leftarrow \mathcal{T} \cdot (\frac{\mathcal{T}^f}{\mathcal{T}_0})^{Itr}$  //  $\mathcal{T}^f = 1, \mathcal{T}_0 = 100$ : the final and the initial temperature

```

Output: Best solution

5.5 Computational Experiments

This section presents the instances used and the modification made to adapt to the new problem and then the results of the experiments to evaluate the mathematical model and the modified SISRs metaheuristic.

5.5.1 Benchmark instances

In order to test the performance of the formulation and the modified SISRs metaheuristic, we use some medium- and large-scale MCGRP instances: BHW, CBMix and DI-NEARP. The benchmark instances can be found at <https://github.com/vidalt/HGS-CARP/tree/master/Instances/MCGRP>; the instance characteristics are shown in Table 5.1. Our metaheuristic works well on these classic instances (i.e., without time dependent demand) with an average gap of 3% compared with the best results found so far (Vidal, 2017), we kept the same parameters used in the original SISRs. These instances have been then modified in order to be used for the time dependent demand problem.

Table 5.1 – Instance characteristics

Instance name	$ V $	$ E $	$ A $	$ V \cup E \cup A $	Reference
CBMix (23 instances)	[0, 93]	[0, 94]	[0, 149]	[20, 212]	Prins et Bouchenoua (2005a)
BHW(20 instances)	[4, 50]	[0, 51]	[7, 380]	[20, 410]	Bach et al. (2013)
DI-NEARP (24 instances)	[120, 347]	[120, 486]	0	[240, 833]	Bach et al. (2013)

5.5.2 Instances modification

Because the TDD-MCGRP is a new problem, it is necessary to create a new benchmark. The new benchmark is derived from the set of the classical instances presented previously. We propose different functions for generating the time-dependent demand on each required element i , where $\alpha \in [0, 1]$ is the perturbation of the fixed demand d_i from the classical instances, and h the trip's maximum duration, which is divided into n_h equal periods. The generated instances used $n_h = 10$ and $h = \frac{\text{best known solution}}{\text{number of vehicles}} \times 1.5$, the idea behind this equation is that the solutions found are imbalanced.

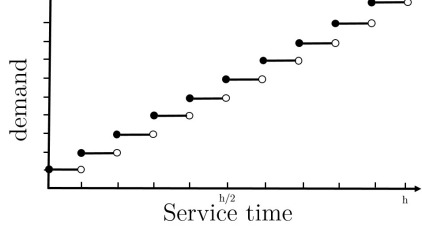
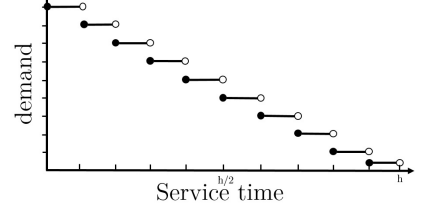
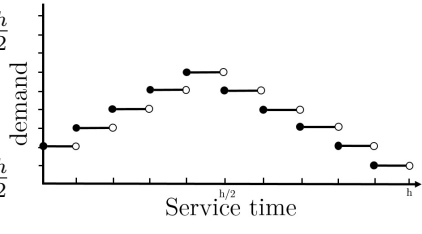
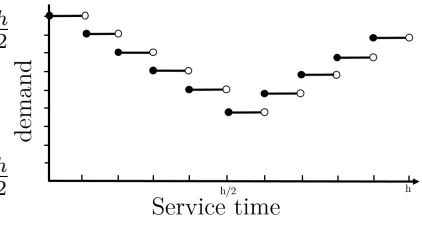
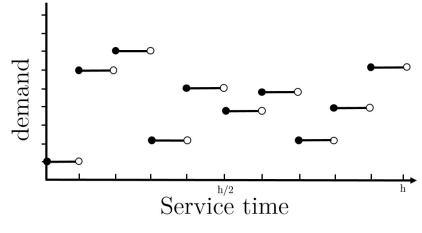
Function	Definition	Example
Linear positive	$\mathcal{F}(x) = \frac{2\alpha d_i}{n_h - 1} \left\lfloor \frac{x}{\frac{h}{n_h}} \right\rfloor + d_i(1 - \alpha), 0 \leq x \leq h$	
Linear negative	$\mathcal{F}(x) = \frac{-2\alpha d_i}{n_h - 1} \left\lfloor \frac{x}{\frac{h}{n_h}} \right\rfloor + d_i(1 + \alpha), 0 \leq x \leq h$	
Triangular Positive	$\mathcal{F}(x) = \begin{cases} \frac{2\alpha d_i}{\left\lfloor \frac{n_h - 1}{2} \right\rfloor} \left\lfloor \frac{x}{\frac{h}{n_h}} \right\rfloor + d_i(1 - \alpha) & \text{if } x \leq \frac{h}{2} \\ \frac{-2\alpha d_i}{\left\lfloor \frac{n_h - 1}{2} \right\rfloor} \left\lfloor \frac{x - \frac{n_h}{2}}{\frac{h}{n_h}} \right\rfloor + d_i(1 + \alpha) & \text{if } x > \frac{h}{2} \end{cases}$	
Triangular negative	$\mathcal{F}(x) = \begin{cases} \frac{-2\alpha d_i}{\left\lfloor \frac{n_h - 1}{2} \right\rfloor} \left\lfloor \frac{x}{\frac{h}{n_h}} \right\rfloor + d_i(1 + \alpha) & \text{if } x \leq \frac{h}{2} \\ \frac{2\alpha d_i}{\left\lfloor \frac{n_h - 1}{2} \right\rfloor} \left\lfloor \frac{x - \frac{n_h}{2}}{\frac{h}{n_h}} \right\rfloor + d_i(1 - \alpha) & \text{if } x > \frac{h}{2} \end{cases}$	
Random	A random number is generated between $d_i(1 - \alpha)$ and $d_i(1 + \alpha)$ for each period n_h	

Table 5.2 – Time dependent demand functions

5.5.3 Experimental results

Experiments were carried out on a PC with a 3.6-GHz processor and with 32 Go of RAM. Our metaheuristic was implemented in C++ and compiled with GNU under Ubuntu 16.04 operating system. The mathematical formulation was solved with CPLEX 12.9, all the CPLEX parameters set to their default values. The maximum allowed time to find a solution was set to 3600 seconds. Several tests were conducted in order to evaluate the performance of our metaheuristic. For each benchmark (BHW,CBMix, DI-NEARP) we test all the time dependent functions with values of $\alpha \in \{0.3, 0.5, 0.7\}$. The CPLEX found the optimal solution only for 6 smallest instances within 1 hour. Table 5.3, 5.4 and 5.5 present the comparison of results obtained by CPLEX and the metaheuristic for different functions and different values of α . The calculation of GAP is based on the lower bound found after 1 hour. We observe in the three tables the good performance of the metaheuristic in terms of time. In the tables the optimum values found by CPLEX and the metaheuristic are in bold. We can notice that the variation of functions and the perturbation values α have a considerable impact on the objective value. To confirm these remarks, an in-depth study is established on all instances.

Since there are no results for the TDD-MCGRP to compare with, a new metric (Equation 5.49) is defined in order to conduct a sensitivity analysis. This metric does not allow to evaluate the performance of our metaheuristic, but only to standardize the objective value in order to study the impact of different functions and different values of α on the objective.

$$Gap_{vidal} = \frac{\frac{\text{Vidal objective value on the fixed demand}}{\text{Vidal objective value on the fixed demand}} - \frac{\text{Modified SISRs objective value with TD demand}}{\text{Vidal objective value on the fixed demand}}}{\text{Vidal objective value on the fixed demand}} \times 100 \quad (5.49)$$

Figure 5.6 shows the sensitivity analysis on the objective with different parameters, in the form of box plots in order to display the distribution of results. We can notice that the impact of alpha on the objective is less strong than the impact of the different functions. Figure 5.7 presents only the impact of the different functions. We can observe the increase of objective value with linear negative and triangular positive which is the opposite for the linear positive and triangular negative functions. This huge variation is due to road imbalance, otherwise known as inequity workload.

Table 5.3 – Comparison of CPLEX results and metaheuristic results ($\alpha = 0.3$)

Instance name					CPLEX		Modified SISRs			
		$ E_R $	$ A_R $	$ V_R $	Objective	Lower bound	Time(s)	Objective	Gap	Time(s)
linear positive	BHW1	11	11	7	–	327	3600	337	3.05	14
	BHW2	0	25	4	470	470	895	470	0.0	24
	BHW3	8	7	5	404	404	2823	404	0.0	28
	BHW4	0	44	6	–	240	3600	240	0.0	75
	CBMix1	0	37	11	–	1944	3600	2445	25.77	88
	CBMix23	2	15	3	631	631	87	631	0.0	15
linear negative	BHW1	11	11	7	–	327	3600	377	15.29	158
	BHW2	0	25	4	520	520	3480	520	0.0	87
	BHW3	8	7	5	440	415	3600	422	1.68	35
	BHW4	0	44	6	–	240	3600	240	0.0	105
	CBMix1	0	37	11	–	1944	3600	2775	42.74	143
	CBMix23	2	15	3	677	677	1256	677	0.0	23
Triangular positive	BHW1	11	11	7	–	327	3600	365	11.62	85
	BHW2	0	25	4	520	480	3550	520	0.0	45
	BHW3	8	7	5	440	410	3600	422	2.93	36
	BHW4	0	44	6	–	240	3600	240	0.0	142
	CBMix1	0	37	11	–	1944	3600	2917	50.05	118
	CBMix23	2	15	3	743	743	3052	743	0.0	36
Triangular negative	BHW1	11	11	7	–	327	3600	331	1.22	42
	BHW2	0	25	4	470	470	423	470	0.0	23
	BHW3	8	7	5	393	393	3489	393	0.0	40
	BHW4	0	44	6	–	240	3600	240	0.0	86
	CBMix1	0	37	11	–	1788	3600	2333	30.48	74
	CBMix23	2	15	3	631	631	45	631	0.0	14
Random	BHW1	11	11	7	–	327	3600	340	3.89	42
	BHW2	0	25	4	470	470	680	470	0.0	28
	BHW3	8	7	5	415	393	3600	404	2.8	46
	BHW4	0	44	6	–	240	3600	240	0.0	88
	CBMix1	0	37	11	–	1944	3600	2566	32	97
	CBMix23	2	15	3	667	667	240	667	0.0	21

Table 5.4 – Comparison of CPLEX results and metaheuristic results ($\alpha = 0.5$)

Instance name					CPLEX		Modified SISRs			
		$ E_R $	$ A_R $	$ V_R $	Objective	Lower bound	Time(s)	Objective	Gap	Time(s)
linear positive	BHW1	11	11	7	–	327	3600	337	3.06	20
	BHW2	0	25	4	470	470	347	470	0.0	20
	BHW3	8	7	5	404	404	3210	404	0.0	30
	BHW4	0	44	6	–	240	3600	240	0.0	80
	CBMix1	0	37	11	–	1944	3600	2401	23.51	88
	CBMix23	2	15	3	631	631	20	631	0.0	10
linear negative	BHW1	11	11	7	–	327	3600	357	9.17	123
	BHW2	0	25	4	511	511	3560	511	0.0	72
	BHW3	8	7	5	415	404	3600	404	0.0	35
	BHW4	0	44	6	–	240	3600	240	0.0	80
	CBMix1	0	37	11	–	1944	3600	2738	40.84	104
	CBMix23	2	15	3	677	677	560	677	0.0	15
Triangular positive	BHW1	11	11	7	–	327	3600	353	7.64	60
	BHW2	0	25	4	480	480	3450	480	0.0	30
	BHW3	8	7	5	420	410	3600	415	1.22	42
	BHW4	0	44	6	–	240	3600	240	0.0	150
	CBMix1	0	37	11	–	1944	3600	2896	48.97	118
	CBMix23	2	15	3	743	743	2420	743	0.0	26
Triangular negative	BHW1	11	11	7	–	327	3600	337	3.05	34
	BHW2	0	25	4	470	470	70	470	0.0	20
	BHW3	8	7	5	383	377	3600	383	1.59	35
	BHW4	0	44	6	–	240	3600	240	0.0	120
	CBMix1	0	37	11	–	1788	3600	2076	16.11	98
	CBMix23	2	15	3	631	631	9	631	0.0	10
Random	BHW1	11	11	7	–	327	3600	337	3.05	25
	BHW2	0	25	4	470	470	429	470	0.0	35
	BHW3	8	7	5	393	393	3389	393	0.0	37
	BHW4	0	44	6	–	240	3600	240	0.0	88
	CBMix1	0	37	11	–	1944	3600	2435	25.26	97
	CBMix23	2	15	3	667	667	71	667	0.0	12

Table 5.5 – Comparison of CPLEX results and metaheuristic results ($\alpha = 0.7$)

Instance name					CPLEX		Modified SISRs			
		$ E_R $	$ A_R $	$ V_R $	Objective	Lower bound	Time(s)	Objective	Gap	Time(s)
linear positive	BHW1	11	11	7	–	315	3600	323	2.5	15
	BHW2	0	25	4	470	470	243	470	0.0	22
	BHW3	8	7	5	404	404	2456	404	0.0	25
	BHW4	0	44	6	–	240	3600	240	0.0	72
	CBMix1	0	37	11	–	1944	3600	2331	19.91	78
	CBMix23	2	15	3	631	631	37	631	0.0	8
linear negative	BHW1	11	11	7	–	327	3600	351	7.33	78
	BHW2	0	25	4	508	508	3352	508	0.0	54
	BHW3	8	7	5	386	386	3540	386	0.0	47
	BHW4	0	44	6	–	240	3600	240	0.0	88
	CBMix1	0	37	11	–	1944	3600	2646	36.11	113
	CBMix23	2	15	3	677	677	423	677	0.0	11
Triangular positive	BHW1	11	11	7	–	327	3600	345	5.5	44
	BHW2	0	25	4	480	480	2873	480	0.0	26
	BHW3	8	7	5	404	377	3600	386	2.38	65
	BHW4	0	44	6	–	240	3600	240	0.0	123
	CBMix1	0	37	11	–	1944	3600	2770	42.49	98
	CBMix23	2	15	3	707	707	1895	707	0.0	18
Triangular negative	BHW1	11	11	7	–	315	3600	323	2.5	12
	BHW2	0	25	4	470	470	78	470	0.0	14
	BHW3	8	7	5	377	377	3352	377	0.0	18
	BHW4	0	44	6	–	240	3600	240	0.0	123
	CBMix1	0	37	11	–	1788	3600	2067	15.6	85
	CBMix23	2	15	3	631	631	17	631	0.0	13
Random	BHW1	11	11	7	–	327	3600	337	3.05	12
	BHW2	0	25	4	470	470	687	470	0.0	41
	BHW3	8	7	5	393	393	3145	393	0.0	26
	BHW4	0	44	6	–	240	3600	240	0.0	72
	CBMix1	0	37	11	2569	1944	3600	2413	24.12	109
	CBMix23	2	15	3	631	631	145	631	0.0	8

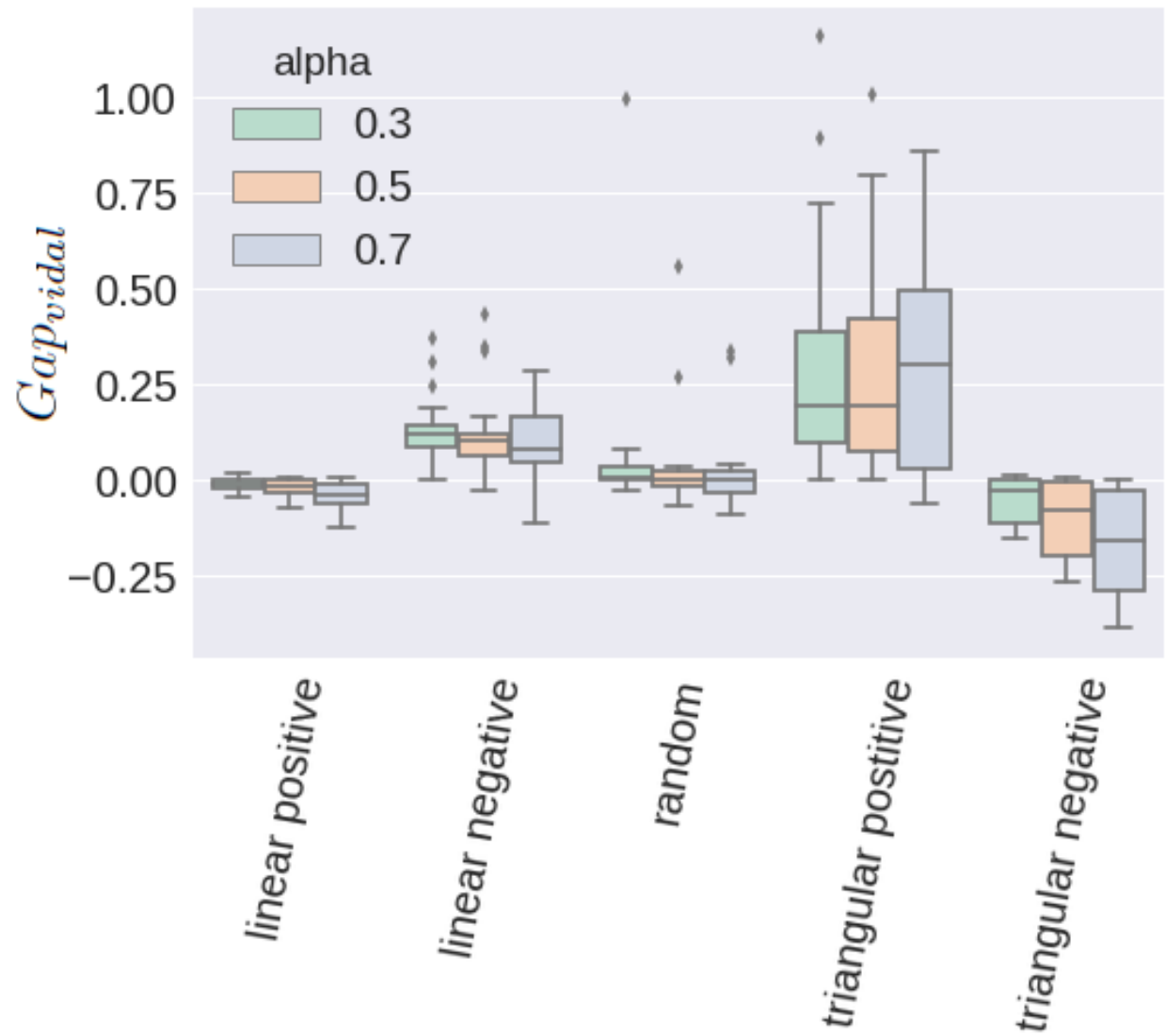


Figure 5.6 Box plots of sensitivity analysis for each α and for each function

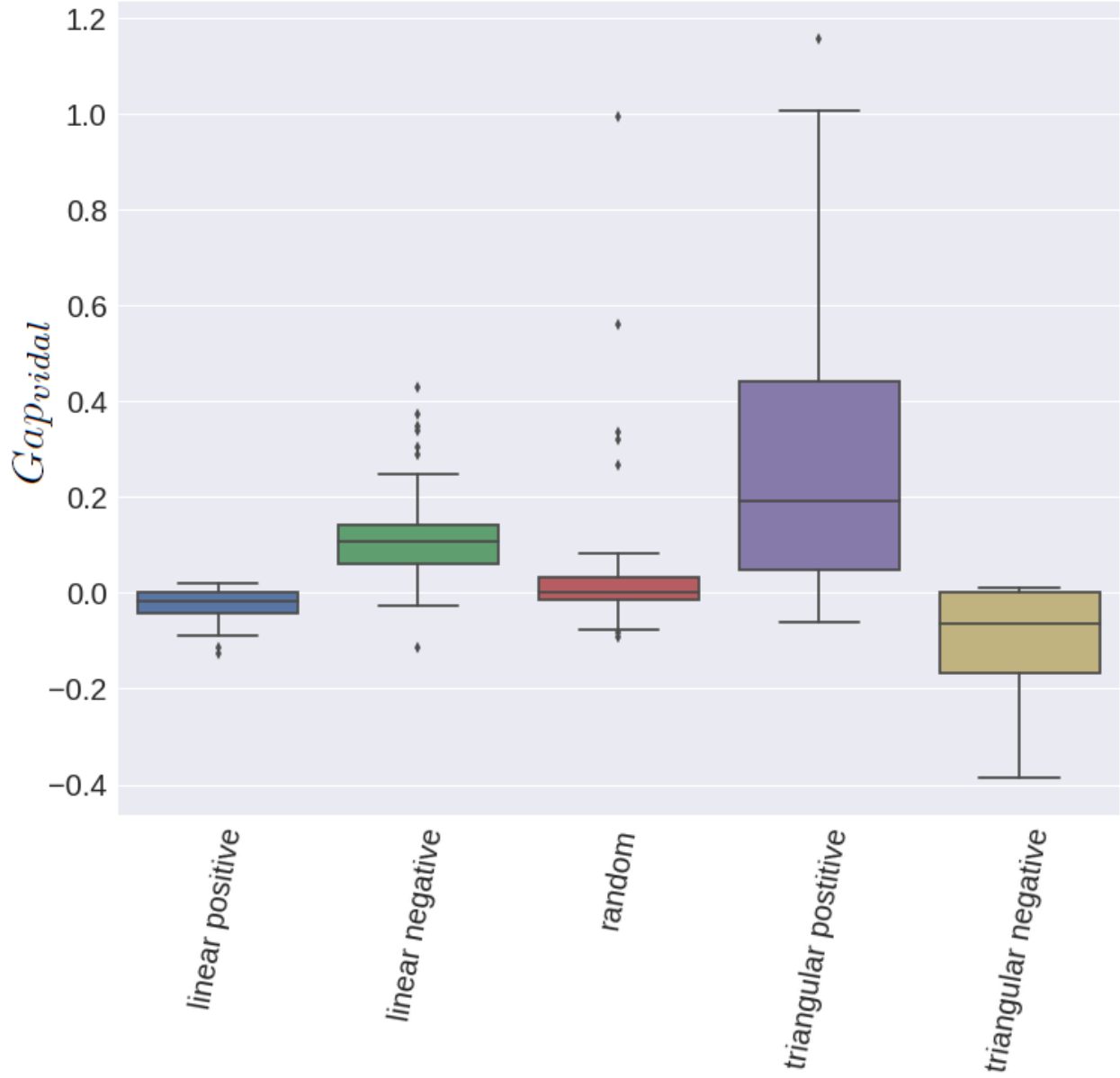


Figure 5.7 Box plots of sensitivity analysis for each function

5.6 Conclusion

This article introduces a new problem, the Mixed Capacitated General Routing Problem with Time dependent Demand TDD-MCGRP. This problem was inspired by winter salt spreading on roads when the quantity spread depends on the weather and the different periods of the day. New instances derived from known classic instances (BHW, CBMix, DI-NEARP) were

created using different functions to model the time dependent demand. A Mixed Integer Programming model was proposed to solve the small instances using CPLEX. The medium- and large-scale instances were solved using a SIRSs metaheuristic with some modification in order to be adapted to the TDD-MCGRP. The modified SIRS shown good results on classic instances with fixed demand with a gap of 3% on average to the best known solution. A sensitivity analysis was conducted showing a significant variation on the objective with different time dependent demand functions even with a small perturbation rate. This huge variation on the objective is due to the road imbalance. The limitation of this work is that to apply this study one must have a good prediction model. Further research along this direction of work would be using robust optimization in order to study demand uncertainty when the prediction of demand over time is not available.

CHAPITRE 6 ARTICLE 3: ROBUST OPTIMIZATION FOR THE HIERARCHICAL MIXED CAPACITATED GENERAL ROUTING PROBLEM APPLIED TO WINTER ROAD MAINTENANCE

C. Ahabchane, A. Langevin et M. Trépanier

Cet article a été soumis pour publication dans *Computers & Industrial Engineering*.

6.1 Abstract

This paper studies the Mixed Capacitated General Routing Problem (MCGRP) under a demand uncertainty and service hierarchy using a robust optimization approach. The problem is motivated by an industrial problem : the winter road salt spreading with street hierarchy and demand variation due to the weather or traffic conditions. The street hierarchy or the street priority is modeled with time dependent cost. We present a robust counterpart formulation with graph transformation to node routing in order to optimize the worst case realization of the demands from the uncertainty set. We use CPLEX to solve small instances and we developed a variant of the Slack Induction by String Removals metaheuristic for large-scale instances called the Robust SISRs. In the computational analysis we used Monte carlo simulation to study the robust approach on real-life case studies.

6.2 Introduction

Winter road maintenance is an important and essential activity to ensure the safe movement of citizens. However, it remains an extremely expensive industrial problem for municipalities. For example, the budget allocated to the city of Edmonton in 2019 for snow removal operations is 65M\$ ([Ville d'Edmonton, 2019](#)). Calgary budgeted 40M\$ in 2019 ([Ville de Calgary, 2019](#)). For Quebec City, snow removal operations require an annual budget of more than 43M\$ ([Ville de Quebec, 2020](#)). The city of Montreal increased the budget by 2.2M\$ in 2020 ([Ville de Montréal, 2019](#)) compared to the 2019 budget by 166M\$ ([Ville de Montréal, 2020a](#)).

In this paper, we study the mixed capacitated general routing problem with service hierarchy and in which the demand is uncertain. This problem is motivated by winter road network salt spreading application where streets have different priority levels, i.e., streets with a high priority level must be serviced first. However, some lower priority streets can be serviced with the higher priority streets, if it would not delay significantly the important street segments (Quirion-Blais *et al.*, 2015). Due to the lack of resources, most of snow removal agencies prioritize their response efforts in order to ensure the highest service level. Generally, they use traffic volume as a criterion in order to partition the road network into classes. Based on that, the street with heavy traffic are assigned the highest level of priority (Perrier *et al.*, 2007c). Many studies consider the importance of the service hierarchy in road winter maintenance (Campbell *et al.*, 1995b; Korteweg *et al.*, 2006; Perrier *et al.*, 2008; Campbell *et al.*, 2014b; Quirion-Blais *et al.*, 2015), recently Quirion-Blais *et al.* (2017) consider the service hierarchy in snow plowing operations with turn restrictions, route balancing, and variable vehicle speeds in a real large-scale network. Sullivan *et al.* (2019) not only highlighted the importance of the service hierarchy, but also showed that the level priority could change each day by introducing a novel measure of priority depending on the travel time and the vehicle flow. This study considers a new approach to modeling the service hierarchy instead of the classical approach: we propose to address the service hierarchy as time dependent cost. Time dependent cost was presented previously for the winter spreading application by Tagmouti *et al.* (2007), Tagmouti *et al.* (2010) and Tagmouti *et al.* (2011) the service cost was presented as the timing of the intervention (if the service of an arc is too early or too late, the cost in time and de-icing material increases) which is different from the service hierarchy cost.

Ahabchane *et al.* (2019) showed that the demand (quantity to spread) depends on street characteristics, weather, and traffic conditions. This demand variation can be modeled as time dependent demand if there is a good prediction model of the demand (Ahabchane *et al.*, 2020). When there is not enough data to develop a good prediction model, which happens in most real life applications, a robust optimization approach under demand uncertainty is a proper approach, especially when there are no information about the demand distribution.

The robust optimization approach was introduced firstly for convex optimization problems (Ben-Tal et Nemirovski, 1998; El Ghaoui *et al.*, 1998). It assumes that the uncertain parameters belong to a bounded uncertainty set (Ben-Tal et Nemirovski, 1998) or budget uncertainty set (Bertsimas et Sim, 2003) and aims to minimize the worst case, and immunize against the uncertainty (Ordóñez, 2010).

The robust vehicle routing problem (RVRP) was introduced by Sungur *et al.* (2008) who showed the relevance of using a robust approach. To handle the demand uncertainty, they used three types of bounded sets (convex hull, box, ellipsoidal). A real life application for the robust optimization was introduced later in Sungur *et al.* (2010) for the courier delivery problem, a variant of the vehicle routing problem with time windows (VRPTW) where the service times are uncertain. Ordóñez (2010) extended the previous work of Sungur *et al.* (2008) by considering multiple sources of uncertainty (costs, demand, time, customers) and correlation between uncertain coefficients, for which they presented different robust models.

Several exact methods were applied to tackle the RVRP. The column generation approach was used for the first time on robust optimisation by (Lee *et al.*, 2012) for the vehicle routing problem with deadlines (VRPD); the demand and travel time uncertainty were encapsulated into the sub-problem called the robust shortest path problem with resource constraints (RSP-PRC). The cutting plane approach was introduced for the VRPTW by (Agra *et al.*, 2013) with travel time uncertainty which was an extension of their previous work (Agra *et al.*, 2012). The branch-and-cut approach was firstly introduced by (Sungur *et al.*, 2008) and improved by adding Robust Rounded Capacity Inequalities (RCI) by Gounaris *et al.* (2013) for the (CVRP) under demand uncertainty. Lu et Gzara (2019) proposed the branch-and-price-and-cut algorithm for the CVRTW under demand uncertainty. They also proposed a new separation procedure that solves a robust bin packing problem. All these studies are based on dualization formulation to obtain the corresponding robust counterpart model. Jaillet *et al.* (2016) and Adulyasak et Jaillet (2016) proposed a new robust optimisation framework based on some information on the probability distributions of the random variables described by an interval and a mean value. In addition they proposed a new criterion, termed the Re-

quirements Violation Index, which evaluates the violation risk of a solution. Recently [Munari et al. \(2019\)](#) developed a novel compact robust optimisation formulation based on dynamic programming, instead of using a dualization scheme.

On the other hand, heuristics has also a considerable part on RVRP. [Moghaddam et al. \(2012\)](#) used Particle Swarm Optimization for the CVRP with demand uncertainty and results are compared with the ones obtained by [Sungur et al. \(2008\)](#). An ant colony optimization metaheuristic applied for the CVRP with uncertain travel costs was proposed by [Toklu et al. \(2013a\)](#) [Toklu et al. \(2013b\)](#). [Cao et al. \(2014\)](#) used the differential evolution algorithm for the the open vehicle routing problem with uncertain demands, and four robust strategies were proposed to handle the uncertain demand. [Gounaris et al. \(2016\)](#) extended their previous work by using adaptive memory programming metaheuristic instead of exact algorithm developed previously. [Solano-Charris et al. \(2015\)](#) used a Local Search Based Metaheuristic with an approach to handle the uncertainty differently from the bounded set or budget set. They proposed a set of discrete scenarios for the CVRP with uncertain arc costs by using the lexicographic min–max criterion. The min-max criterion on the travel cost was proposed also by [Bederina et Hifi \(2018\)](#) using hybrid multi-objective evolutionary algorithm. [Braaten et al. \(2017\)](#) proposed a variant of adaptive large neighborhood search for the VRPTW with uncertain travel time. [De La Vega et al. \(2019\)](#) extended the Solomon heuristics [Solomon \(1987\)](#) to address the vehicle routing problem with time windows and multiple deliverymen with uncertain demands. [Hu et al. \(2018\)](#) designed a two-stage algorithm based on a modified adaptive variable neighborhood search heuristic for the VRPTW under demand and travel time uncertainty based on route-dependent uncertainty sets. Recently [Subramanyam et al. \(2020\)](#) developed a robust local search for the classical Heterogeneous Vehicle Routing Problem (HVRP) with demand uncertainty, the robust local search was incorporated into two metaheuristic (Iterated Local Search and Adaptive Memory Programming), and tested on five popular classes of uncertainty sets. The quality of the metaheuristic solutions is quantified using branch-and-cut algorithm that provides lower bounds on the optimal solution.

Robust arc routing problem has been addressed only in two papers: [Chen et al. \(2016\)](#) intro-

duced the arc routing problem with service time uncertainty for the road daily maintenance service and compared the robust optimization approach against the chance-constraint programming method using a branch-and-cut algorithm. A periodic capacitated arc routing problem (PCARP) with demand uncertainty for the waste management application was proposed by [Tirkolaei et al. \(2018\)](#) and solved using an improved hybrid simulated annealing algorithm.

Our study presents a robust counterpart formulation using the budget set uncertainty for the Hierarchical MCGRP problem with a graph modification, and a robust slack induction by slack removals (SISRs) metaheuristic inspired by [Christiaens et Vanden Berghe \(2020\)](#). The mathematical model is used to evaluate the metaheuristic and then we use the developed metaheuristic to solve real instances from two mid-size Canadian cities: Granby and Saint-Jean-sur-Richelieu.

The main contributions of this paper are:

- To address the hierarchy with time dependent cost which is different from the classical approaches (min-max formulation)
- To apply robust optimization to address the MCGRP for the first time.
- To consider the uncertainty of the demand (quantity to spread on each segment of streets) in winter road maintenance with real data.

The remainder of the paper is organized as follows: Section [6.3](#) presents a graph transformation from MCGRP to VRP with a mathematical model, then a robust counterpart formulation is developed with budget uncertainty set. A brief description of the metaheuristic is presented in Section [6.4](#). Section [6.5](#) presents the computational results in which we evaluate the performance of the metaheuristic, followed by two real-life case studies. Final conclusions are drawn in Section [6.6](#).

6.3 Formal description and mathematical formulations

This section starts with a description of the graph transformation from MCGRP to VRP inspired by [Monroy-Licht *et al.* \(2014\)](#) and then a mathematical model is presented followed by a robust formulation based on [Chen *et al.* \(2016\)](#).

6.3.1 Formal description and graph transformation

The MCGRP can be defined on a mixed graph $G(V, A, E)$, where V is the set of vertices where vertex 1 represents the depot, A is the set of arcs, E is the set of edges. Each link $(i, j) \in A \cup E$ has a cost c_{ij} , additionally each required link $(i, j) \in A_R \cup E_R$ (A_R and E_R represent the required arcs and edges respectively) has a demand d_{ij} and each vertex $i \in V_R$ has a demand d_i .

From the graph G , we define a new asymmetric graph $G' = (N', A')$, where the vertices of N' correspond to the required elements of G in the following way: each required arc and vertex plus the depot in G are replaced by a vertex in G' , and each required edge in G is replaced by two vertices i, i' in G' , where $i, i' \in N'$ represent the two possible directions to travel on the edge. We have $|N'| = |V_R| + |A_R| + 2|E_R| + 1$, the vertex 1 refers to the depot. All the vertices in N' are connected to each other by arcs in A' except the vertices i and i' corresponding to the same edge. The distance from vertex i to vertex j in G' is equal to the shortest path in G from the end vertex of the arc or the edge or from the vertex represented by i to the end vertex of the edge or arc or to the vertex represented by j . Figure 6.1 shows an example of the transformation: Required vertex 5 in G is replaced by vertex 5 in G' , the arc (1,3) in G is replaced by vertex 7 in G' , vertex 8 represents the edge (4,6) traveled from 4 to 6 and vertex 8' presents the other direction (6,4).

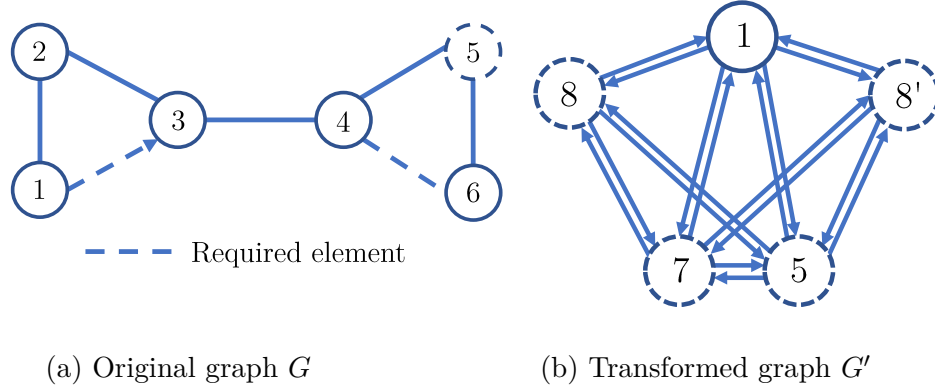


Figure 6.1 Graph transformation

6.3.2 Mathematical formulation

In the three-index vehicle flow formulation, we define the binary decision variable x_{ij}^k that takes value 1 if and only if node j is served immediately after node i by vehicle k , for $i, j \in N'$, where $k \in K$, the set of vehicles with a capacity Q . t_i represents the service starting time at node $i \in N'$. c_{ij} and t_{ij} are respectively the cost and time duration from node i to j . $\mathcal{F}(\cdot)$ is the cost function for a given starting time which is explained in the next subsection.

$$\text{Min } \sum_{k \in K} \sum_{i \in N'} \sum_{j \in N'} c_{ij} x_{ij}^k + \sum_{k \in K} \sum_{i \in N'} \mathcal{F}(t_i) \quad (6.1)$$

s.t.

$$\sum_{k \in K} \sum_{j \in N'} (x_{ij}^k + x_{i'j}^k) = 1, \quad \forall (i, i') \in N' \quad |i, i' \text{ correspond to the same edge } e \in E_R \quad (6.2)$$

$$\sum_{k \in K} \sum_{j \in N'} x_{ij}^k = 1, \quad \forall i \in N' \quad |i \text{ does not correspond to edge } e \in E_R \quad (6.3)$$

$$\sum_{i \in N'} x_{ih}^k - \sum_{j \in N'} x_{hj}^k = 0, \quad \forall h \in N', k \in K \quad (6.4)$$

$$\sum_{k \in K} \sum_{j \in N'} x_{1j}^k \leq |K|, \quad (6.5)$$

$$t_j \geq t_i + t_{ij} - M(1 - x_{ij}^k) \quad \forall i, j \in N', k \in K \quad (6.6)$$

$$\sum_{i \in N'} \sum_{j \in N'} d_i x_{ij}^k \leq Q, \quad \forall k \in K \quad (6.7)$$

$$x_{ij}^k \in \{0, 1\}, \quad \forall i, j \in N', k \in K \quad (6.8)$$

$$t_i \in \mathcal{R}^+, \quad \forall i \in N' \quad (6.9)$$

The objective function (6.1) minimizes the total cost of the routes and the cost corresponding to the starting time service. Constraints (6.2) ensure that only one node is included in the tour for each pair of nodes that represent the same required edge. Constraints (6.3) ensure that all the other nodes are visited exactly once. Constraints (6.4) guarantee the correct flow of vehicles through the arcs. Constraint (6.5) limits the maximum number of routes to $|K|$, the number of vehicles. Constraints (6.6) define the time constraints. They ensure that the service starting time for node j must be more than the service starting time for node i plus the travel time. Constraints (6.7) ensure that the vehicle capacity is not exceeded.

6.3.3 Linearization of time dependent cost

The purpose of using a time dependent cost function $\mathcal{F}(\cdot)$ is to simulate service priorities. The network of roads is partitioned into class priorities. These priorities are generally presented in literature with a lexicographic objective which is expressed in terms of a weighted finishing time of each priority class. The weights of each priority is determined by local authorities and present the importance of this class (Quirion-Blais *et al.*, 2017). This study considers

the class priority with a linear function $\mathcal{F}(\cdot)$ of the service starting time on each i (see Figure 6.2). For a given node i the cost function $\mathcal{F}(t_i) = \alpha_p t_i$, where α_p the weight of the class priority p . Thus the costs increase more quickly for the higher priorities. It is necessary to serve the nodes with the highest priority as soon as possible in order to reduce the costs.

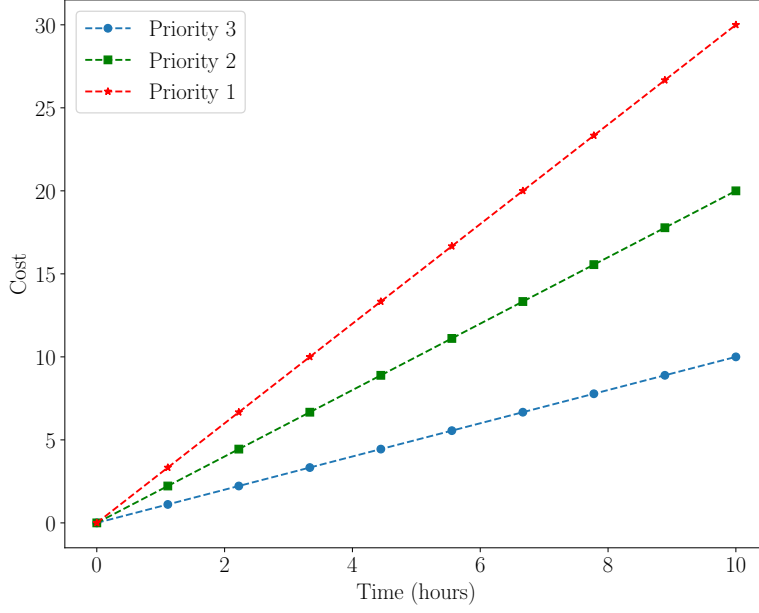


Figure 6.2 Cost functions

6.3.4 Robust formulation

We propose a robust optimization formulation in order to deal with the demand uncertainty, the aim of this robust optimization being to construct routes that take into account the variation of the demand.

The variation of demand is presented by an uncertainty set U^k (Equation 6.10) for each route. For each route, we define a budget uncertainty Γ^k in order to limit the accumulation of the deviation, which means that we suppose that the total variation of the demand cannot exceed some threshold Γ^k on each route. This demand variation is bounded in an interval $[\bar{d}_i, \bar{d}_i + \gamma_i^{max}]$, where $\bar{d}_i$ is the nominal demand of node i and γ_i^{max} is the maximum demand

deviation on the node i . The budget of uncertainty used in this study is different from the classical definition in [Bertsimas et Sim \(2003\)](#) ; we define our budget of uncertainty as:

$\Gamma^k = \varepsilon \frac{\theta^k \sum_{i \in N'} \gamma_i^{max}}{|K|}$, where ε presents the level of robustness $\varepsilon \in [0, 1]$, $\varepsilon = 1$ refers to the maximum protection against uncertainties, and the absence of uncertainties when $\varepsilon = 0$, θ^k presents the level of spreading, that reflects the driver behavior of the vehicle k compared to the average spreading of all the drivers.

$$U^k = \{d \in R^{|N'|} : d_i = \bar{d}_i + \xi_i \gamma_i^{max}, 0 \leq \xi_i \leq 1, \forall i \in N'; \sum_{i \in N'} \xi_i \gamma_i^{max} \leq \Gamma^k\} \quad (6.10)$$

We obtain a robust formulation if we replace constraints (6.7) in the original model by :

$$\sum_{i \in N'} \sum_{j \in N'} d_i x_{ij}^k \leq Q, \quad \forall k \in K, d \in U^k \quad (6.11)$$

which is equivalent to :

$$\max_{d \in U^k} \left(\sum_{i \in N'} \sum_{j \in N'} d_i x_{ij}^k \right) \leq Q, \quad \forall k \in K \quad (6.12)$$

Going through the dualization, we obtain from

$$\max_{\xi_i} \sum_{i \in N'} \sum_{j \in N'} \xi_i \gamma_i^{max} x_{ij}^k$$

s.t.

$$\xi_i \leq 1, \forall i \in N'$$

$$\sum_{i \in N'} \xi_i \gamma_i^{max} \leq \Gamma^k$$

$$\xi_i \geq 0, \forall i \in N'$$

Its dual:

$$\begin{aligned}
& \min_{\alpha, \beta} \alpha^k \Gamma^k + \sum_{i \in N'} \beta_i^k \\
& \text{s.t.} \\
& \alpha^k \gamma_i^{max} + \beta_i^k \geq \gamma_i^{max} \sum_{j \in N'} x_{ij}^k, \forall i \in N', k \in K \\
& \alpha^k \geq 0, \forall k \in K, \\
& \beta_i^k \geq 0, \forall k \in K, i \in N'
\end{aligned}$$

Constraints (6.12) can then be written:

$$\sum_{(i,j) \in N'} \bar{d}_i x_{ij}^k + \min_{\alpha, \beta} (\alpha^k \Gamma_k + \sum_{i \in N'} \beta_i^k) \leq Q, \quad \forall k \in K \quad (6.13)$$

s.t.

$$\alpha^k \gamma_i^{max} + \beta_i^k \geq \gamma_i^{max} \sum_{j \in N'} x_{ij}^k, \forall i \in N', k \in K \quad (6.14)$$

$$\alpha^k \geq 0, \forall k \in K, \quad (6.15)$$

$$\beta_i^k \geq 0, \forall k \in K, i \in N' \quad (6.16)$$

Constraints (6.13) can then be written:

$$\sum_{i \in N'} \sum_{j \in N'} \bar{d}_i x_{ij}^k + \alpha^k \Gamma_k + \sum_{i \in N'} \beta_i^k \leq Q, \quad \forall k \in K \quad (6.17)$$

The complete robust formulation is:

$$\text{Min} \sum_{k \in K} \sum_{i \in N'} \sum_{j \in N'} c_{ij} x_{ij}^k + \sum_{k \in K} \sum_{i \in N'} \mathcal{F}(t_i^k)$$

subject to : (6.2-6.6), 6.17, (6.14-6.16), (6.8-6.9).

6.4 Robust SISRs

This study uses a robust version of the modified SISRs (Slack Induction by String Removals) developed in [Ahabchane *et al.* \(2020\)](#), based on the Classical SISRs [Christiaens et Van-den Berghe \(2020\)](#). The SISRs metaheuristic, which is based on simulated annealing and Ruin and Recreate Strategies, had demonstrated a good performance on different variants of vehicle routing problems and also on arc routing problems. The robust SISRs contains two phases :

- **Ruin phase:** It starts by choosing a random client from a random route, and from this client K adjacent clients from different routes are chosen. From every client chosen, a string of L adjacent clients is created from the same route (a string is a part of the solution that contains adjacent clients from the same route). All these chosen clients are removed from the solution. L and K are determined by equations that depend on the average size of strings and clients to remove.
- **Recreate phase:** This phase fills the slack induced by the the removed strings into the ruined solution. The removed clients are sorted by distance or demand, then a stochastic best insertion heuristic is applied, i.e., the removed clients are inserted into the best position with a probability of 99% as in the original SISRs. After this phase a local search can be added if the results are not satisfactory.

The robust SISRs has two main contributions different from other metaheuristics; the first contribution is introduction of the robust heuristic into the recreate phase, the second one is consideration of the time dependent cost and the demand deviation into the objective function.

6.4.1 Robust heuristic

To deal with the demand uncertainty, we introduce the robust heuristic into the SISRs; the concept behind the robust heuristic is inspired by the uncertainty set U^k (Equation 6.7). The robust heuristic (explained in Algorithm 2) aims at finding the values of ξ_i that maximize the sum of demands on vehicle k while respecting the uncertainty budget constraint. The

objective function aims also at minimizing the difference between the uncertainty budget Γ^k and the sum of deviations $\sum_{j=1}^n \xi_j \gamma_j^{max}$.

Algorithm 2 starts at the recreate phase. For each inserted client i in a route k , the sum of the deviations from this route is calculated. If it exceeds the uncertainty budget Γ^k the variable ξ_i takes the value 0. Otherwise, if the difference between the sum of the deviations and the budget uncertainty is less than the maximum deviation of client i γ_i^{max} , then ξ_i is equal to this difference divided by the maximum deviation, otherwise ξ takes a random value between 0.5 and 1.

Algorithm 2: The robust heuristic

Input: Chosen client i for insertion in a route k containing n clients

```

1 Robust heuristic :
2   if  $\sum_{j=1}^n \xi_j \gamma_j^{max} \geq \Gamma^k$  then
3      $\xi_i = 0$ 
4   else
5     if  $\frac{\Gamma^k - \sum_{j=1}^n \xi_j \gamma_j^{max}}{\gamma_i^{max}} \leq 1$  then
6        $\xi_i = \frac{\Gamma^k - \sum_{j=1}^n \xi_j \gamma_j^{max}}{\gamma_i^{max}}$ 
7     else
8        $\xi_i = \text{Uniform}(0.5, 1)$ 
9     end
10  end
11  add client  $i$  to  $k$  and update the demand of route  $k$ 
Output: insertion of  $i$  in route  $k$ 

```

6.4.2 Objective function

Since this study deals with service priorities and demand uncertainty, the objective function considers the total distance, service priorities as a time dependent cost, and the demand deviation. The service priorities as explained above are represented by a linear function of the service time. The objective function aims at minimizing the time dependent cost by minimizing the weighted starting time for each street segment (each priority class have a different weight). The robust approach is also considered in the objective function by minimizing the gap between the uncertainty budget and the sum of deviations, which is

exactly the same as maximizing the uncertain demand on each route.

Algorithm 3: The objective function algorithm

Input: Solution of $|K|$ routes

Objective function evaluation :

```

1   $Cost \leftarrow 0$ 
2  for route  $R_k$  in  $K$  do
3       $deviation_k \leftarrow 0$ 
4       $i \leftarrow 1$  //  $i$  presents the order of visit in route  $k$ ,  $i=0$  refers to the depot
5       $Cost_k \leftarrow 0$ 
6      for client  $i \in R_k$  do
7          // clients are sorted in order of passage
8           $t_i = t_{i-1} + t_{i-1,i}$  //  $t_{ij}$  presents the travel time from  $i$  to  $j$ 
9           $w_i \leftarrow$  the weight of the class priority  $p$  // where  $p$  is a class priority of
10              $i$ 
11           $Cost_k \leftarrow Cost_k + d_{i-1,i} + w_i t_i$  //  $d_{ij}$  presents the distance from  $i$  to  $j$ 
12           $deviation_k \leftarrow deviation_k + \epsilon_i \gamma_i^{max}$ 
13      end
14       $Cost_k \leftarrow Cost_k + d_{i,0}$ 
15  end
16   $Cost \leftarrow Cost + \lambda(\Gamma^k - deviation_k)$  //  $\lambda$  presents the penalty coefficient

```

Output: Updated cost

6.5 Computational experiments

Before solving the real life applications with the Robust SISRs, its performance is evaluated. The Robust SISRs is compared with the exact formulation presented in Section 6.3.4. For that, we produced small-sized instances based on Quirion-Blais *et al.* (2017) approach by selecting between 10 and 15 required elements (arcs, edges or vertices) from the real instances with different priorities. Then a graph transformation is applied on the selected elements. We also consider the variation of the maximum deviation $\gamma_i^{max} = \delta d_i$ where $\delta \in \{0.3, 0.5, 0.7\}$, in order to study the impact of different deviation on the results. Since we do not have enough data to calculate the coefficient θ^k on the uncertainty budget Γ^k , we consider that $\theta^k = 1$ for each vehicle. A maximum running time limit of 7200 seconds was imposed in CPLEX for all tests. The characteristics of the instances and the results are described in Table 6.1. Table 6.1 indicates the average result and the average gap of 10 runs of the Robust SISRs (the same parameters of the original SISRs are used), the best result and the best gap of the 10 runs, and the average time (in seconds). The optimal values are highlighted in bold.

The results showed that metaheuristic reaches the optimal solution quickly for all instances except one. The purpose of the performance tests is not only to evaluate the metaheuristic

Table 6.1 – Comparison of CPLEX results and metaheuristic results

Instance name		CPLEX					Modified SISRs				
		$ E_R $	$ A_R $	$ V_R $	Objective	Time(s)	Avg-10	Avg Gap(%)	Best-10	Best Gap(%)	Avg time(s)
$\delta = 0.3$	G10	3	5	2	788	1656	790.7	0.34	788	0.0	24
	G11	4	6	1	1206	3450	1208.4	0.18	1206	0.0	28
	G12	2	9	1	1976	5524	1989.2	0.65	1976	0.0	33
	G13	0	10	3	1676	17	1677.5	0.09	1676	0.0	27
	G14	2	10	2	1043	4320	1050.1	0.68	1043	0.0	29
	G15	1	10	4	1776	5348	1792.3	0.91	1776	0.0	38
$\delta = 0.5$	G10	3	5	2	863	1384	865.2	0.25	863	0.0	27
	G11	4	6	1	1206	6850	1221.2	1.26	1206	0.0	28
	G12	2	9	1	2045	6515	2056.7	0.57	2045	0.0	34
	G13	0	10	3	1676	18	1679.7	0.22	1676	0.0	28
	G14	2	10	2	1057	5320	1064.8	0.74	1057	0.0	35
	G15	1	10	4	1776	5923	1811.5	1.99	1776	0.0	42
$\delta = 0.7$	G10	3	5	2	837	2497	841.9	0.58	837	0.0	26
	G11	4	6	1	1206	5450	1210.6	0.31	1206	0.0	32
	G12	2	9	1	2252	7200	2271.2	13.56	2252	11.0	44
	G13	0	10	3	1677	14	1676.8	0.06	1677	0.0	31
	G14	2	10	2	1253	6580	1293.4	3.22	1253	0.0	38
	G15	1	10	4	1928	6358	1954.8	1.72	1928	0.0	47

but also to validate the robust heuristic used and show that the Robust SISRs converges to good solutions. The results showed that the instances with many edges take longer, because edges turn into two nodes with the graph transformation, which greatly increases the size of the problem. For more than 15 elements CPLEX cannot find the optimal solution within 7200 seconds. We can observe the impact of the maximum deviation on the cost, when the deviation increases: the cost also increases, which means that the solution is more immunized against uncertainties.

6.5.1 Monte Carlo simulation

Since the demand is uncertain, using Monte Carlo simulation is essential to evaluate the performance of robust solutions under uncertainty. The sensitivity analysis is conducted by generating 10,000 random uniform realization of demands in the interval $[\bar{d}_i, \bar{d}_i + \gamma_i^{max}]$. The objective of these simulations is to analyze the behaviour of the robust solutions in terms of the level of robustness ϵ defined on the uncertainty budget, because the extra cost of the robust solution serves to protect it against uncertainty, and it results in an increase in robustness.

We also analyse the trade-off between the cost and the level of robustness/conservativeness. To carry out these analyses we defined three performance measures:

1. **The price of robustness:** It represents the gap between the robust solution and the deterministic solution; it is used to calculate the relative increase in cost of the robust model compared to the deterministic model
2. **The service level:** It represents the quality of service of the robust solution; it measures the average ratio between the number of served elements and the number of required elements using Monte Carlo simulation.
3. **The estimated risk:** It represents the probability of constraint violation calculated as the frequency of infeasible solutions of all the realizations generated by the Monte Carlo simulation

These performance measures are used to evaluate the robust solutions found by the robust metaheuristic on real life applications, since we do not have the optimal solution.

6.5.2 Real-life case studies

The main objective behind this study is to find a good solution that can serve all the required elements of the city under demand uncertainty and respect the priority levels with an acceptable cost. We use real data containing some relevant information like the demand and the maximum deviation for each required elements and priority levels. The project was conducted with two Canadian cities:

- **Granby** is a town located in the province of Quebec, Canada. The complete network comprises approximately 661 km of traffic lanes, which once transformed into a graph network corresponds to 3280 arcs and 2250 nodes. Figure 6.3 shows the map of the studied network. We received data collected by the on-board systems of 12 spreading trucks. These data are recorded at each 0.5s from the various sensors on-board. Each record provides information on the date and time of collection, geographic position (longitude, latitude), salt and abrasives quantity and speed. The data covers the 2014-2015 season and includes 29.1 million records on 12 trucks.

- **Saint-Jean-sur-Richelieu (SJSR)** is the second largest city in Montérégie, located also in the province of Quebec, Canada. The Richelieu River divides the city in two. The complete network has about 870 km of traffic lanes, which once transformed into a graph network, corresponds to 6059 arcs and 4177 nodes. Figure 6.4 shows the plan of the studied network. We received data collected by the on-board systems of 75 vehicles. Each record provides information similar to Granby's. The data covers the 2015-2017 season and includes 26.1 million records.

In addition to the collected data, other information was provided by the cities like street length, number of lanes, street segment type. Both cities have 3 priority levels:

- **The first level:** Comprises the main roads of the city. They include inter-municipal roads, and the streets that connect them (arteries, strategic places and bus routes).
- **The second level:** Comprises streets which serve as transit between the local streets and the main arteries. This category includes streets with strategic places where intervention is required over a short distance for safety reasons such as streets with slopes, dangerous curves, dangerous intersections, streets with schools and streets where speed is higher than 50 km/h (collector streets, rural roads exposed to strong winds, strategic places, and bus routes)
- **The third level:** All other local streets, in residential and rural areas.

The experiments on the two cities were conducted with the Robust SISR with a running time of 1 hour. For each level of robustness we ran the metaheuristic 10 times and we took the median results, for which we ran the Monte Carlo simulation in order to calculate the performance measures. Tables 6.2 and 6.3 summarized the results of the Robust SISR and Monte Carlo simulation on Granby and SJSR respectively. The deterministic solution used to quantify the price of robustness is presented at the end of the Tables 6.2 and 6.3. In order to have a very clear vision of the results, the tables were expressed in the form of graphs (Figures 6.5 to 6.12).

We can notice that there is a negative correlation between the risk and the level of service (Figures 6.5 and 6.9). This was expected since increasing the risk generates infeasible solution

Table 6.2 – The results of robust SISR on Granby

Level of robustness	Risk(%)	Service level(%)	Cost	#Vehicles	Unified cost	Price of robustness(%)
0.1	100	75.4	375928	14	239226	70.99
0.2	100	78.6	455231	14	289692	107.06
0.3	93.3	80.4	452763	15	308702	120.65
0.4	87.5	83.2	421084	16	306242	118.89
0.5	64.7	88.7	404957	17	312921	123.67
0.6	55.55	88	377942	18	309225	121.03
0.7	38.88	91.5	387096	18	316714	126.38
0.8	31.5	94.1	354896	19	306501	119.08
0.9	14.2	98.4	343033	21	327440	134.05
1	4.5	99.4	326045	22	326045	133.05
Deterministic			219845	14	139901	0

Table 6.3 – The results of robust SISR on SJSR

Level of robustness	Risk(%)	Service level(%)	Cost	#Vehicles	Unified cost	Price of robustness(%)
0.1	100	77.4	369063	11	238805	60.83
0.2	90.09	79.1	415981	11	269164	81.28
0.3	90.09	82.2	452274	11	292647	97.1
0.4	66.66	87	459876	12	324618	118.63
0.5	50	92	475954	12	335967	126.28
0.6	46.15	94.4	441184	13	337376	127.22
0.7	42.8	93.3	405133	14	333638	124.71
0.8	35.71	94.7	412639	14	339820	128.87
0.9	18.75	95	380807	15	336006	126.3
1	6	98.2	349711	17	349711	135.53
Deterministic			252407	10	148474	0

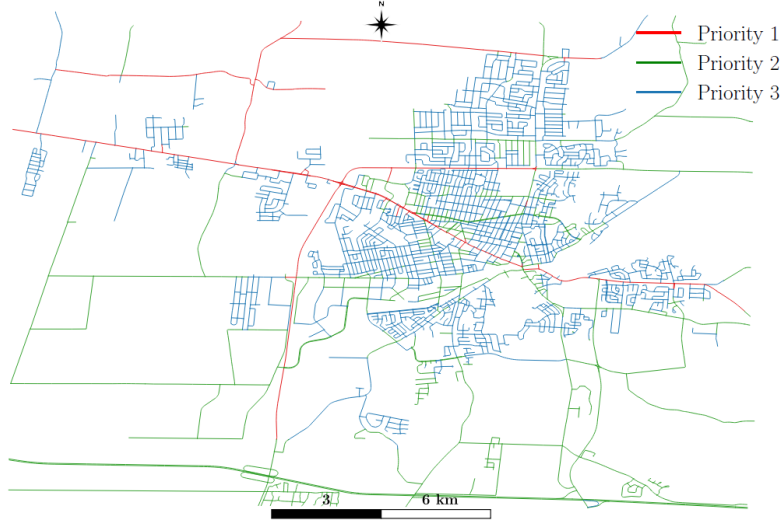


Figure 6.3 Granby, QC

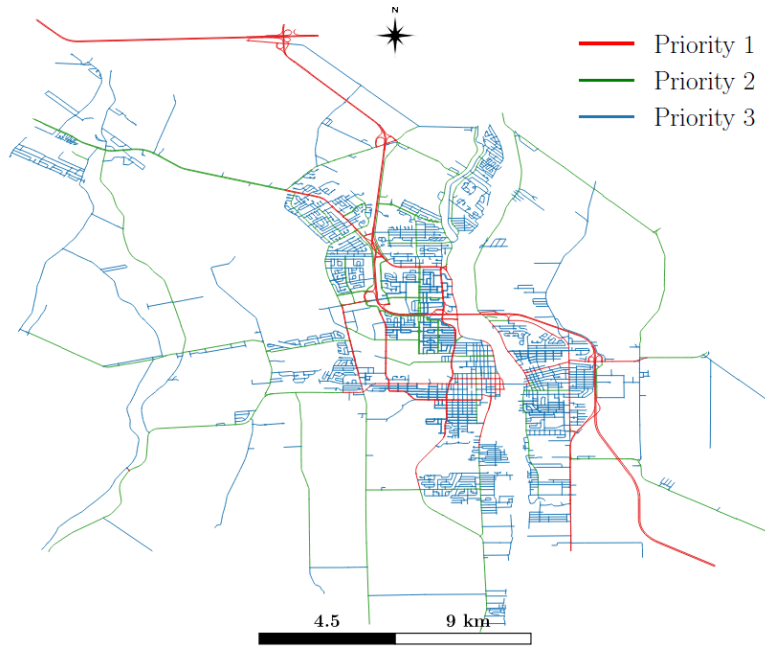


Figure 6.4 Saint-Jean-sur-Richelieu, QC

and consequently decreases the level of service. Moreover the service level is greater than 75.4% and 77.4% for Granby and SJSR, respectively, even for a non-protected solution when $\epsilon = 0.1$. However the risk decreases sharply when the level of robustness exceeds 0.4 and even for a well protected solution ($\epsilon = 1$) the risk still exists. Figures 6.6 and 6.10 shows the relationship between the cost and the number of vehicles; the increase in robustness level

implies more vehicles for better protection against uncertainty. However this cost does not really present the real cost of the solutions because the number of vehicles is not taken into account, that is why we propose to unify the costs according to the number of vehicles, by multiplying by the number of vehicles used and by dividing by the maximum number of vehicles. Figures 6.7 and 6.11 illustrates the difference for the unified cost and the standard cost, although this approach is not exact but at least it is closer to the real life situations; the cost increases with the increase in robustness. The price of robustness presented in Figures 6.8, 6.12 is calculated with unified costs; the price of robustness increases sharply when the level of robustness exceeds 0.2 and 0.4 for Granby and SJSR, respectively, and easily reaches the 100% and more, especially in Granby. It means we pay more than double of the deterministic cost which is huge, but it remains understandable since according to the data collected the average of the maximum deviation is twice as large as the average demand. This analysis with the performance measures provides a great insight for decision makers about the different solutions with the associated risks, service level, and cost for each solution. Furthermore this analysis allows the decision maker to understand the trade-off between level of robustness and other variables: cost, risk, service level, and number of required vehicles.

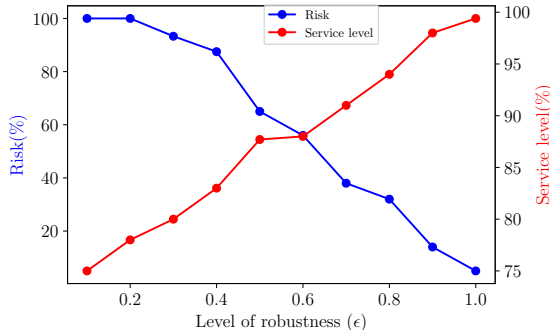


Figure 6.5 Level service and risk for Granby

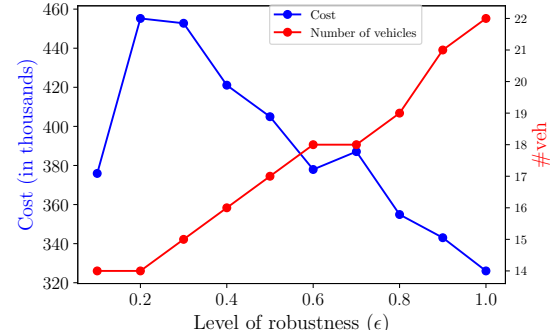


Figure 6.6 Cost and # of vehicles for Granby

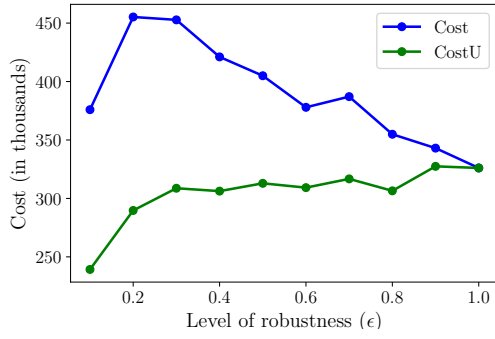


Figure 6.7 Cost and unified cost for Granby

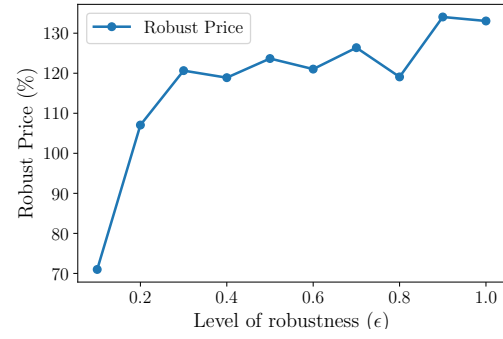


Figure 6.8 Price of robustness for Granby

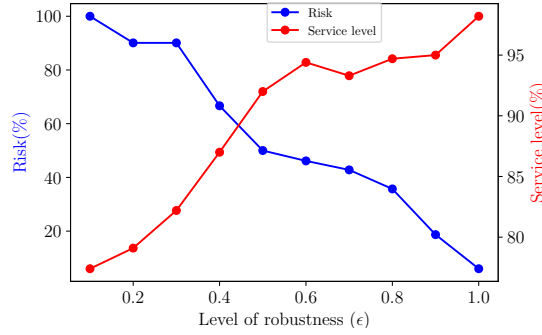


Figure 6.9 Level service and risk for SJSR

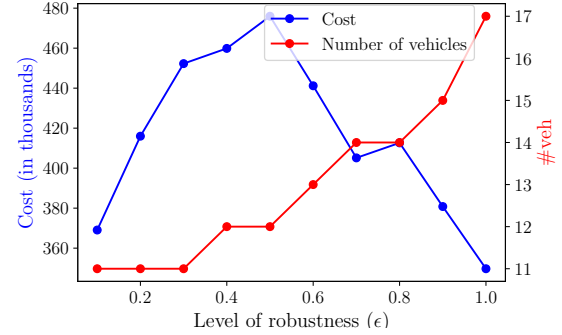


Figure 6.10 Cost and # of vehicles for SJSR

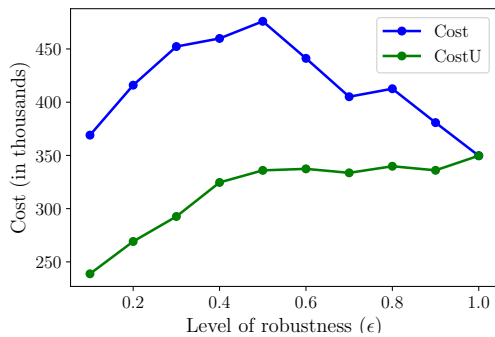


Figure 6.11 Cost and unified cost for SJSR

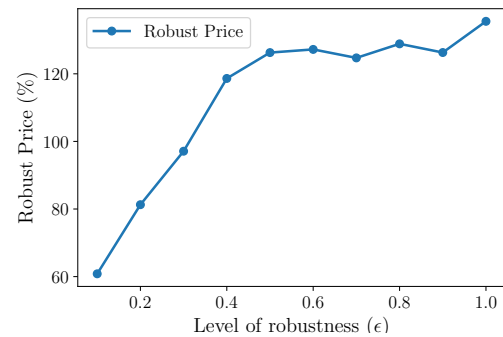


Figure 6.12 Price of robustness for SJSR

6.6 Conclusion

This study proposes a robust optimisation approach for the hierarchical mixed capacitated general routing problem with demand uncertainty. The real life application of this problem arises in winter road salt spreading in the presence of priority levels of the streets. A robust formulation has been developed with graph transformation, in order to solve this problem under demand uncertainty belonging to an uncertainty budget. Since this project deals with real life application, a Robust SISRS has been developed to handle large-scale instances. The exact formulation was used to validate the Robust SISRs on small-sized instances generated from our case studies. For the large-scale instances we used real data from two Canadian cities Granby and Saint-Jean-sur-Richelieu. Data was collected from the spreading trucks and topology information was provided by the cities. The computational experiments on real-life case study were conducted using Monte Carlo simulation. The relevance of the robust approach was emphasized by the performance measures analysis, which gives the managers a performance decision tool to assess different robust solutions and also to understand the trade-off between robustness and other indicators (cost, risk, service level).

CHAPITRE 7 DISCUSSION GÉNÉRALE

Cette thèse propose des contributions importantes en viabilité hivernale, particulièrement en opération d'épandage. Ces contributions sont présentées à travers trois articles : le premier article présente un développement d'un modèle de prédiction de la quantité de sel et d'abrasif basé sur une méthodologie d'apprentissage machine. L'étude commence par le nettoyage et la préparation des données de télémétries de 12 camions d'épandage, ce qui présente 29 millions de lignes de données. Ces données contiennent des informations sur la localisation des camions, la vitesse, la date et l'heure, et la quantité d'épandage. D'autres données ont été utilisées, tels que les données météorologiques (température de l'air, quantités de neige et de pluie, vitesse du vent, point de rosée...) collectées du Ministère du développement durable, de l'environnement et de la lutte contre les changements climatiques du Québec ainsi que du site web du Service météorologique du Canada. En outre, nous avons exploité des données concernant la géomatique de la ville: longueur de la rue, nombre de voies, type de segment de rue, etc. La deuxième phase a été l'extraction des variables (feature engineering). Cette étape consiste à trouver de nouveaux facteurs influant sur le modèle, tels que : les virages, les pentes, le nombre de passages des camions, la vitesse de circulation, les opérations d'épandage précédentes, l'accumulation de la température et de la neige, etc. Puisque le nombre de variables influence significativement le temps de calcul, une étape de réduction du nombre de variables (feature selection) était nécessaire. Pour ce faire, nous avons utilisé la méthode Lasso aléatoire. Cette étape permet de garder les variables les plus importantes, en passant de 46 variables à 35 variables. La dernière phase consiste à appliquer différents algorithmes d'apprentissage machine (les forêts aléatoires, les arbres extrêmement aléatoires, les réseaux de neurones artificiels, *Adaboost*, *Gradient Boosting Machine* et *XG-Boost*), en réalisant une validation croisée et différents réglages des hyperparamètres (Hyperparamètres Tuning). Les résultats prouvent la bonne performance de l'algorithme XGBoost ($R^2 = 0.83$; $MSE = 1,256.93$). Les variables les plus significatives sur la quantité d'épandage sont : le nombre de passages, la longueur des segments, le trafic, la quantité d'épandage précédente, le nombre total de passages, l'identité du camion, la largeur, la pression, la tem-

pérature et l'élévation. Les résultats présentés dans cet article ont été jugés intéressants par les gestionnaires. Premièrement, l'importance des variables montre que le facteur humain (les conducteurs) est parmi les facteurs les plus significatifs, ce qui pourrait améliorer les opérations d'épandage en contrôlant ce facteur. Deuxièmement, la répartition des erreurs de prédiction sur la carte de la ville permet au gestionnaire de localiser les segments présentant des erreurs de prédiction élevées. Par la suite, plusieurs politiques correctives peuvent être appliquées, comme la normalisation du comportement du conducteur. Enfin, le modèle de prédiction peut être utilisé pour optimiser les tournées d'épandage, car, on peut prédire la quantité nécessaire pour chaque segment et pour chaque heure. La confection de ces tournées avec la variation de la quantité d'épandage est l'objectif des deux articles suivants.

Le deuxième article est basé sur l'hypothèse qu'on a un bon modèle qui permet de prédire la quantité de sel et d'abrasif pour chaque heure et pour chaque segment de rue. L'article introduit un nouveau problème nommé le problème de tournées de véhicules générales avec contrainte de capacité dont la demande est dépendante du temps (MCGRP-TDD). Une formulation mathématique de ce problème a été présentée sans la modification du graphe, d'où la nécessité d'introduire la notion des étapes pour exprimer les contraintes du temps, puisqu'il est difficile d'associer un temps des début et de fin uniques à un arc ou à une arête. La demande qui représente la quantité d'épandage est présentée sous forme d'une fonction linéaire par morceaux. Une métaheuristique (SISRS modifiée) pour résoudre les grandes instances a été développée. Cette nouvelle métaheuristique est basée sur les stratégies de destruction et de construction. Elle a été inspirée par Christiaens and Vanden Berghe (2019). Nous avons apporté des modifications importantes, afin de prendre en considération la demande dépendante du temps, ainsi que l'évaluation des coûts et de la capacité en utilisant la programmation dynamique. La SISRS modifiée a montré une bonne performance sur les instances avec la demande fixe. Afin d'évaluer la formulation mathématique et la métaheuristique avec la demande dépendante du temps, des instances ont été créées basées sur des instances classiques (CBMix, BHW, DI-NEARP). Cinq différentes fonctions ont été proposées pour générer la demande en fonction du temps. Une analyse de sensibilité a été menée. Elle montre une variation significative de l'objectif avec les différentes fonctions, même en considérant un faible taux de perturbation par rapport à la demande moyenne.

Cette énorme variation de l'objectif est due au déséquilibre des tournées.

Le manque de données massives dans beaucoup de cas réels ne permet pas de développer un modèle fiable de prédiction, ni d'estimer la distribution de probabilité de la demande, pour appliquer l'optimisation stochastique. Dans ce cas, l'approche robuste présente une solution pratique pour traiter la variation de la demande en manque d'information. Le troisième article propose l'optimisation robuste pour traiter le problème de tournées de véhicules générales hiérarchique avec contrainte de capacité sous l'incertitude de la demande. La hiérarchie de service signifie que les rues ont différents niveaux de priorité. Les rues avec un niveau de priorité élevé doivent être desservies en premier. Cependant, certaines rues de priorité inférieure peuvent être desservies par les rues de priorité supérieure, si cela ne retarde pas considérablement les segments de rue importants. Une formulation mathématique robuste a été présentée, après avoir effectué la transformation du graphe vers un problème sur les nœuds. Cependant, l'utilisation de la dualisation était indispensable pour linéariser la contrainte de la demande incertaine. L'article présente aussi une métaheuristique SIRS robuste pour résoudre les grandes instances. La métaheuristique développée propose une heuristique robuste pour traiter l'incertitude de la demande en l'intégrant dans la fonction objective. Cette SIRS robuste a été validée par la formulation exacte, et par la suite elle a été appliquée sur les données réelles de deux villes canadiennes : Granby et Saint-Jean-sur-Richelieu. Les analyses expérimentales sur les deux villes ont été menées en utilisant la simulation de Monte-Carlo. La pertinence de l'approche robuste a été soulignée par l'analyse des mesures de performance. Cette dernière donne aux gestionnaires un outil de décision pertinent pour évaluer différentes solutions robustes, et également pour comprendre le compromis entre la robustesse et d'autres indicateurs comme : le coût des tournées, le risque d'avoir des tournées infaisables, et le niveau de service par rapport à la demande requise.

CHAPITRE 8 CONCLUSION ET RECOMMANDATIONS

8.1 Synthèse des travaux

Les opérations d'épandage sont modélisées généralement en recherche opérationnelle par un problème de CARP avec une demande fixe. Cette thèse propose quatre contributions majeures dans la modélisation des opérations d'épandage :

La première contribution est de considérer la variation de la quantité d'épandage durant la période des opérations, et d'extraire les variables influentes sur cette variation. Et par la suite, pour ce faire, nous développons un modèle de prédiction de sel et d'abrasif basé sur l'apprentissage machine.

La deuxième contribution consiste à suggérer deux modélisations pour traiter cette variation en fonction de la qualité des données. La première modélisation est une modélisation dépendante du temps dans le cas où on est capable de prédire la quantité de sel et d'abrasif pour chaque heure de passage de camions, et pour chaque segment de rue. La deuxième modélisation est une modélisation robuste dans le cas du manque de données.

La troisième contribution est de modéliser les opérations d'épandage par un problème de MCGRP au lieu de CARP. En effet, en pratique, certains endroits peuvent être considérés comme des nœuds, par exemple : des intersections, des virages, des arrêts de bus, des entrées d'hôpitaux et d'écoles.

La dernière contribution comporte l'inclusion du niveau de service dans l'objectif sous forme d'une fonction qui dépend du temps, au lieu de la présentation classique avec un objectif lexicographique.

La manipulation des données massives était indispensable dans cette thèse, premièrement, pour développer le modèle de prédiction ; et deuxièmement, pour tester les algorithmes développés. Ces données provenaient des capteurs des camions d'épandage, des sites météorologiques et des topologies des deux villes Granby et Saint-Jean-sur-Richelieu. Toutefois, l'utilisation des données réelles dans cette thèse nécessite beaucoup d'efforts. Premièrement,

la quantité de données de télémétrie collectées des camions était énorme, environ 29 millions lignes de données brutes. Il fallait les attribuer aux segments de rues appropriés et les nettoyer par la suite. De plus, il fallait considérer les données météorologiques horaires et extraire les caractéristiques des rues comme les virages et les pentes. Deuxièmement, la transformation du réseau géographique des deux villes vers un réseau mathématique est compliquée dues aux erreurs de topologie (absence d'intersection, des arcs pendants, des pseudo nœuds, des îlots isolés. . .).

8.2 Limitations des solutions proposées

Une des limitations de notre modèle de prédiction est l'hypothèse que les conducteurs des camions d'épandage sont en mesure d'assurer un bon niveau de service sans trop épandre. Un surépandage de sel a un impact environnemental négatif ; cependant, il n'y a pas des mesures pour évaluer la qualité d'épandage ou pour évaluer le niveau de service. Une autre limitation est liée à la nature du modèle de prédiction. En effet, le modèle ne prévoit pas les quantités de sel et d'abrasif séparément ; car les données couvrent seulement deux saisons hivernales, ce qui n'est pas suffisant pour avoir de très bonnes prédictions. Cependant, le facteur température permet de tenir compte du type à épandre.

Les limites des modèles mathématiques sont liées à la taille des instances qu'on est en mesure de résoudre de façon exacte. Le premier modèle qui traite la demande dépendante du temps ne peut pas résoudre plus de 30 éléments requis (arcs, arêtes, nœuds). Le deuxième modèle qui traite l'incertitude de la demande avec les niveaux de priorité ne peut pas résoudre plus de 15 éléments requis.

En analysant les résultats du deuxième objectif qui consiste à développer un programme mathématique modélisant le problème de tournées de véhicules pour les opérations d'épandage dont la quantité dépend du temps, nous avons remarqué une limite qui est due au déséquilibre des tournées. Cela engendre une différence significative entre les objectifs des différentes fonctions de la demande.

L'utilisation des métaheuristiques au lieu des méthodes exactes est considérée comme une limitation de la solution proposée, ce qui ne permet pas d'avoir une idée sur la qualité de

notre solution. Cependant, les approches heuristiques restent plus efficaces et performantes en pratique : premièrement grâce à leur temps de résolution, deuxièmement grâce à leur implémentation qui ne demande pas de logiciels commerciaux, et dernièrement grâce à leur flexibilité pour toute adaptation.

8.3 Améliorations futures

Une avenue de recherche intéressante sera de proposer des méthodes de résolution exacte comme la génération de colonne ou le branch-and-cut pour les deux problèmes proposés. Une autre piste pertinente pour amélioration serait d'intégrer le temps de service dépendant du temps dans les deux problèmes, ce qui correspond davantage à la réalité des opérations d'épandage, dont le niveau de service dépend aussi du temps de déplacement. Une autre amélioration importante serait de comparer l'approche robuste avec une approche stochastique en présence de données suffisantes. Certes, l'approche stochastique avec des informations sur la distribution de probabilité serait intéressante. Finalement, la construction des tournées dynamiques qui prendront en compte des modifications en temps réel dans les opérations d'épandage (accidents et pannes, changement des conditions météorologiques, congestion routière, etc.). Afin d'assurer l'aspect dynamique, la méthode de résolution devra être relativement rapide pour fournir des résultats dans un temps raisonnable. De plus, elle devra être robuste afin de garder une certaine constance par rapport aux réévaluations suite à des mises à jour des conditions météo-routières.

RÉFÉRENCES

- ADULYASAK, Y. et JAILLET, P. (2016). Models and algorithms for stochastic and robust vehicle routing with deadlines. *Transportation Science*, 50(2):608–626.
- AGRA, A., CHRISTIANSEN, M., FIGUEIREDO, R., HVATTUM, L. M., POSS, M. et REQUEJO, C. (2013). The robust vehicle routing problem with time windows. *Computers & Operations Research*, 40(3):856 – 866. Transport Scheduling.
- AGRA, A., CHRISTIANSEN, M., FIGUEIREDO, R., MAGNUS HVATTUM, L., POSS, M. et REQUEJO, C. (2012). Layered formulation for the robust vehicle routing problem with time windows. In MAHJOUR, A. R., MARKAKIS, V., MILIS, I. et PASCHOS, V. T., éditeurs : *Combinatorial Optimization*, pages 249–260, Berlin, Heidelberg. Springer Berlin Heidelberg.
- AHABCHANE, C., LANGEVIN, A. et TRÉPANIÉ, M. (2020). The mixed capacitated general routing problem with time-dependent demands. Working paper, Department of Mathematics and Industrial Engineering and CIRRELT, École Polytechnique de Montréal.
- AHABCHANE, C., LANGEVIN, A. et TRÉPANIÉ, M. (2015). Développement d’indicateurs de performance sur les opérations d’épandage de fondants et d’abrasifs sur réseau routier à partir de données de télémétrie. *CIGI 2015. Congrès international de génie industriel*, http://www.simagi.polymtl.ca/congresgi/cigi2015/Articles/CIGI_2015_submission_53.pdf.
- AHABCHANE, C., TRÉPANIÉ, M. et LANGEVIN, A. (2019). Street-segment-based salt and abrasive prediction for winter maintenance using machine learning and GIS. *Transactions in GIS*, 23(1):48–69.
- AKIN, M. et SHI, X. (2010). Development of standardized test procedures for evaluating deicing chemicals. *Rapport technique*.
- AMINU, U. et EGGLESE, R. (2006). A constraint programming approach to the Chinese postman problem with time windows. *Computers & Operations Research*, 33(12):3423 – 3431. Part Special Issue: Recent Algorithmic Advances for Arc Routing Problems.

- BACH, L., HASLE, G. et WØHLK, S. (2013). [A lower bound for the Node, Edge, and Arc Routing Problem](#). *Computers & Operations Research*, 40(4):943 – 952.
- BACH, L., LYSGAARD, J. et WØHLK, S. (2016). A branch-and-cut-and-price algorithm for the mixed capacitated general routing problem. *Networks*, 68(3):161–184.
- BALSEIRO, S., LOISEAU, I. et RAMONET, J. (2011). [An Ant Colony algorithm hybridized with insertion heuristics for the Time Dependent Vehicle Routing Problem with Time Windows](#). *Computers & Operations Research*, 38(6):954 – 966.
- BEDERINA, H. et HIFI, M. (2018). A hybrid multi-objective evolutionary optimization approach for the robust vehicle routing problem. *Applied Soft Computing*, 71:980 – 993.
- BEN-TAL, A. et NEMIROVSKI, A. (1998). Robust convex optimization. *Mathematics of Operations Research*, 23(4):769–805.
- BENGIO, Y., COURVILLE, A. et VINCENT, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828.
- BENSON, D., BANDER, J. et WHITE, C. (1998). A planning and operational decision support system for winter storm maintenance in an its environment. *IEEE International Conference on Intelligent Vehicles* (pp. 673-675).
- BERGSTRA, J. et BENGIO, Y. (2012). Random Search for Hyper-Parameter Optimization. *JOURNAL OF MACHINE LEARNING RESEARCH*, 13:281–305.
- BERTSIMAS, D. et SIM, M. (2003). Robust discrete optimization and network flows. *Mathematical Programming*, 98:49–71.
- BEULLENS, P., MUYLDERMANS, L., CATTRYSSSE, D. et OUDHEUSDEN, D. V. (2003). [A guided local search heuristic for the capacitated arc routing problem](#). *European Journal of Operational Research*, 147(3):629 – 643.
- BLACK, D., EGGLESE, R. et WØHLK, S. (2013). [The time-dependent prize-collecting arc routing problem](#). *Computers & Operations Research*, 40(2):526 – 535.

- BLACKBURN, R. R., AMSLER, D. et BAUER, K. M. (2004). Guidelines for snow and ice control materials and methods. *Transportation Research Circular*, pages 31–49.
- BLOMQVIST, G., GUSTAFSSON, M., ERAM, M. et ÜNVER, K. (2011). Prediction of salt on road surface: Tool to minimize use of salt. *Transportation Research Record*, 2258(1):131–138.
- BOSCO, A., LAGANÀ, D., MUSMANNO, R. et VOCATURO, F. (2013). Modeling and solving the mixed capacitated general routing problem. *Optimization Letters*, 7(7):1451–1469.
- BOSCO, A., LAGANÀ, D., MUSMANNO, R. et VOCATURO, F. (2014). A matheuristic algorithm for the mixed capacitated general routing problem. *Networks*, 64(4):262–281.
- BRAATEN, S., GJØNNES, O., HVATTUM, L. M. et TIRADO, G. (2017). Heuristics for the robust vehicle routing problem with time windows. *Expert Systems with Applications*, 77:136 – 147.
- BREIMAN, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.
- BREIMAN, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.
- CALOGIURI, T., GHIANI, G., GUERRIERO, E. et MANSINI, R. (2019). A branch-and-bound algorithm for the time-dependent rural postman problem. *Computers & Operations Research*, 102:150 – 157.
- CAMPBELL, J., LANGEVIN, A. et PERRIER, N. (2014a). *Chapter 14: Advances in Vehicle Routing for Snow Plowing*, pages 321–350. SIAM.
- CAMPBELL, J. F. et LANGEVIN, A. (1995a). Operations management for urban snow removal and disposal. *Transportation Research Part A: Policy and Practice*, 29(5):359 – 370.
- CAMPBELL, J. F. et LANGEVIN, A. (1995b). Operations management for urban snow removal and disposal. *Transportation Research Part A: Policy and Practice*, 29(5):359 – 370.
- CAMPBELL, J. F. et LANGEVIN, A. (1995c). The snow disposal assignment problem. *Journal of the Operational Research Society*, 46(8):919–929.

- CAMPBELL, J. F. et LANGEVIN, A. (2000). *Roadway Snow and Ice Control*, pages 389–418. Springer US, Boston, MA.
- CAMPBELL, J. F., LANGEVIN, A. et PERRIER, N. (2014b). *Chapter 14: Advances in Vehicle Routing for Snow Plowing*, chapitre 14, pages 321–350. Society for Industrial and Applied Mathematics.
- CAO, E., LAI, M. et YANG, H. (2014). Open vehicle routing problem with demand uncertainty and its robust strategies. *Expert Systems with Applications*, 41(7):3569 – 3575.
- CHEN, L., GENDREAU, M., HÀ, M. H. et LANGEVIN, A. (2016). [A robust optimization approach for the road network daily maintenance routing problem with uncertain service time](#). *Transportation Research Part E: Logistics and Transportation Review*, 85:40 – 51.
- CHEN, T. et GUESTRIN, C. (2016). Xgboost: A scalable tree boosting system. *In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794, New York, NY, USA. Association for Computing Machinery.
- CHRISTIAENS, J. et VANDEN BERGHE, G. (2020). Slack induction by string removals for vehicle routing problems. *Transportation Science*, 54(2):417–433.
- CIANCIO, C., LAGANÁ, D. et VOCATURO, F. (2018). Branch-price-and-cut for the mixed capacitated general routing problem with time windows. *European Journal of Operational Research*, 267(1):187 – 199.
- COOK, T. M. et ALPRIN, B. S. (1976). Snow and ice removal in an urban environment. *Management Science*, 23(3):227–234.
- CORBERAN, A. et LAPORTE, G. (2014). *Arc Routing*. Society for Industrial and Applied Mathematics, Philadelphia, PA.
- CÉSAR, J., GUTIÉRREZ, A. et HERVÁS, A. (2002). The capacitated general routing problem on mixed graphs. *Revista Investigación Operacional*, 23.

- DABIA, S., ROPKE, S., van WOENSEL, T. et DE KOK, T. (2013). [Branch and Price for the Time-Dependent Vehicle Routing Problem with Time Windows](#). *Transportation Science*, 47(3):380–396.
- DANTZIG, G. B. et RAMSER, J. H. (1959). The truck dispatching problem. *Management Science*, 6(1):80–91.
- DE LA VEGA, J., MUNARI, P. et MORABITO, R. (2019). Robust optimization for the vehicle routing problem with multiple deliverymen. *Central European Journal of Operations Research*, 27(4):905–936.
- DELL’AMICO, M., DÍAZ, J. C. D., HASLE, G. et IORI, M. (2016). An adaptive iterated local search for the mixed capacitated general routing problem. *Transportation Science*, 50(4):1223–1238.
- DOMINGOS, P. (2012). A few useful things to know about machine learning. *Commun. ACM*, 55(10):78–87.
- DONATI, A. V., MONTEMANNI, R., CASAGRANDE, N., RIZZOLI, A. E. et GAMBARDELLA, L. M. (2008). [Time dependent vehicle routing problem with a multi ant colony system](#). *European Journal of Operational Research*, 185(3):1174 – 1191.
- DRUSCHEL, S. J. (2012). Salt brine blending to optimize deicing and anti-icing performance. *Rapport technique*.
- DUNNING, I., HUCHETTE, J. et LUBIN, M. (2017). Jump: A modeling language for mathematical optimization. *SIAM Review*, 59(2):295–320.
- EGLESE, R., GOLDEN, B. et WASIL, E. (2014). *Chapter 13: Route Optimization for Meter Reading and Salt Spreading*, chapitre 13, pages 303–320. Society for Industrial and Applied Mathematics.
- EL GHAOUI, L., OUSTRY, F. et LEBRET, H. (1998). Robust solutions to uncertain semidefinite programs. *SIAM Journal on Optimization*, 9(1):33–52.

- EVANS, J. (1990). Design and implementation of a vehicle routing and planning system for snow and ice control. *Rapport technique, Université de Cincinnati, Ohio*.
- EVANS, J. et WEANT, M. (1990). Strategic planning for snow and ice control using computer-based routing software. *Rapport technique*, pages 121, 60–64.
- FENG, F. et FU, L. (2015). Winter road surface condition forecasting. *Journal of Infrastructure Systems*, 21(3):04014049.
- FREUND, Y. et SCHAPIRE, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1): 119 – 139.
- FRIEDMAN, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Ann. Statist.*, 29(5):1189–1232.
- FU, L., TRUDEL, M. et KIM, V. (2009). Optimizing winter road maintenance operations under real-time information. *European Journal of Operational Research*, 196(1):332 – 341.
- FUKASAWA, R., LONGO, H., LYSGAARD, J., ARAGÃO, M. P. d., REIS, M., UCHOA, E. et WERNECK, R. F. (2006). Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Mathematical Programming*, 106(3):491–511.
- GAZE, K. A. (2013). Exact optimization methods for the mixed capacitated general routing problem. *Mémoire de maitrise, Université de science et technologie Norvège, 2007*.
- GENDREAU, M., GHIANI, G. et GUERRIERO, E. (2015). [Time-dependent routing problems: A review](#). *Computers & Operations Research*, 64:189 – 197.
- GEURTS, P., ERNST, D. et WEHENKEL, L. (2006). Extremely randomized trees. *Machine Learning*, 63(1):3–42.
- GOLDEN, B. L. et WONG, R. T. (1981). Capacitated arc routing problems. *Networks*, 11(3):305–315.

- GOUNARIS, C. E., REPOUSSIS, P. P., TARANTILIS, C. D., WIESEMANN, W. et FLOUDAS, C. A. (2016). An adaptive memory programming framework for the robust capacitated vehicle routing problem. *Transportation Science*, 50(4):1239–1260.
- GOUNARIS, C. E., WIESEMANN, W. et FLOUDAS, C. A. (2013). The robust capacitated vehicle routing problem under demand uncertainty. *Operations Research*, 61(3):677–693.
- GUYON, I. et ELISSEEFF, A. (2006). *An Introduction to Feature Extraction*, pages 1–25. Springer Berlin Heidelberg, Berlin, Heidelberg.
- HAGHANI, A. et QIAO, H. (2002). Snow emergency vehicle routing with route continuity constraints. *Transportation Research Record: Journal of the Transportation Research Board*, 1783:119–124.
- HAJIBABAI, L., NOURBAKHS, S. M., OUYANG, Y. et PENG, F. (2014). Network routing of snowplow trucks with resource replenishment and plowing priorities: Formulation, algorithm, and application. *Transportation Research Record*, 2440(1):16–25.
- HAJIBABAI, L. et OUYANG, Y. (2016). Dynamic snow plow fleet management under uncertain demand and service disruption. *IEEE Transactions on Intelligent Transportation Systems*, 17(9):2574–2582.
- HAMBY, D. M. (1994). A review of techniques for parameter sensitivity analysis of environmental models. *Environmental Monitoring and Assessment*, 32(2):135–154.
- HANDA, H., CHAPMAN, L. et YAO, X. (2005). Dynamic salting route optimisation using evolutionary computation. volume 1, pages 158 – 165 Vol.1.
- HASLE, G., KLOSTER, O., SMEDSRUD, M. et GAZE, K. A. (2012). Experiments on the node, edge, and arc routing problem. *Rapport technique*.
- HOSSAIN, S., FU, L. et LU, C.-Y. (2014). Deicing performance of road salt. *Transportation Research Record: Journal of the Transportation Research Board*, 2440:76–84.

- HOSSAIN, S. K., FU, L., HOSSEINI, F., MURESAN, M., DONNELLY, T. et KABIR, S. (2016). Optimum winter road maintenance: effect of pavement types on snow melting performance of road salts. *Canadian Journal of Civil Engineering*, 43(9):802–811.
- HOSSAIN, S. M. K., FU, L., LI, S. D., DONNELLY, T. et MURESAN, M. (2015). *Modeling the Snow Melting Performance of Salt: A Mechanistic-Empirical Approach*, pages 464–475.
- HU, C., LU, J., LIU, X. et ZHANG, G. (2018). Robust vehicle routing problem with hard time windows under demand and travel time uncertainty. *Computers & Operations Research*, 94:139 – 153.
- ICHOUA, S., GENDREAU, M. et POTVIN, J.-Y. (2003). [Vehicle dispatching with time-dependent travel times](#). *European Journal of Operational Research*, 144(2):379 – 396.
- IRNICH, S., LAGANÀ, D., SCHLEBUSCH, C. et VOCATURO, F. (2015). Two-phase branch-and-cut for the mixed capacitated general routing problem. *European Journal of Operational Research*, 243(1):17 – 29.
- JAILLET, P., QI, J. et SIM, M. (2016). Routing optimization under uncertainty. *Operations Research*, 64(1):186–200.
- JIE, L., van ZUYLEN, H., CHUNHUA, L. et SHOUFENG, L. (2011). Monitoring travel times in an urban network using video, gps and bluetooth. *Procedia - Social and Behavioral Sciences*, 20:630 – 637. The State of the Art in the European Quantitative Oriented Transportation and Logistics Research – 14th Euro Working Group on Transportation & 26th Mini Euro Conference & 1st European Scientific Conference on Air Transport.
- JUNG, S. et HAGHANI, A. (2001). [Genetic Algorithm for the Time-Dependent Vehicle Routing Problem](#). *Transportation Research Record*, 1771(1):164–171.
- KAYACI ÇODUR, M. et YILMAZ, M. (2018). [A time-dependent hierarchical Chinese postman problem](#). *Central European Journal of Operations Research*.
- KINABLE, J., van HOEVE, W.-J. et SMITH, S. (2016). Optimization models for a real-world snow plow routing problem. pages 229–245.

- KING, D. M. et PERERA, B. J. C. (2011). Sensitivity analysis of yield estimate of urban water supply systems. *Australasian Journal of Water Resources*, 14(2):141–155.
- KOKUBUGATA, H., MORIYAMA, A. et KAWASHIMA, H. (2007). A practical solution using simulated annealing for general routing problems with nodes, edges, and arcs. In STÜTZLE, T., BIRATTARI, M. et H. HOOS, H., éditeurs : *Engineering Stochastic Local Search Algorithms. Designing, Implementing and Analyzing Effective Heuristics*, pages 136–149, Berlin, Heidelberg. Springer Berlin Heidelberg.
- KORTEWEG, P. et VOLGENANT, T. (2006). On the hierarchical chinese postman problem with linear ordered classes. *European Journal of Operational Research*, 169(1):41 – 52.
- LAPORTE, G., RØPKE, S. et VIDAL, T. (2014). *Heuristics for the Vehicle Routing Problem*, chapitre 4, pages 87–116. SIAM - Society for Industrial and Applied Mathematics, 2 édition.
- LEE, C., LEE, K. et PARK, S. (2012). Robust vehicle routing problem with deadlines and travel time/demand uncertainty. *Journal of the Operational Research Society*, 63(9):1294–1306.
- LETCHFORD, A. N. et OUKIL, A. (2009). Exploiting sparsity in pricing routines for the capacitated arc routing problem. *Computers & Operations Research*, 36(7):2320 – 2327.
- LI, L. Y. O. et EGGLESE, R. W. (1996). An interactive algorithm for vehicle routing for winter - gritting. *The Journal of the Operational Research Society*, 47(2):217–228.
- LU, D. et GZARA, F. (2019). The robust vehicle routing problem with time windows: Solution by branch and price and cut. *European Journal of Operational Research*, 275(3):925 – 938.
- LUKMAN, R. K., CERINŠEK, M., VRTIČ, P. et HORVAT, B. (2018). Improving efficient resource usage and reducing carbon dioxide emissions by optimizing fleet management for winter services. *Journal of Cleaner Production*, 177:1 – 11.
- LYSTLUND, L. et WØHLK, S. (2011). *The Service-Time Restricted Capacitated Arc Routing Problem*. Workingpaper, Aarhus School of Business, Aarhus University, CORAL - Centre for Operations Research Applications in Logistics.

- MALANDRAKI, C. et DASKIN, M. S. (1992). [Time Dependent Vehicle Routing Problems: Formulations, Properties and Heuristic Algorithms](#). *Transportation Science*, 26(3):185–200.
- MANDAL, S. K., PACCIARELLI, D., LØKKETANGEN, A. et HASLE, G. (2015). A memetic nsga-ii for the bi-objective mixed capacitated general routing problem. *Journal of Heuristics*, 21(3):359–390.
- MEINSHAUSEN, N. et BÜHLMANN, P. (2010). Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4):417–473.
- MOGHADDAM, B. F., RUIZ, R. et SADJADI, S. J. (2012). Vehicle routing problem with uncertain demands: An advanced particle swarm algorithm. *Computers & Industrial Engineering*, 62(1):306 – 317.
- MONROY-LICHT, M., AMAYA, C. A. et LANGEVIN, A. (2014). [The Rural Postman Problem with time windows](#). *Networks*, 64(3):169–180.
- MOURÃO, M. C. a. et PINTO, L. S. (2017). [An updated annotated bibliography on arc routing problems](#). *Networks*, 70(3):144–194.
- MTQ (2020). Guide de gestion des zones vulnÉrables aux sels de voirie , accédé le 28 janvier 2013. https://www.transports.gouv.qc.ca/fr/gestion-environnementale-sels-voirie/Documents/GSV/references-utiles/publications_MTQ/Guide_gestion_zones_vulnerables-2Mai_HQ.pdf.
- MULLASERIL, P. A. (1997). [Capacitated rural postman problem with time windows and split delivery](#). Thèse de doctorat, University of Arizona.
- MUNARI, P., MORENO, A., DE LA VEGA, J., ALEM, D., GONDZIO, J. et MORABITO, R. (2019). The robust vehicle routing problem with time windows: Compact formulation and branch-price-and-cut method. *Transportation Science*, 53(4):1043–1066.
- NATEKIN, A. et KNOLL, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics*, 7:21.

- NIXON, W. (2009). National advances in winter maintenance. *Paper presented at the 8th Annual Road Salt Symposium, Minnesota.*
- NOORIZADEGAN, M., GALLI, L. et CHEN, B. (2012). On the heterogeneous vehicle routing problem under demand uncertainty.
- NORMANDIN, P.-A. (2015). Déneiger coûte plus d'un milliard par an au québec , accédé le 28 janvier 2013. <http://www.lapresse.ca/actualites/national/2014/12/10/01-4827154-deneiger-coute-plus-dun-milliard-par-an-au-quebec.php>.
- NOSSACK, J., GOLDEN, B., PESCH, E. et ZHANG, R. (2017). [The windy rural postman problem with a time-dependent zigzag option](#). *European Journal of Operational Research*, 258(3):1131 – 1142.
- OMER, M. (2007). Efficient routing of snow removal vehicles. *Mémoire de maitrise, Université de West Virginia, College of Engineering and Mineral Resources, 2007.*
- ORDONEZ, F. (2010). *Robust Vehicle Routing*, chapitre Chapter 7, pages 153–178. Informs.
- PAN, GUANGYUAN, FU, LIPING, YU, RUIFAN, MURESAN et MATTHEW (2018). Winter road surface condition recognition using a pretrained deep convolutional network. *Transportation Research Board 97th Annual Meeting*.
- PANDI, R. et MURALIDHARAN, B. (1995). A capacitated general routing problem on mixed networks. *Computers & Operations Research*, 22(5):465 – 478.
- PERRIER, N., CAMPBELL, J., GENDREAU, M. et LANGEVIN, A. (2012). Vehicle routing models and algorithms for winter road spreading operations. *Hybrid Algorithms for Service, Computing and Manufacturing Systems: Routing and Scheduling Solutions*, pages 15–45.
- PERRIER, N., LANGEVIN, A. et AMAYA, C.-A. (2008). Vehicle routing for urban snow plowing operations. *Transportation Science*, 42(1):44–56.
- PERRIER, N., LANGEVIN, A. et CAMPBELL, J. (2006a). A survey of models and algorithms for winter road maintenance. part i: System design for spreading and plowing. *Computers & Operations Research*, 33:209–238.

- PERRIER, N., LANGEVIN, A. et CAMPBELL, J. (2006b). A survey of models and algorithms for winter road maintenance. part ii: System design for snow disposal. *Computers & Operations Research*, 33:239–262.
- PERRIER, N., LANGEVIN, A. et CAMPBELL, J. (2007a). A survey of models and algorithms for winter road maintenance. part iii: Vehicle routing and depot location for spreading. *Computers & Operations Research*, 34:211–257.
- PERRIER, N., LANGEVIN, A. et CAMPBELL, J. (2007b). A survey of models and algorithms for winter road maintenance. part iv: Vehicle routing and fleet sizing for plowing and snow disposal. *Computers & Operations Research*, 33:258–294.
- PERRIER, N., LANGEVIN, A. et CAMPBELL, J. F. (2007c). [A survey of models and algorithms for winter road maintenance. Part III: Vehicle routing and depot location for spreading.](#) *Computers & Operations Research*, 34(1):211 – 257.
- PRINS, C. et BOUCHENOUA, S. (2005a). [A Memetic Algorithm Solving the VRP, the CARP and General Routing Problems with Nodes, Edges and Arcs](#), pages 65–85. Springer Berlin Heidelberg, Berlin, Heidelberg.
- PRINS, C. et BOUCHENOUA, S. (2005b). [A Memetic Algorithm Solving the VRP, the CARP and General Routing Problems with Nodes, Edges and Arcs](#), pages 65–85. Springer Berlin Heidelberg, Berlin, Heidelberg.
- QIAO, H. (2002). Capacitated arc routing problem: formulations, algorithms and application. *Thèse de doctorat, Université de Maryland, College Park, 2015.*
- QUIRION-BLAIS, O., LANGEVIN, A., LEHUÉDÉ, F., PÉTON, O. et TRÉPANIER, M. (2017). [Solving the large-scale min-max K-rural postman problem for snow plowing.](#) *Networks*, 70(3):195–215.
- QUIRION-BLAIS, O., TRÉPANIER, M. et LANGEVIN, A. (2015). A case study of snow plow routing using an adaptive large hood search metaheuristic. *Transportation Letters*, 7(4):201–209.

- REFAEILZADEH, P., TANG, L. et LIU, H. (2009). Cross-validation. *Encyclopedia of database systems*, 5.
- ROPKE, S. et PISINGER, D. (2006). [An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows](#). *Transportation Science*, 40(4):455–472.
- ROSENBLATT, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408.
- ROUSSEAU, L.-M., GENDREAU, M. et GILLES, P. (2002). Using constraint-based operators to solve the vehicle routing problem with time windows. *Journal of Heuristics*, 8:43–58.
- ROUSSEAU, T. (2019). Politique de déneigement, accédé le 08 février 2020. https://villededonnacona.com/upload/faq/politique_de_deneigement.pdf.
- SHI, X. et AKIN, M. (2012). Holistic approach to decision making in the formulation and selection of anti-icing products. *Journal of Cold Regions Engineering*, 26(3):101–117.
- SHI, X., FORTUNE, K., SMITHLIN, R., AKIN, M. et FAY, L. (2013). Exploring the performance and corrosivity of chloride deicer solutions: Laboratory investigation and quantitative modeling. *Cold Regions Science and Technology*, 86:36 – 44.
- SOLANO-CHARRIS, E., PRINS, C. et SANTOS, A. C. (2015). Local search based metaheuristics for the robust vehicle routing problem with discrete scenarios. *Applied Soft Computing*, 32:518 – 531.
- SOLOMON, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35(2):254–265.
- SOYSTER, A. (1974). Examination of alternative models for routing winter maintenance trucks. *Rapport technique No 7408 (pp. 11-24), Institut de Transport Pennsylvania*.
- SUBRAMANYAM, A., REPOUSSIS, P. P. et GOUNARIS, C. E. (2020). Robust optimization of a broad class of heterogeneous vehicle routing problems under demand uncertainty. *INFORMS Journal on Computing*, 0(0):null.

- SULLIVAN, J. L., DOWDS, J., NOVAK, D. C., SCOTT, D. M. et RAGSDALE, C. (2019). Development and application of an iterative heuristic for roadway snow and ice control. *Transportation Research Part A: Policy and Practice*, 127:18 – 31.
- SUN, J., MENG, Y. et TAN, G. (2015). [An integer programming approach for the Chinese postman problem with time-dependent travel time](#). *Journal of Combinatorial Optimization*, 29(3):565–588.
- SUN, J., TAN, G. et HOU, G. (2011). [A New Integer Programming Formulation for the Chinese Postman Problem with Time Dependent Travel Times](#). *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, 5(4):410 – 414.
- SUNGUR, I., ORDÓÑEZ, F. et DESSOUKY, M. (2008). A robust optimization approach for the capacitated vehicle routing problem with demand uncertainty. *IIE Transactions*, 40(5):509–523.
- SUNGUR, I., REN, Y., ORDÓÑEZ, F., DESSOUKY, M. et ZHONG, H. (2010). A model and algorithm for the courier delivery problem with uncertainty. *Transportation Science*, 44(2):193–205.
- TAGMOUTI, M., GENDREAU, M. et POTVIN, J.-Y. (2007). [Arc routing problems with time-dependent service costs](#). *European Journal of Operational Research*, 181(1):30 – 39.
- TAGMOUTI, M., GENDREAU, M. et POTVIN, J.-Y. (2010). [A variable neighborhood descent heuristic for arc routing problems with time-dependent service costs](#). *Computers & Industrial Engineering*, 59(4):954 – 963.
- TAGMOUTI, M., GENDREAU, M. et POTVIN, J.-Y. (2011). [A dynamic capacitated arc routing problem with time-dependent service costs](#). *Transportation Research Part C: Emerging Technologies*, 19(1):20 – 28.
- TAN, G. et SUN, J. (2011). [An Integer Programming Approach for the Rural Postman Problem with Time Dependent Travel Times](#). In FU, B. et DU, D.-Z., éditeurs : *Computing and Combinatorics*, pages 414–431, Berlin, Heidelberg. Springer Berlin Heidelberg.

- TIRKOLAEI, E. B., MAHDAVI, I. et ESFAHANI, M. M. S. (2018). A robust periodic capacitated arc routing problem for urban waste collection considering drivers and crew's working time. *Waste Management*, 76:138 – 146.
- TOKLU, N. E., MONTEMANNI, R. et GAMBARDELLA, L. M. (2013a). An ant colony system for the capacitated vehicle routing problem with uncertain travel costs. *In 2013 IEEE Symposium on Swarm Intelligence (SIS)*, pages 32–39.
- TOKLU, N. E., MONTEMANNI, R. et GAMBARDELLA, L. M. (2013b). A robust multiple ant colony system for the capacitated vehicle routing problem. *In 2013 IEEE International Conference on Systems, Man, and Cybernetics*, pages 1871–1876.
- TONG, D., MERRY, C. J. et COIFMAN, B. (2009). New perspectives on the use of gps and gis to support a highway performance study. *Transactions in GIS*, 13(1):69–85.
- TOOBAIE, S. et HAGHANI, A. (2004). Minimal arc partitioning problem: Formulation and application in snow routing with service route continuity. *Transportation Research Record*, 1882(1):167–175.
- TOTH, P. et VIGO, D. (2003). [The Granular Tabu Search and Its Application to the Vehicle-Routing Problem](#). *INFORMS Journal on Computing*, 15(4):333–346.
- TOTH, P. et VIGO, D. (2014). *Vehicle Routing*. Society for Industrial and Applied Mathematics, Philadelphia, PA.
- TRENOUTH, W. R., GHARABAGHI, B. et PERERA, N. (2015). Road salt application planning tool for winter de-icing operations. *Journal of Hydrology*, 524:401 – 410.
- UNGERER, R. (1989). Decision models for winter highway maintenance. *Rapport technique No FHWA/NY/SR-89/95, Département des transports de l'État de New York*.
- VANSTEENWEGEN, P., SOUFFRIAU, W. et SÖRENSEN, K. (2010). [Solving the mobile mapping van problem: A hybrid metaheuristic for capacitated arc routing with soft time windows](#). *Computers & Operations Research*, 37(11):1870 – 1876. Metaheuristics for Logistics and Vehicle Routing.

VICTORIA, J. F., AFSAR, H. M. et PRINS, C. (2016). [Column Generation based heuristic for the Vehicle Routing Problem with Time-Dependent Demand](#). *IFAC-PapersOnLine*, 49(12):526 – 531. 8th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2016.

VIDAL, T. (2017). [Node, Edge, Arc Routing and Turn Penalties: Multiple Problems—One Neighborhood Extension](#). *Operations Research*, 65(4):992–1010.

VILLE DE CALGARY (2019). Snow and ice control faq, accédé le 28 janvier 2020. <https://www.calgary.ca/Transportation/Roads/Pages/Road-Maintenance/Snow-and-ice-control/SNIC-policy-FAQs.aspx?master=old>.

VILLE DE LAVAL (2018). Stratégie neige, accédé le 08 février 2020. <https://www.laval.ca/Documents/Pages/Fr/Citoyens/transport-deneigement-et-travaux/strategie-neige.pdf>.

VILLE DE LONGUEUIL (2020). Les opérations de déneigement : étapes et priorités, accédé le 08 février 2020. <https://www.longueuil.quebec/fr/neige/operations>.

VILLE DE MONT-ROYAL (2020). Politique de déneigement, accédé le 08 février 2020. <https://www.ville.mont-royal.qc.ca/sites/default/files/documents/politique-de-deneigement.pdf>.

VILLE DE MONTRÉAL (2016). Budget de fonctionnement 2016, accédé le 28 janvier 2020. <https://www.ledevoir.com/documents/pdf/Budget2016.pdf>.

VILLE DE MONTRÉAL (2019). Budget de fonctionnement 2019, accédé le 28 janvier 2020. http://ville.montreal.qc.ca/pls/portal/docs/page/service_fin_fr/media/documents/2019_budget_fonctionnement.pdf.

VILLE DE MONTRÉAL (2020a). Budget 2020 pti 2020-2022, accédé le 28 janvier 2020. https://ville.montreal.qc.ca/pls/portal/docs/page/service_fin_fr/media/documents/budget_2020_pti_2020_2022_fr.pdf.

VILLE DE MONTRÉAL (2020b). Déneigement municipal, accédé le 08 février 2020. <http://www1.ville.montreal.qc.ca/banque311/content/d%C3%A9neigement-municipal>.

VILLE DE QUEBEC (2009). Politique de déneigement, accédé le 28 janvier 2020. https://www.ville.quebec.qc.ca/publications/docs_ville/politique_deneigement.pdf.

VILLE DE QUEBEC (2020). Déneigement, accédé le 28 janvier 2020. <https://www.ville.quebec.qc.ca/citoyens/deneigement/>.

VILLE D'EDMONTON (2019). Operating budget 2019-2022, accédé le 28 janvier 2020. https://www.edmonton.ca/city_government/documents/APPROVED_2019-2022_OPERATING_BUDGET.pdf.

WANG, H.-F. et WEN, Y.-P. (2002). Time-constrained Chinese postman problems. *Computers & Mathematics with Applications*, 44(3):375 – 387.

WANG, S., NAN, B., ROSSET, S. et ZHU, J. (2011). Random lasso. *Ann. Appl. Stat.*, 5(1):468–485.

WEI, P., LU, Z. et SONG, J. (2015). Variable importance analysis: A comprehensive review. *Reliability Engineering & System Safety*, 142:399 – 432.

WØHLK, S. (2005). *Contributions to arc routing*. Thèse de doctorat, University of southern Denmark.

WØHLK, S. (2006). New lower bound for the capacitated arc routing problem. *Computers & Operations Research*, 33(12):3458 – 3472. Part Special Issue: Recent Algorithmic Advances for Arc Routing Problems.

YE, Z., SHI, X., STRONG, C. K. et GREENFIELD, T. H. (2009). Evaluation of effects of weather information on winter maintenance costs. *Transportation Research Record*, 2107(1): 104–110.

YE, Z., XU, Y., VENEZIANO, D. et SHI, X. (2014). Evaluation of winter maintenance chemicals and crashes with an artificial neural network. *Transportation Research Record*, 2440(1):43–50.

YUN, YANG, D. G. K. S., KIM, C. H., GUKPARK, J. et HONG, J. (2018). Monitoring the road surface temperature by evaluating the climatological and ambient temperature data collected from probe vehicles using machine learning technique. *Transportation Research Board 97th Annual Meeting*, pages 18–02505.

İKİZ, N. et GALİP, E. (2016). Computerized decision tree for anti-icing/pretreatment applications as a result of laboratory and field testings. *Cold Regions Science and Technology*, 126:90 – 108.