



HAL
open science

Méta-modélisation, simulation et optimisation de flux urbains

Florian Leprêtre

► **To cite this version:**

Florian Leprêtre. Méta-modélisation, simulation et optimisation de flux urbains. Modélisation et simulation. Université du Littoral Côte d'Opale, 2020. Français. NNT : 2020DUNK0571 . tel-03178850

HAL Id: tel-03178850

<https://theses.hal.science/tel-03178850v1>

Submitted on 24 Mar 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DU LITTORAL CÔTE D'OPALE

École doctorale **Sciences Pour l'Ingénieur** Université Lille Nord-de-France

Laboratoire d'Informatique Signal et Image de la Côte d'Opale

Florian LEPRÊTRE

Soutenue le **20 novembre 2020**

En vue de l'obtention du grade de docteur de l'Université du Littoral Côte d'Opale

Discipline **Sciences et Technologies de l'Information et de la
Communication : Informatique**

Méta-Modélisation, Simulation et Optimisation de Flux Urbains

Thèse supervisée par Cyril FONLUPT Directeur
Sébastien VEREL Co-directeur
Virginie MARION Co-encadrante

Composition du jury

<i>Rapporteurs</i>	Adrien GOËFFON	LERIA	
	Lhassane IDOUMGHAR	UHA	Président du jury
<i>Examineur</i>	Abdeljalil ABBAS-TURKY	UTBM	
<i>Invitée</i>	Marie CAPON	Service Aménagement de la Mairie de Calais	
<i>Superviseurs</i>	Cyril FONLUPT	LISIC	
	Sébastien VEREL	LISIC	
	Virginie MARION	LISIC	

MÉTA-MODÉLISATION, SIMULATION ET OPTIMISATION DE FLUX URBAINS**Résumé**

La conception et la planification efficace des infrastructures urbaines représentent un enjeu essentiel pour les experts urbanistes qui souhaitent atteindre une mobilité fiable et durable pour les citoyens. Les villes en perpétuelle évolution et les habitudes changeantes des populations incitent les urbanistes à réinventer constamment leur utilisation de l'espace urbain afin de contribuer à la fluidité des déplacements et à la sécurité des voyageurs, tout en limitant les impacts environnementaux des flux de trafic. De nombreuses méthodes d'optimisation permettent de résoudre une variété de problèmes associés à des questions de mobilité urbaine, tels que l'optimisation du réglage des feux de signalisation, la conception de réseaux routiers fiables ou la planification optimale d'un système de transports en commun. Toutefois, ces méthodes sont généralement conçues et paramétrées pour l'étude d'espaces urbains très spécifiques, et semblent difficilement généralisables à d'autres villes du monde pour l'optimisation de leur mobilité. Aussi, ce mémoire introduit une méthodologie pour la modélisation, la simulation et l'optimisation des flux urbains. Les travaux rapportés proposent une analyse rigoureuse des caractéristiques intrinsèques aux problèmes d'optimisation de la mobilité urbaine, afin de parvenir à une meilleure compréhension de ces défis, et de permettre la conception d'algorithmes d'optimisation robustes et efficaces. Finalement, les travaux rapportés par ce mémoire visent à susciter la réflexion sur des problèmes de mobilité, en s'appuyant sur les points de vue des experts urbanistes et des experts en optimisation, afin de parvenir à une mobilité urbaine optimale et durable.

Mots clés : méta-modèle, simulation, optimisation, mobilité urbaine

Abstract

The efficient design and planning of urban infrastructure is a key issue for urban planning experts who want to achieve reliable and sustainable mobility for city dwellers. Ever-changing cities and changing population habits prompt urban planners to constantly reinvent their use of urban space in order to contribute to the fluidity of travel and passenger safety, while limiting the environmental impacts of traffic flows. Numerous optimization methods can be used to solve a variety of problems associated with urban mobility issues, such as optimizing traffic light settings, designing reliable road networks or optimizing the planning of a public transit system. However, these methods are generally designed and parameterized for the study of very specific urban spaces, and seem difficult to generalize to other cities in the world for mobility optimization. Therefore, this paper introduces a methodology for modeling, simulation and optimization of urban flows. The reported work proposes a rigorous analysis of the intrinsic characteristics of urban mobility optimization problems, in order to achieve a better understanding of these challenges, and to enable the design of robust and efficient optimization algorithms. Finally, the work reported in this thesis aims to stimulate reflection on mobility problems, based on the views of urban planners and optimization experts, in order to achieve optimal and sustainable urban mobility.

Keywords: surrogate model, simulation, optimization, urban mobility

Table des matières

Résumé	ii
Table des matières	iii
Introduction	1
I Problèmes d’optimisation et mobilité urbaine	4
1 Problèmes d’optimisation	5
1.1 Typologies d’optimisation	6
1.1.1 Espace de recherche	7
1.1.2 Réglage des paramètres	8
1.1.3 Dérivabilité de la fonction objectif	9
1.1.4 Définition analytique du problème	10
1.2 Métaheuristiques	11
1.2.1 Opérateurs de variation	12
1.2.2 Algorithmes à solution unique	12
1.2.3 Algorithmes à population de solutions	14
1.2.4 Compromis exploration et exploitation	17
1.3 Paysages de fitness	19
1.3.1 Marche aléatoire	19
1.3.2 Métriques pour l’étude des paysages	20
1.4 Optimisation assistée par méta-modèle	22
1.4.1 Méthodologie	23
1.4.2 Méta-modèles pour l’optimisation combinatoire	23
1.5 Problèmes de référence	29
1.5.1 Problème <i>OneMax</i>	30
1.5.2 Problème du voyageur de commerce	30
1.5.3 <i>Unconstrained Binary Quadratic Programming</i>	31
1.5.4 Paysages NK	32

2 Problèmes d'optimisation en mobilité urbaine	34
2.1 Simulation des flux urbains	35
2.2 Optimisation de la mobilité urbaine	37
2.2.1 Réglage des feux de signalisation	37
2.2.2 Planification des transports en commun	38
2.2.3 Conception d'infrastructures et de véhicules intelligents	39
2.2.4 Développement des modes de mobilité douce	40
2.3 Connaissance experte et robustesse des méthodes d'optimisation	42
2.4 Optimisation du réglage des feux de circulation	43
2.4.1 Panorama des approches pour l'optimisation	43
2.4.2 Définition du problème d'optimisation	44
2.5 Optimisation du positionnement des arrêts de bus	47
II Contributions	50
3 SIALAC : Famille de scénarios de mobilité	51
3.1 Méthodologie	52
3.2 SIALAC Benchmark	53
3.2.1 Simulateur et état d'équilibre	53
3.2.2 Réseau routier et durée des trajets	54
3.2.3 Définition des scénarios de mobilité	55
3.2.4 Liste des scénarios	56
3.3 Valorisation du benchmark	58
4 Modèle d'importance	62
4.1 Modèle d'importance	63
4.2 Analyse expérimentale	64
4.2.1 Protocole expérimental	64
4.2.2 Temps de trajet des voyageurs	65
4.2.3 Rugosité des paysages de fitness	67
4.2.4 Importance des variables	70
5 Algorithme adaptatif	76
5.1 <i>Bandit Descent</i>	77
5.2 Analyse expérimentale	79
5.2.1 Protocole expérimental	80
5.2.2 Performances des algorithmes d'optimisation	81
5.3 Analyse des méta-paramètres de <i>Bandit Descent</i>	82
5.3.1 Stratégie de sélection et groupes de feux	84
5.3.2 Robustesse des solutions	86

5.4 Application	89
5.4.1 Application à la ville de Quito, Equateur	89
5.4.2 Convergence des algorithmes d'optimisation	92
5.4.3 Temps d'exécution des algorithmes d'optimisation	93
5.4.4 Validation de la méthodologie	95
6 Méta-modèles combinatoires	99
6.1 Optimisation assistée par méta-modèle de Walsh	100
6.2 Analyse expérimentale sur des problèmes artificiels	101
6.2.1 Protocole expérimental	103
6.2.2 Précision des méta-modèles	104
6.2.3 Performances en optimisation	108
6.3 Analyse expérimentale pour le positionnement des arrêts de bus	110
6.3.1 Définition du problème et simulation	114
6.3.2 Protocole expérimental	116
6.3.3 Précision des méta-modèles	116
6.3.4 Performances en optimisation	117
Conclusion	121
Bibliographie	124

Introduction

La mobilité urbaine est l'un des plus grands enjeux auxquels les sociétés modernes sont confrontées aujourd'hui. Chaque jour, près de trois milliards de déplacements sont effectués à travers le monde en utilisant des véhicules motorisés [140, 141]. L'augmentation constante de la demande sur les réseaux de transport exige que leur configuration soit sans cesse renouvelée ou adaptée, tout en tenant compte de l'évolution perpétuelle des espaces urbains et des habitudes changeantes des populations. Cependant, l'extension, le renouvellement ou la réorganisation de la voirie et des infrastructures urbaines peut comporter de nombreuses contraintes : un manque d'espace, des budgets financiers considérables, voire des restrictions environnementales. C'est pourquoi les experts urbanistes cherchent avant tout à améliorer la mobilité des citoyens en faisant un meilleur usage des infrastructures déjà en place dans les villes.

A ce jour, environ 54% de la population mondiale vit dans des agglomérations urbaines. La population mondiale ne cesse de croître, tout comme la proportion de la population vivant en zones urbaines, qui devrait représenter près de 66% de la population mondiale d'ici 2050 selon l'Organisation des Nations Unies [148]. Aussi, la croissance démographique et les tendances à l'urbanisation ont accru la demande en transports urbains. En outre, dans les grandes villes situées dans les pays en voie de développement, le nombre de véhicules augmente considérablement [52], ce qui intensifie encore la difficulté des problèmes de mobilité des citoyens. Bien souvent, lorsque les infrastructures routières et les systèmes de transports en commun existants ne suffisent plus à satisfaire cette demande en pleine expansion, les habitants ont tendance à utiliser des moyens de transport privés. Par conséquent, cela engendre inévitablement l'encombrement des réseaux de transport et contribue à des retards, à une augmentation du

risque d'accidents, ou à une démultiplication des rejets en émissions polluantes. Finalement, la bonne conception d'un système de transport et des infrastructures urbaines associées représentent une tâche complexe. L'amélioration de la mobilité des citoyens constitue donc un défi permanent pour les experts en aménagement urbain. Aussi, ces derniers s'appuient parfois sur des techniques d'optimisation pour conforter leurs décisions.

L'optimisation de la mobilité urbaine est un problème connu de longue date par la communauté scientifique. La conception d'un système de transport fiable et durable est un problème difficile impliquant de nombreuses interdépendances, et représente un véritable enjeu pour les urbanistes et les experts en optimisation. De nombreux algorithmes de recherche s'avèrent efficaces pour optimiser les problèmes difficiles issus des domaines de la mobilité et du transport. Par exemple, de nombreuses heuristiques à objectif unique ou à objectifs multiples sont appliquées à l'optimisation des feux de signalisation, à la conception des réseaux routiers, à l'optimisation des réseaux de bus et à la planification des itinéraires des transports publics, entre autres [5, 18, 87, 107, 143, 147].

La résolution optimale de ces problèmes requiert généralement une bonne connaissance de l'aménagement urbain et nécessite de nombreuses études préliminaires, telles que la collecte de données de recensement ou des enquêtes sur le profil type et les habitudes de déplacement des citoyens. Bien qu'assistées par les connaissances expertes en la matière fournies par les urbanistes, la collecte et la mise à jour régulière de ces informations à l'échelle d'une agglomération urbaine représentent généralement une tâche difficile. En outre, les travaux rapportés par ce mémoire s'inscrivent avant tout dans le domaine de l'informatique. Aussi, la masse d'information susmentionnée, même si elle est incomplète ou obsolète, peut être agrégée et exploitée pour l'exécution d'une multitude de simulations numériques. Ces outils puissants de simulation permettent de tester une variété de modèles de la mobilité urbaine, afin de prouver ou de réfuter leur faisabilité et leur justesse, et visent à susciter la réflexion en confrontant les points de vue des experts en aménagement et des experts en optimisation. Toutefois, de telles simulations sollicitent généralement des temps et des ressources calculatoires substantiels, qui soulèvent d'autres défis en termes d'optimisation.

Par conséquent, les travaux rapportés par ce mémoire visent à automatiser la

connaissance des experts en aménagement urbain et proposent des méthodes d'optimisation pour résoudre une variété de problèmes de mobilité, applicables à des villes du monde entier. De nombreux défis peuvent être identifiés, tant du côté des urbanistes, pour comprendre la dynamique d'une ville et anticiper ses changements futurs, que du côté des experts en optimisation, pour comprendre les interactions entre les variables d'un problème et déterminer les mécanismes qui permettent la conception d'algorithmes de résolution efficaces. Aussi, ces travaux visent principalement deux objectifs :

- L'analyse rigoureuse des caractéristiques intrinsèques aux problèmes d'optimisation de la mobilité urbaine, afin de parvenir à une meilleure compréhension de ces défis,
- La conception de techniques d'optimisation performantes sur la base des analyses précédentes, capables de résoudre des problèmes difficiles pour l'optimisation de la mobilité avec un budget calculatoire restreint, afin de faire face à la spécificité d'une variété d'espaces urbains à travers le monde.

Les travaux rapportés dans la suite de ce mémoire sont organisés en deux parties. La première partie propose un aperçu de la littérature pour les problèmes d'optimisation et de mobilité urbaine. Le chapitre 1 définit les problèmes d'optimisation, présente leurs diverses typologies et introduit une variété de techniques d'analyses et de résolution. Le chapitre 2 offre un panorama des travaux de la littérature traitant de la simulation des flux de trafic et de l'optimisation de la mobilité urbaine. La deuxième partie présente les contributions scientifiques valorisées dans le cadre des travaux de cette thèse. Le chapitre 3 propose un benchmark de référence pour l'étude de la mobilité urbaine. Le chapitre 4 tire parti d'une analyse approfondie de ce benchmark et propose un modèle d'importance des variables dans un problème d'optimisation. Le chapitre 5 présente un algorithme adaptatif pour l'optimisation du réglage des feux de signalisation, fondé sur une méthode d'apprentissage par renforcement et sur la connaissance des espaces urbains établie par les chapitres précédents. Enfin, le chapitre 6 propose un algorithme d'optimisation assistée par un modèle de substitution, pour l'optimisation difficile et coûteuse d'un système de transports en commun.

Première partie

Problèmes d'optimisation et mobilité urbaine

Problèmes d'optimisation

Introduction

Ce premier chapitre donne une définition formelle des problèmes d'optimisation. Il introduit diverses typologies pour ces problèmes et décrit les principaux procédés algorithmiques nécessaires à leur résolution. Il définit une variété de métriques pour la description et l'analyse des caractéristiques intrinsèques aux problèmes d'optimisation. Il introduit les méthodes d'optimisation assistées par des modèles de substitution pour la résolution de problèmes d'optimisation difficiles. Enfin, il dresse une liste des principaux problèmes d'optimisation étudiés dans la littérature.

Résoudre un problème d'optimisation consiste à trouver la meilleure solution pour un critère donné, à partir d'un espace de recherche. Dans ce mémoire, l'espace de recherche est défini et contient l'ensemble des solutions possibles, tandis que le critère, également appelé fonction objectif, permet d'évaluer la qualité d'une solution. La recherche de la solution optimale s'effectue par la minimisation (ou la maximisation) du critère de qualité. Formellement, dans le cadre d'une minimisation :

$$x^* = \operatorname{arg\,min}_{x \in \mathcal{X}} f(x) \tag{1.1}$$

avec x une solution candidate au problème, \mathcal{X} l'espace de recherche, f la fonction

objectif et x^* la solution optimale. Dans le contexte des travaux rapportés par ce mémoire, une solution candidate est composée de variables de décision :

$$x = (x_1, \dots, x_n) \quad (1.2)$$

avec n la dimension du problème d'optimisation.

Les algorithmes d'optimisation permettent de rechercher la solution optimale. La procédure consiste à trouver la combinaison des variables de décision qui satisfait au mieux la fonction objectif. Selon la typologie et la dimensionnalité du problème, la résolution optimale peut s'avérer difficile en un temps raisonnable. Aussi, l'algorithme d'optimisation pourra parfois chercher une solution robuste, de bonne qualité, ou une bonne approximation de la solution optimale. La section suivante introduit ces diverses typologies.

1.1 Typologies d'optimisation

Nous pouvons distinguer deux grandes classes de problèmes d'optimisation. Les problèmes mono-objectif recherchent la solution qui satisfait au mieux un unique critère de qualité, tandis que les problèmes multi-objectifs recherchent l'ensemble des solutions qui satisfont simultanément plusieurs critères. Les travaux rapportés dans ce mémoire se concentrent essentiellement sur des problèmes d'optimisation mono-objectif. Il existe diverses typologies pour ces problèmes, telles que la nature de leurs espaces de recherche, la définition analytique de leurs critères de qualité, ou la dérivabilité de leurs fonctions objectif. Ces propriétés intrinsèques aux problèmes influencent particulièrement le choix des techniques d'optimisation à privilégier. La suite de cette section introduit les principales typologies des problèmes d'optimisation rencontrés dans le contexte des travaux de cette thèse, en commençant par exposer la nature de leurs espaces de recherche.

1.1.1 Espace de recherche

L'espace de recherche définit l'ensemble des solutions possibles à un problème d'optimisation. Selon la représentation formelle du problème, on distingue généralement les domaines de l'optimisation numérique et de l'optimisation combinatoire.

L'optimisation numérique, ou continue, consiste à minimiser (ou maximiser) un critère de qualité modélisé par une fonction à variables réelles :

$$f : \mathbb{R}^n \rightarrow \mathbb{R} \quad (1.3)$$

Ainsi, les solutions candidates qui forment l'espace de recherche sont composées de variables de décisions exprimant des valeurs continues. Les problèmes d'optimisation numérique suscitent de nombreux défis dans la recherche d'algorithmes de résolution efficaces [68], et sont très largement répandus dans la modélisation mathématique des phénomènes du monde réel [81, 130]. Par exemple, le succès récent des techniques d'apprentissage profond réside en partie dans la résolution d'un problème d'optimisation numérique [96]. En effet, nous pouvons naturellement interpréter le processus d'apprentissage artificiel comme la minimisation d'une erreur.

L'optimisation combinatoire, ou discrète, consiste à minimiser (ou maximiser) un critère de qualité à partir d'un espace de recherche fini et dénombrable. Les variables du problème sont discrètes (par exemple des entiers), et peuvent exprimer des données qualitatives ou quantitatives. De façon naïve, il n'y a donc *pas de problème* : il suffirait de lister toutes les solutions possibles pour sélectionner la meilleure d'entre elles. Toutefois, cette approche est rarement réalisable en un temps raisonnable, notamment lorsque la dimension des problèmes d'optimisation est élevée [33]. Compte tenu de la nature NP-difficile de la majorité des problèmes combinatoires [125], le défi de la résolution réside dans la conception d'algorithmes plus efficaces que la simple énumération des solutions candidates [97]. Les problèmes d'optimisation combinatoire portent tant sur des questions académiques, telles que les problèmes de coloration de graphe [27] ou de voyageur de commerce [128], que sur des problèmes du monde réel, comme l'affectation des portes d'aéroport [42] ou le pilotage d'un cœur

d'une centrale nucléaire [117].

Certains problèmes d'optimisation combinatoire peuvent se définir comme des problèmes d'optimisation pseudo-booléenne. Ceux-ci consistent à optimiser un critère de qualité exprimé par une valeur réelle, à partir d'un espace de recherche composé de variables de décision booléennes :

$$f : \{0,1\}^n \rightarrow \mathbb{R} \quad (1.4)$$

Ainsi, la structure des solutions candidates est une chaîne binaire constituée de 0 ou de 1. Cette représentation permet à chaque variable de décision de s'exprimer (1) ou non (0) lors de l'évaluation de la fonction objectif. Les problèmes d'optimisation pseudo-booléens suscitent l'intérêt pour la résolution de problèmes théoriques (voir la section 1.5), tout comme pour la résolution des problèmes du monde réel tels que la recherche médicamenteuse [119], le contrôle de la contamination dans une chaîne d'approvisionnement alimentaire [74] ou le positionnement optimal des arrêts de bus [98], entre autres.

Enfin, il convient de préciser qu'il existe des problèmes d'optimisation qui ne sont pas des problèmes d'optimisation numérique, ni combinatoire. C'est par exemple le cas pour les problèmes de programmation génétique [39, 90].

1.1.2 Réglage des paramètres

Selon la nature et la quantité d'information à disposition des techniques de résolution, il est fréquent de distinguer deux typologies d'optimisation dans la littérature : les techniques hors ligne et les techniques en ligne.

L'optimisation hors ligne désigne les problèmes d'optimisation qui disposent d'une information complète. A titre d'exemple, si les déplacements de population dans une ville sont connus pour un jour type, alors chercher le réglage des feux de circulation qui engendre le moins de congestion pour ce jour revient à résoudre un problème d'optimisation hors ligne [106]. En effet, l'information est complète puisque les flux de population sont estimés au préalable (par exemple à l'aide de données de comptage ou de recensement).

Quant à l'optimisation en ligne, elle désigne les problèmes d'optimisation

qui font face à une connaissance incomplète ou non fiable de l'avenir. Le réglage optimal des feux pour une journée type, trouvé par la technique d'optimisation hors ligne de l'exemple précédent, risque de devenir obsolète si un ordre de confinement est imposé dans la ville pour le jour suivant. En effet, le peu d'usagers autorisé à sortir serait contraint d'attendre inutilement à certains feux de signalisation, bien que la chaussée soit complètement dégagée. Les méthodes de résolution en ligne doivent donc prendre des décisions *à la volée*, en tenant compte de la situation présente et en émettant parfois des hypothèses sur le futur (par exemple en analysant un flux de données de capteurs ou de vidéo-surveillance au cours de l'optimisation [114]).

1.1.3 Dérivabilité de la fonction objectif

Quand cela est possible, la résolution d'un problème d'optimisation se base généralement sur la dérivabilité de la fonction objectif. C'est typiquement le cas pour le domaine de l'optimisation numérique, où l'expression analytique de la fonction est connue et dérivable [81, 130]. La méthode bénéficie de la connaissance du gradient de la fonction pour guider la recherche vers des solutions de meilleures qualités. Par exemple dans le cadre d'une minimisation, l'algorithme du gradient consiste à démarrer d'une solution candidate (un point de la fonction), puis à effectuer successivement des déplacements dans la direction opposée au gradient, de manière à faire décroître la fonction objectif. Les méthodes basées sur ce principe sont particulièrement efficaces pour converger vers une solution optimale¹ tout en recourant à peu d'évaluations de la fonction à optimiser. Elles peuvent résoudre des problèmes avec de nombreuses variables de décision, et nécessitent généralement peu d'ajustement dans leurs paramétrages [152]. Toutefois, ces techniques ont tendance à converger vers des optima locaux (lorsque la fonction à minimiser n'est pas convexe), et peuvent être susceptibles au bruit numérique. Le recours à des mécanismes de recherche stochastique peut contribuer à y remédier [21].

Parfois, la dérivée de la fonction objectif est peu fiable, difficile à calculer, ou simplement indisponible. C'est particulièrement le cas pour les problèmes

1. Notamment lorsque la fonction à optimiser est convexe, tel qu'illustré par la Figure 1.1.

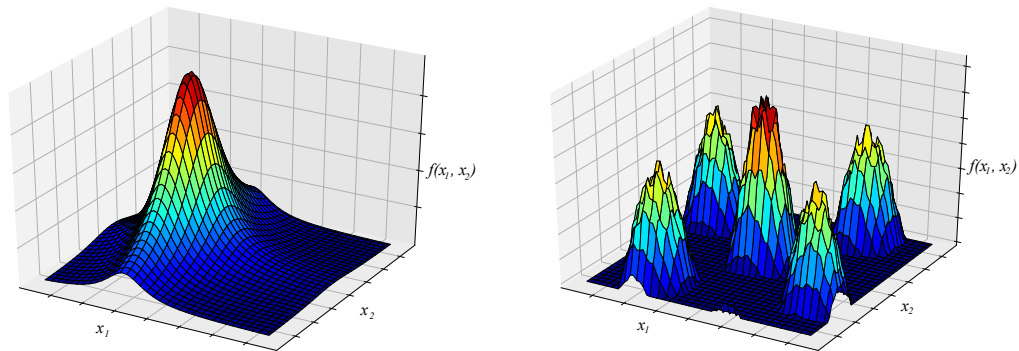


FIGURE 1.1 – A gauche, une fonction convexe avec un optimum global. A droite, une fonction non convexe et bruitée avec plusieurs optima locaux [102].

d'optimisation combinatoire où la fonction objectif est non convexe ou bruitée, comme illustré à la Figure 1.1. Pour ces classes de problèmes, les algorithmes d'optimisation basés sur le calcul du gradient sont inutilisables [163, 89]. Pourtant, de nombreuses applications en sciences [44, 63], en médecine [110, 123] ou en ingénierie [67], requièrent l'optimisation de fonctions non dérivables. Aussi, d'autres techniques de résolution sont employées, telles que les métaheuristiques, introduites à la section 1.2.

1.1.4 Définition analytique du problème

La connaissance ou la définition analytique de la fonction objectif constitue une caractéristique importante pour la conception de techniques d'optimisation efficaces et robustes. Aussi, on distingue généralement les problèmes d'optimisation dits boîtes blanches, boîtes noires ou boîtes grises.

L'optimisation boîte blanche désigne les problèmes d'optimisation où la fonction objectif est définie formellement (par exemple, les problèmes *MaxSat* [69]). Les méthodes d'optimisation peuvent donc exploiter les caractéristiques algébriques du problème pour le résoudre efficacement.

L'optimisation boîte noire désigne, au contraire, les problèmes d'optimisation où aucune information sur la définition de la fonction objectif n'est disponible. D'une manière générale, cette typologie d'optimisation peut également désigner des problèmes dont la définition de la fonction objectif est connue (par exemple,

le code source d'un simulateur de réacteur nucléaire [117]), mais difficilement prédictible et exploitable par des études analytiques. Dans ce cas, la qualité d'une solution candidate ne peut être évaluée que par une simulation ou un calcul, généralement coûteux en temps et en ressources (par exemple, une simulation informatique de mobilité urbaine nécessite plusieurs minutes de calcul [101]). Dans ce contexte, le choix d'une méthode d'optimisation adaptée relève plus de l'intuition et de l'analyse expérimentale que d'études analytiques.

A la frontière de l'optimisation boîte blanche et boîte noire, on distingue l'optimisation boîte grise, qui désigne les problèmes d'optimisation où seule une connaissance partielle de la fonction objectif est disponible (par exemple, le niveau d'interaction entre les variables de décision [32]).

Lorsque la typologie des problèmes d'optimisation est réputée difficile, tels que les problèmes d'optimisation boîte noire ou les problèmes dont la fonction objectif n'a pas de dérivée disponible, il est d'usage d'avoir recours à des techniques de résolution globales, également désignées par le terme de métaheuristiques.

1.2 Métaheuristiques

Les métaheuristiques sont des algorithmes d'optimisation visant à résoudre des problèmes d'optimisation difficiles. Elles peuvent être employées pour la résolution d'une variété de problèmes d'optimisation, et s'avèrent particulièrement adaptées à l'optimisation de problèmes pour lesquels aucune méthode classique plus efficace n'est connue. Ces techniques ne garantissent pas de trouver la solution optimale (bien qu'elles puissent parfois la découvrir). Toutefois, elles permettent généralement d'en obtenir une bonne approximation en requérant un temps de calcul raisonnable. En outre, ces algorithmes sont applicables aussi bien aux problèmes d'optimisation continue qu'aux problèmes d'optimisation combinatoire. Généralement, les métaheuristiques se distinguent selon deux catégories : les algorithmes à *une* solution et les algorithmes à *population* de solutions [145]. Cette section en présente les principes algorithmiques et définit tout d'abord les processus fondamentaux et communs à ces méthodes : les opérateurs de variation.

1.2.1 Opérateurs de variation

La plupart des métaheuristiques reposent sur un ou plusieurs opérateurs de variation. Ces opérateurs altèrent les propriétés internes d'une solution² pour en former une autre, également appelée solution voisine. La variation permet ainsi le parcours de l'espace de recherche. Il existe une multitude d'opérateurs de variation, qui dépendent de la représentation numérique de la solution [11]. Par exemple, dans un espace de recherche booléen, le voisinage d'une solution pourrait être l'ensemble des solutions situées à une distance de Hamming égale à 1 de la solution originale, ce qui revient à utiliser un opérateur de variation d'inversion de variables (communément désignée par le terme *bitflip* en anglais). Pour l'espace des permutations, un opérateur de variation pourrait effectuer l'échange de deux variables de décision choisies aléatoirement dans une solution. Pour le contexte des algorithmes à population de solutions, un opérateur de variation pourrait effectuer le croisement entre deux solutions issues de la population afin d'en concevoir une troisième qui hériterait de la combinaison des caractéristiques des deux autres [62]. D'une manière générale, la résolution d'un problème d'optimisation dépendra de la capacité de l'algorithme à explorer et à exploiter efficacement l'espace de recherche à l'aide de ces opérateurs. L'exploration favorise la recherche dans tout l'espace, tandis que l'exploitation concentre la recherche autour d'une solution spécifique.

1.2.2 Algorithmes à solution unique

Les algorithmes de recherche à solution unique constituent les fondements de la plupart des métaheuristiques pour l'optimisation. Cette section introduit trois des méthodes les plus fréquemment rencontrées dans la littérature : la méthode *hill-climbing* stochastique, la recherche tabou et le recuit simulé.

Méthode *hill-climbing* stochastique

La méthode *hill-climbing*, ou méthode du grimpeur, est typiquement une technique d'exploitation maximale. Elle est à la base de beaucoup de métaheuri-

2. Sauf mention contraire dans la suite du mémoire, nous emploierons le terme *solution* pour désigner une solution *candidate* au problème d'optimisation.

tiques, et consiste à sélectionner une solution améliorante à partir d'un ensemble de solutions voisines. La recherche de proche en proche s'arrête généralement dès que plus aucun voisin n'est améliorant. Dans sa version stochastique implémentée par l'Algorithme 1, des solutions voisines x' sont échantillonnées aléatoirement puis évaluées, jusqu'à ce que l'une d'entre elles améliore la meilleure solution connue jusqu'à présent. Cette solution constitue alors le prochain point de départ pour l'itération suivante de l'algorithme. D'une manière générale, ces techniques peuvent rester coincées dans des optima locaux. Toutefois, l'opérateur de variation stochastique (implémenté à la ligne 3) a de bonnes chances de les éviter ou de s'en échapper. Du point de vue des stratégies évolutionnaires (voir la section 1.2.3), cette métaheuristique s'apparente à l'algorithme (1+1)-ES [112, 75].

Algorithme 1 : Méthode *hill-climbing* stochastique (minimisation).

```
1  $x \leftarrow$  initialize random solution
2 while stopping criterion not met do
3    $x' \leftarrow$  mutate  $x$ 
4   if  $f(x') < f(x)$  then
5      $x \leftarrow x'$ 
6   end
7 end
```

Recherche tabou

Classiquement, la recherche d'un optimum démarre à partir d'une solution initiale, puis à chaque nouvelle itération, l'algorithme sélectionne la meilleure solution voisine au sens de la fonction objectif. Parfois, ce processus peut conduire l'algorithme à sélectionner une solution qui détériore la fonction objectif : c'est le cas lorsque toutes les solutions voisines sont de moins bonne qualité que la solution courante. Cependant, il est important de souligner que c'est à partir de ce mécanisme que l'heuristique peut s'échapper d'un optimum local. Néanmoins, le risque est de retomber, à l'itération suivante, sur le minimum local auquel l'algorithme vient d'échapper. Aussi, la recherche tabou [60], ou *Tabu Search* en anglais, est une métaheuristique qui intègre une notion de mémoire.

La recherche tabou interdit à l'algorithme de revenir sur des solutions explorées récemment. Pour éviter que des solutions ne soient considérées plusieurs fois, celles-ci sont enregistrées dans une file avec une taille limitée, également appelée liste tabou. Ainsi, la sélection d'une solution de cette liste est rendue impossible pendant un certain nombre d'itérations. Cette liste doit conserver l'expression complète des solutions, ce qui peut nécessiter la sauvegarde d'une grande quantité d'information pour certains types de problèmes. Toutefois, cette difficulté peut être contournée en n'archivant que les opérateurs de variations et la valeur de la fonction objectif associée à chaque itération. De nombreuses variantes existent, principalement au niveau de la définition du voisinage et de la gestion de la mémoire [3, 58].

Recuit simulé

L'algorithme du recuit simulé [30], ou *Simulated Annealing* en anglais, est directement inspiré d'un processus utilisé dans le domaine de la métallurgie. Le recuit (réel) consiste à alterner des cycles de refroidissement lent et de réchauffage afin de stabiliser l'énergie d'un matériau. Dans sa transposition pour le domaine de l'optimisation, l'algorithme du recuit simulé considère la fonction objectif comme la quantité d'énergie fictive E d'un matériau. A partir d'une solution initiale, un opérateur de variation est appliqué à chaque itération de la recherche. La solution ainsi obtenue est sélectionnée pour la prochaine itération si elle est améliorante au sens de la fonction objectif, sinon, elle peut tout de même être sélectionnée avec une probabilité $e^{-\frac{\Delta E}{T}}$. Il est supposé qu'il y a équilibre thermodynamique après un certain nombre d'itérations, qui est généralement déterminé empiriquement. L'algorithme diminue alors la température T , avant de réaliser une nouvelle série de variations [78].

1.2.3 Algorithmes à population de solutions

Contrairement aux algorithmes à solution unique, les algorithmes à population de solutions font évoluer un ensemble de solutions, parfois désignées par le terme *individus*, afin de résoudre un problème d'optimisation [70]. Cette section introduit trois des méthodes les plus fréquemment rencontrées dans

la littérature : les algorithmes évolutionnaires, l'évolution différentielle et les algorithmes de colonies de fourmis.

Algorithmes évolutionnaires

Les stratégies évolutionnaires forment une famille d'algorithmes à population de solutions, qui s'inspire de la théorie de l'évolution : le principe Darwinien est de faire en sorte que les individus les plus adaptés survivent et se reproduisent. Dans un contexte d'optimisation, la pression de l'environnement est simulée par la fonction objectif. La survie est caractérisée par un opérateur de sélection (le choix des meilleurs individus au sens de la fonction objectif), et la reproduction est donnée par un opérateur de variation. Ces méthodes sont usuellement stochastiques, puisqu'au moins un de ces opérateurs (en général l'opérateur de variation) utilise des processus aléatoires. Une implémentation d'algorithme évolutionnaire est proposée par l'Algorithme 2. Une population de solutions échantillonnées aléatoirement constitue le point de départ de la recherche. A chaque itération de l'algorithme, un ensemble de solutions est sélectionné parmi la population. Cet ensemble est généralement constitué d'individus qui satisfont le mieux la fonction objectif. Ces individus seront les géniteurs de la prochaine génération de solutions, par application d'opérateurs de mutation ou de croisement. Ainsi, la stratégie évolutionnaire a tendance à converger vers des populations de solutions de qualités croissantes, qui auront hérité des meilleures caractéristiques des populations génitrices.

Algorithme 2 : Algorithme évolutionnaire.

```
1  $P = \{x_1, \dots, x_n\} \leftarrow$  initialize random solutions
2 while stopping criterion not met do
3   | genitor  $\leftarrow$  selection from  $P$ 
4   | children  $\leftarrow$  mutate genitor
5   |  $P \leftarrow$  replace ( $P$ , children)
6 end
```

Evolution différentielle

Une variante de base de l'algorithme d'évolution différentielle [144] considère une population de solutions, également désignées sous le terme d'agents, représentées sous forme de vecteurs de variables de décision. L'algorithme effectue une série de déplacements dans l'espace de recherche en utilisant des opérations algébriques simples sur les vecteurs, afin de combiner les positions des agents existants dans la population. Ainsi, pour trois agents a_1 , a_2 et a_3 sélectionnés aléatoirement dans la population, un nouvel agent a est généré en calculant :

$$a = a_1 + m \times (a_2 - a_3) \quad (1.5)$$

avec m une constante appelée facteur de croisement. Si la nouvelle position d'un agent constitue une amélioration au sens de la fonction objectif, l'agent est accepté et remplace l'agent géniteur dans la population. Sinon, l'agent généré est simplement rejeté. Ce processus est répété pour chaque agent de la population. Ainsi, certains des agents originaux sont remplacés par de meilleures solutions et, après de nombreuses itérations, l'ensemble de la population finit par converger vers une solution optimale.

Colonies de fourmis

Les algorithmes à colonie de fourmis sont inspirés par le comportement social des fourmis, ou d'autres espèces formant un superorganisme [45, 46]. Bien qu'elles possèdent individuellement des capacités cognitives limitées, les fourmis sont capables de s'organiser collectivement pour accomplir des tâches complexes. L'une des tâches les plus importantes pour la fourmilière est la recherche de nourriture. Lorsqu'une fourmi trouve une source de nourriture, celle-ci rentre à la fourmilière en déposant des phéromones sur son trajet. Ces phéromones étant attractives, les fourmis qui passent à proximité ont tendance à suivre cette piste, renforçant ainsi l'attractivité du chemin. Les phéromones étant volatiles, les pistes les plus longues finissent par disparaître. A terme, l'ensemble des fourmis détermine donc le chemin le plus court entre la fourmilière et la source de nourriture. La Figure 1.2 propose une illustration de ces processus.

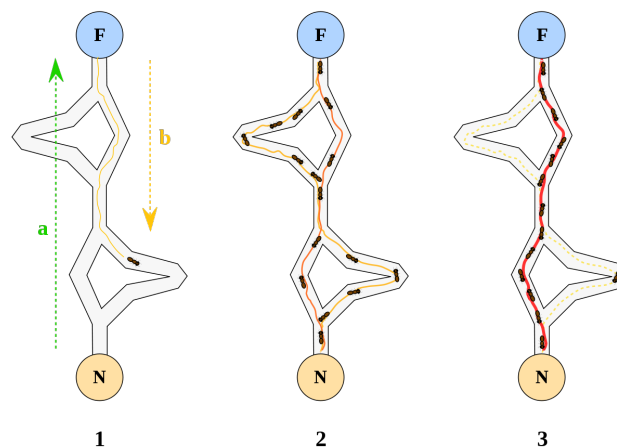


FIGURE 1.2 – Une fourmi trouve la source de nourriture F via une piste (a) puis revient au nid N en déposant des phéromones (b). Les fourmis empruntent indifféremment les chemins possibles vers la nourriture, mais les phéromones rendent le chemin le plus court plus attractif. Les phéromones s'évaporent sur les autres chemins (les plus longs) [36].

Finalement, une fourmi seule ne peut trouver la piste la plus courte : c'est l'intelligence collective qui est à l'oeuvre dans la découverte du plus court chemin. A l'origine employés pour la résolution du problème du voyageur de commerce, les algorithmes de colonies de fourmis ont été appliqués à un grand nombre de problèmes d'optimisation combinatoire [111].

1.2.4 Compromis exploration et exploitation

Comme introduit à la section 1.2.1, la robustesse d'une métaheuristique pour la résolution d'un problème d'optimisation dépend de sa capacité à explorer et à exploiter efficacement l'espace de recherche. Cependant, équilibrer le compromis entre l'exploration et l'exploitation n'est pas une tâche évidente. Aussi, les métaheuristiques sont parfois associées à divers procédés qui aident à décider si l'algorithme doit plutôt explorer ou exploiter l'espace de recherche pour les itérations ultérieures.

Un modèle historique pour concevoir des stratégies de recherche d'équilibre optimal est défini par les problèmes de *bandits manchots* [93], qui impliquent k bras de bandits avec des probabilités de récompense inconnues. A chaque

étape, un joueur ou un programme sélectionne un bras et reçoit une récompense. L'objectif est de maximiser la récompense cumulée au cours de toutes les étapes S . Cela équivaut à minimiser le regret, défini comme la perte induite vis-à-vis du meilleur bras. Une technique efficace pour résoudre ce problème est l'algorithme d'*Upper Confidence Bound* (UCB) [10]. A chaque itération, l'algorithme sélectionne le bras i qui maximise la quantité :

$$\hat{r}_i + R \cdot \sqrt{\frac{2 \ln S}{s_i}} \quad (1.6)$$

avec \hat{r}_i la récompense empirique moyenne observée du bras i et s_i le nombre de fois que le bras i a été sélectionné au fil des S étapes. Cette équation cherche à assurer le compromis entre l'exploitation et l'exploration, équilibré par une constante R . L'exploitation tend à sélectionner le bras qui a la récompense moyenne optimale, tandis que l'exploration tend à sélectionner le bras qui a été le plus rarement choisi.

L'utilisation d'un tel algorithme est répandu dans le domaine de l'optimisation, en particulier lorsque le processus d'optimisation peut être réalisé avec divers opérateurs de variation dont le potentiel est inconnu (en termes d'amélioration de la fonction objectif). Ainsi, l'algorithme apprend l'opérateur à sélectionner, tout en ayant tendance à privilégier l'opérateur le plus gratifiant au cours de l'optimisation. Récemment, Liu *et al.* ont utilisé les algorithmes UCB pour optimiser une solution partie par partie [104]. Chaque partie est considérée comme le bras d'un bandit. À chaque itération du processus d'optimisation, c'est-à-dire à chaque fois qu'une variable de décision est altérée, la différence induite dans la fonction objectif est calculée comme suit :

$$r = f(x) - f(x') \quad (1.7)$$

avec x la solution actuelle et x' la solution altérée. La moyenne empirique de ces différences représente le terme d'exploitation pour chaque bras dans la formule UCB de l'Equation 1.6. Ainsi, l'algorithme est poussé à focaliser la recherche sur la partie de la solution qui offre la meilleure amélioration de la fonction objectif.

Enfin, la diversité de typologies des problèmes d'optimisation, ainsi que la

multitude d’approches algorithmiques pour leur résolution, soulève parfois la question de savoir quelle technique est la mieux adaptée à l’optimisation d’un problème donné. Aussi, la section suivante présente un outil pour décrire et analyser la géométrie des espaces de recherche, du point de vue des algorithmes d’optimisation.

1.3 Paysages de fitness

L’étude des paysages de fitness, ou *fitness landscapes* en anglais, est un outil pour décrire une représentation métaphorique de l’espace de recherche (sommets, vallées, plateaux) [109], comme illustré à la Figure 1.3. Cette description permet d’aborder avec discernement un processus d’optimisation. Elle établit également un portrait de la structure du problème, grâce à un ensemble de métriques qui permettent de quantifier et de comparer la difficulté de la recherche pour divers opérateurs de mutation, typologies du problème ou fonctions objectif. Conformément aux travaux en optimisation combinatoire [142], un paysage de fitness est formellement défini par un triplet :

$$(\mathcal{X}, \mathcal{N}, f) \tag{1.8}$$

avec $\mathcal{N} : \mathcal{X} \rightarrow 2^{\mathcal{X}}$ la relation de voisinage entre les solutions, \mathcal{X} l’espace de recherche et f la fonction objectif.

Dans le contexte des travaux rapportés par ce mémoire, la suite de cette section donne une courte synthèse de la littérature concernant l’étude des paysages de fitness. En outre, l’analyse des paysages repose sur un échantillonnage préalable de l’espace de recherche, permettant d’établir une variété d’estimateurs et de métriques. Un exemple de technique d’échantillonnage est introduit ci-après.

1.3.1 Marche aléatoire

La marche aléatoire est une technique d’échantillonnage récurrente dans l’étude des paysages de fitness. L’Algorithme 3 en propose une implémentation. Une première solution est générée aléatoirement et définit le point de départ

de la marche. A chaque pas, autrement dit à chaque itération de l'algorithme, sa qualité est évaluée par la fonction objectif, puis une solution voisine est échantillonnée aléatoirement. Aussi, ces algorithmes représentent typiquement des techniques d'exploration maximale qui ne sont généralement pas employées pour la résolution d'un problème d'optimisation. Néanmoins ils fournissent une approche instructive quant à l'espace de recherche du problème étudié, et permettent d'analyser la capacité d'un algorithme à résoudre un problème donné lorsqu'elles sont couplées à certaines métriques, introduites par la section suivante.

Algorithme 3 : Marche aléatoire.

```
1  $x \leftarrow$  initialize random solution
2 while stopping criterion not met do
3   | compute  $f(x)$ 
4   |  $x \leftarrow$  mutate  $x$  (see section 1.2.1)
5 end
```

1.3.2 Métriques pour l'étude des paysages

Il existe de nombreux travaux de recherche sur l'analyse des paysages de fitness, depuis les débuts des algorithmes évolutionnaires jusqu'aux développements récents [109]. A la fin des années 80, l'objectif des chercheurs était de comprendre et de caractériser la difficulté des problèmes (principalement des problèmes d'optimisation combinatoire) à l'aide d'une métrique, comme la célèbre distance de corrélation [84]. Plus récemment, avec le progrès des techniques d'apprentissage artificiel, l'analyse de la fonction objectif est utilisée pour apporter un ensemble de caractéristiques numériques qui permettent, par exemple, de prédire la durée de résolution (complexité temporelle) des algorithmes de recherche [118, 41]. L'étude des paysages de fitness a principalement été menée sur des problèmes combinatoires artificiels tels que les problèmes du voyageur de commerce, les problèmes d'affectation quadratique, les paysages \mathcal{NK} [133], ou plus récemment sur des problèmes numériques artificiels [86]. Bien que les paysages de fitness soient adaptés aux problèmes d'optimisation boîte

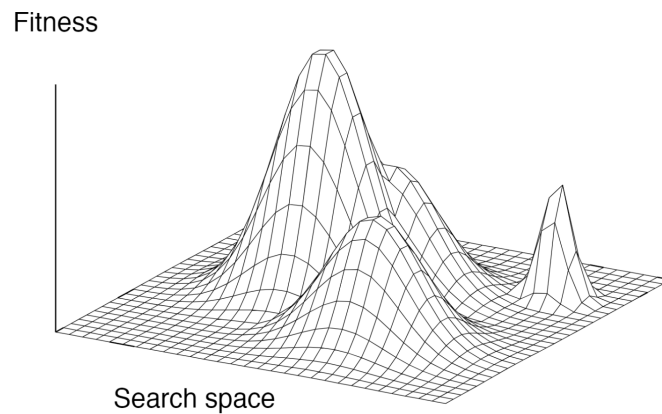


FIGURE 1.3 – Représentation d’un paysage de fitness [153].

noire où la définition analytique de la fonction objectif n’est pas nécessaire, seuls quelques travaux les ont employés pour l’étude des problématiques liées au monde réel [117].

Nous pouvons discerner deux géométries récurrentes dans la forme des paysages de fitness. Les géométries neutres dominées par les plateaux, et les géométries vallonnées dominées par les optima locaux. Les premières géométries nécessitent des marches aléatoires pour déterminer des caractéristiques liées aux réseaux neutres, tels que les graphes induits par les relations de voisinage. Dans le cas des géométries vallonnées, il est possible de définir diverses mesures telles que la détection des bassins d’attraction. Toutefois, ces métriques peuvent être coûteuses à estimer puisqu’elles sont généralement établies à partir de multiples itérations d’algorithmes de type *hill-climbing* [2]. Un autre ensemble de caractéristiques pour ces géométries comportant plusieurs optima est basé sur la rugosité du paysage, qui peut être mesurée par la valeur de corrélation et la longueur de l’autocorrélation [146]. Cette autocorrélation de la fonction objectif, introduite par Weinberger [161], définit la corrélation des valeurs prises par le critère de qualité pendant une marche aléatoire. En outre, la précision des coefficients d’autocorrélation est définie par :

$$\frac{1}{2\sqrt{L}} \quad (1.9)$$

avec L la longueur de la marche aléatoire [22]. La rugosité des paysages peut éga-

lement être quantifiée avec une métrique basée sur l'entropie [150]. La séquence des valeurs de la fonction objectif, recueillie pendant une marche aléatoire, est convertie en une séquence de -1 , 0 et 1 : 0 quand la différence entre deux valeurs consécutives est inférieure à un seuil ϵ , 1 lorsque la différence est positive et supérieure à ϵ et -1 autrement. La métrique est alors définie par l'entropie de l'ensemble composé des sous-séquences de taille 2 :

$$H(\epsilon) = - \sum_{p \neq q} P_{[pq]} \log_6 P_{[pq]} \quad (1.10)$$

avec pq les sous-séquences possibles parmi l'ensemble $\{-1, 0, 1\}$ et P leurs fréquences respectives dans la séquence originale. Cette approche permet d'estimer la rugosité d'un paysage tout en tenant compte de son éventuelle neutralité, déterminée par le seuil ϵ .

L'analyse et l'interprétation approfondies des paysages de fitness à l'aide des différentes métriques présentées dans cette section aideront les experts en optimisation à choisir l'algorithme d'optimisation le plus approprié pour résoudre un problème. Néanmoins, il reste à considérer un facteur déterminant dans le choix de l'algorithme adéquat : le temps de calcul et les ressources calculatoires nécessaires pour l'évaluation de la fonction objectif. La section suivante présente une méthode d'optimisation qui aborde cette problématique.

1.4 Optimisation assistée par méta-modèle

Comme introduit à la section 1.1, de nombreux problèmes d'optimisation difficiles nécessitent en effet l'évaluation d'une fonction objectif boîte noire, coûteuse en temps et en ressources de calcul [23, 8, 117, 101]. Pour faire face à cette difficulté, une technique classique consiste à apprendre un modèle de substitution, également appelé méta-modèle. La substitution revient à formuler des modèles mathématiques rapides à évaluer qui approximent la fonction objectif, afin de limiter le recours à son évaluation. Ainsi, à chaque étape du processus d'optimisation, le méta-modèle permet la recherche rapide de solutions prometteuses.

1.4.1 Méthodologie

Une méthode d'optimisation assistée par méta-modèle combine trois composantes. La première d'entre elles est le modèle de substitution lui-même (généralement un modèle de régression de la fonction objectif). Ce modèle doit être suffisamment expressif pour refléter la sophistication de la fonction à substituer. Cependant, il est préférable qu'il conserve une faible complexité calculatoire, afin de faciliter son apprentissage lorsque peu d'échantillons de la fonction objectif originale est disponible. La deuxième composante est une fonction d'acquisition définie à partir du modèle de substitution. Cette fonction peut être la prédiction du méta-modèle en elle-même, ou par exemple un compromis entre la prédiction du modèle et l'erreur d'estimation qui lui est associée. L'objectif de la fonction d'acquisition consiste à guider la recherche vers une solution prometteuse, tout en maintenant l'équilibre entre l'exploration qui augmente la qualité globale du modèle de substitution, et l'exploitation qui pousse le modèle vers des solutions de haute qualité. La dernière composante est l'algorithme d'optimisation de la fonction d'acquisition. Cet algorithme doit être efficace en temps et en qualité pour converger rapidement vers les solutions prometteuses qui satisfont au mieux la fonction d'acquisition.

Les méthodes d'optimisation assistées par méta-modèles suivent une approche similaire dans les domaines de l'optimisation numérique ou combinatoire [170]. Comme implémenté par l'Algorithme 4, un échantillon de solutions est d'abord évalué par la fonction originale afin de construire le modèle de substitution. Sur la base de ce modèle et d'une fonction d'acquisition, une nouvelle solution prometteuse est sélectionnée, évaluée et ajoutée à l'échantillon pour la prochaine itération de l'algorithme. Les domaines de l'optimisation numérique et combinatoire diffèrent cependant sur la définition des modèles, comme introduit à la section suivante.

1.4.2 Méta-modèles pour l'optimisation combinatoire

La conception de méta-modèles a fait l'objet de travaux de recherche conséquents dans le domaine de l'optimisation numérique [82]. En revanche, il semble que l'élaboration de modèles de substitution spécialement dédiés à l'optimi-

Algorithme 4 : Optimisation assistée par méta-modèle.

```

1  $S \leftarrow$  Initial sample  $\{(x, f(x)), \dots\}$ 
2 while computational budget is not spent do
3    $M \leftarrow$  Build model from  $S$ 
4    $x \leftarrow$  Optimize  $M$  w.r.t. an acquisition function
5   Evaluate  $x$  using  $f$ 
6    $S \leftarrow S \cup \{(x, f(x))\}$ 
7 end

```

sation combinatoire n'ait reçu que peu d'attention [15]. Bien qu'il existe des modèles de représentation mixtes qui combinent variables continues et discrètes [67], seuls quelques travaux sont pleinement consacrés à l'optimisation combinatoire. Le constat est encore plus manifeste pour les problèmes d'optimisation pseudo-booléenne [15]. Pourtant, de nombreux problèmes font face à des variables de décision discrètes dans des domaines d'application divers et variés, tels que le système de partage de vélos [135], la position des arrêts de bus [134], la découverte de médicaments [119] ou plus généralement des problèmes où les variables de décision peuvent modéliser la présence ou l'absence d'un composant. C'est également le cas des travaux rapportés aux chapitres 5 et 6 de ce mémoire. En outre, les problèmes d'optimisation associés à ces applications du monde réel sont généralement constitués de centaines de variables de décision, ce qui semble loin des méthodes de méta-modélisation actuelles [15]. La suite de cette section liste les principaux modèles de substitution de la littérature pour l'optimisation combinatoire, à savoir les approches de Krigeage, la bases de fonctions multilinéaires et la base de fonctions de Walsh.

Processus gaussiens

Une technique état-de-l'art pour la méta-modélisation combinatoire utilise une approche basée sur des processus gaussiens, également appelée Krigeage [170, 169]. Cette approche conçoit la fonction objectif à substituer comme la réalisation d'un processus gaussien :

$$f(x) \approx \mathcal{N}(\mu(x), \sigma(x)) \quad (1.11)$$

Le processus gaussien a pour moyenne l'espérance des observations de la fonction objectif :

$$\mu(x) = E[f(x)] \quad (1.12)$$

Le noyau de covariance est calculé à partir d'une notion de distance entre les solutions :

$$\sigma(x) = \exp\left(-\theta \text{distance}(x, x')^p\right) \quad (1.13)$$

avec $\theta, p \in \mathbb{R}$. La technique de Krigeage pour le domaine combinatoire est une extension directe de son application pour l'optimisation numérique [40], à la différence que la distance Euclidienne est remplacée par la distance de Hamming pour les fonctions pseudo-booléennes. Cette modélisation probabiliste permet aux méthodes basées sur des processus gaussiens d'estimer l'incertitude associée à leurs prédictions, comme illustré à la Figure 1.4.

Le Krigeage tire ainsi profit d'un algorithme d'optimisation particulièrement adapté : l'*Efficient Global Optimizer* [83]. La technique consiste à sélectionner la solution qui maximise le critère d'amélioration espérée, également appelée *Expected Improvement* (EI) en anglais. Cette fonction d'acquisition tient compte de l'incertitude estimée par le processus gaussien afin de sélectionner des solutions prometteuses. Cette sélection vise à augmenter la précision du modèle tout en le guidant vers des solutions de meilleures qualités. Toutefois, la complexité du calcul de l'EI est en haute complexité, et ne peut pas être réduite par certaines techniques classiques issues de l'optimisation combinatoire, telles que l'évaluation incrémentale [34]. En outre, les auteurs utilisent un algorithme génétique pour maximiser la fonction EI. Ils démontrent que cette méthode surpasse la modélisation fondée sur les fonctions de base radiale [113, 168], l'un des rares travaux connexes en méta-modélisation combinatoire. D'une manière générale, ces approches sont testées sur des problèmes d'optimisation en petites dimensions, comportant jusqu'à 25 variables de décision booléennes.

Base des fonctions multilinéaires

Un travail récent a suscité l'intérêt de la communauté pour le développement de méta-modèles spécialement dédiés au domaine combinatoire. La technique

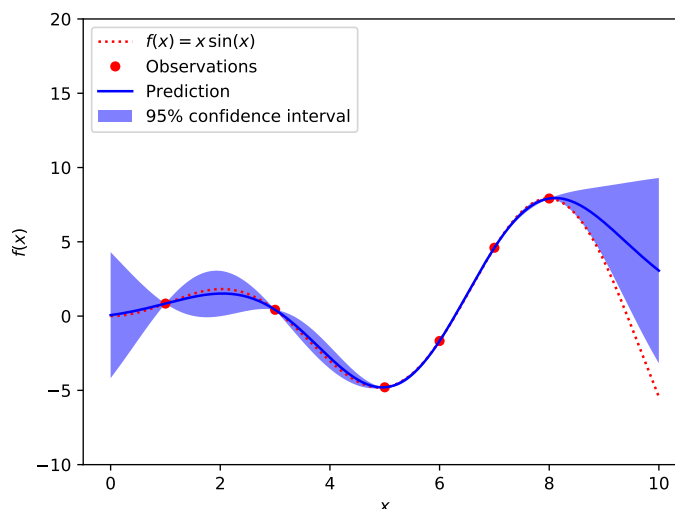


FIGURE 1.4 – Le Krigage formule un modèle de substitution à partir d’observations de la fonction à approximer. Une estimation de l’incertitude est disponible pour chaque prédiction du modèle [127].

nommée *Bayesian Optimisation of Combinatorial Structures* (BOCS) [13], est fondée sur le principe d’inférence Bayésienne. La méthode consiste à approximer une fonction objectif pseudo-booléenne par un modèle statistique, formulé à l’aide d’un polynôme multilinéaire de variables binaires :

$$P(x) = \sum_{T \subset N} a_T \prod_{i \in T} x_i \quad (1.14)$$

avec $N = \{1, \dots, n\}$, x_i les $i^{\text{èmes}}$ variables de décision, et chaque monome est pondéré par un coefficient a_T . Un aperçu de la base des fonctions multilinéaires en une dimension est illustré à la Figure 1.5. De telles modélisations multilinéaires ont déjà été utilisées pour les fonctions pseudo-booléennes dans un autre contexte [66]. Baptista *et. al.* [13] affirment ainsi que cette formulation peut tenir compte de l’interaction entre les variables de décision. Leur travail considère principalement des polynômes multilinéaires quadratiques :

$$P_2(x) = a_0 + \sum_{i \in N} a_i x_i + \sum_{i < j \in N} a_{ij} x_i x_j \quad (1.15)$$

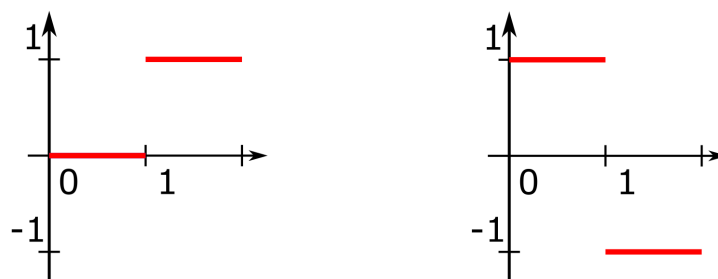


FIGURE 1.5 – A gauche, les valeurs prises par une fonction multilinéaire en une dimension. A droite, les valeurs prises par une fonction de Walsh en une dimension.

L'apprentissage du modèle réside alors dans l'approximation des coefficients de pondération à l'aide d'une technique de régression linéaire. Les auteurs utilisent l'algorithme *Sparse Bayesian Linear Regression* [108] pour faire face à la quantité quadratique des coefficients à approximer, rapportée au Tableau 1.1. L'évaluation du polynôme multilinéaire, modulo un terme de régularisation, constitue la fonction d'acquisition de la méthode `BOCS`. Cette fonction est ensuite optimisée à l'aide d'un recuit simulé classique [30]. Sur des problèmes quadratiques combinant jusqu'à 24 variables de décision booléennes, `BOCS`³ se montre plus efficace qu'une méthode de Krigeage.

Base des fonctions de Walsh

Les fonctions de Walsh, initialement définies par leur auteur du même nom [159], décrivent une base normale et orthogonale de fonctions discontinues. Ces fonctions ont la particularité de pouvoir décomposer n'importe quelle fonction de l'espace de Hilbert [167]. Dans le cadre de l'optimisation de structures combinatoires, elles peuvent être employées comme une base de représentation normale et orthogonale des espaces de recherche pseudo-booléens. Formellement, pour tout entier ℓ écrit sous une forme binaire, la fonction de Walsh φ_ℓ est définie telle que :

$$\varphi_\ell(x) = (-1)^{\sum_{i=1}^n \ell_i x_i} \quad (1.16)$$

3. Le code source est disponible sur le web : <https://github.com/baptistar/BOCS>.

avec ℓ_i et x_i qui représentent la $i^{\text{ème}}$ variable dans les chaînes binaires ℓ et x . Un aperçu de la base des fonctions de Walsh en une dimension est illustré à la Figure 1.5. Ces fonctions posent les fondements de la théorie des schémas pour l'optimisation évolutionnaire [61], et ont dernièrement été employées pour des problèmes d'optimisation boîte grise [33, 32]

Une approche récente propose l'utilisation de la base des fonctions de Walsh pour l'élaboration de méta-modèles spécialement dédiés à l'optimisation pseudo-booléenne [154]. Les auteurs montrent que la fonction pseudo-booléenne à approximer peut être substituée par le polynôme :

$$W_L(x) = \sum_{\ell \text{ s.t. } o(\varphi_\ell) \leq L} a_\ell \varphi_\ell(x) \quad (1.17)$$

avec o l'ordre de la fonction de Walsh, c'est-à-dire le nombre de variables égales à 1 dans la représentation binaire de l'entier ℓ . Le développement du polynôme est ainsi restreinte à un ordre L . Par exemple, pour l'ordre 2 :

$$W_2(x) = a_0 + \sum_{i=1}^n a_i (-1)^{x_i} + \sum_{i < j \in N} a_{ij} (-1)^{x_i + x_j} \quad (1.18)$$

Tout comme pour la méthode `wocs`, l'apprentissage du modèle réside dans l'estimation des coefficients de pondération. L'approximation de la valeur des coefficients est réalisée par des techniques classiques de régression linéaire parcimonieuse. En effet, la décomposition de l'Equation 1.17 peut être interprétée comme un modèle linéaire où les prédicteurs prennent les valeurs des fonctions de Walsh. Il est évident que le potentiel d'approximation de cette formulation croît lorsque l'ordre L augmente, aux dépens du nombre de termes dans le développement du polynôme, comme présenté dans le Tableau 1.1. Aussi, les auteurs emploient les techniques *Lars* ou *Lasso* [149], qui consistent à minimiser le nombre de coefficients non-nuls dans la formulation du modèle. Cette méthode se révèle particulièrement efficace puisque le nombre de fonctions de Walsh d'ordre L , autrement dit le nombre de coefficients de pondération à estimer, est généralement largement supérieur au budget consacré à l'apprentissage du modèle. Mathématiquement, le *Lasso* est régularisé avec la norme ℓ_1 pondérée par

TABLEAU 1.1 – Nombre de coefficients pour les décompositions multilinéaires ou de Walsh, en fonction de la dimension du problème n et de l'ordre L .

L	n					
	10	15	20	25	50	100
0	1	1	1	1	1	1
1	11	16	21	26	51	101
2	56	121	211	326	1,276	5,051
3	176	576	1351	2,626	20,876	166,751

une constante α (détaillée dans la section 6.2.1). Cette propriété géométrique fait de *Lasso* une méthode de choix pour l'apprentissage d'un modèle de substitution fondé sur la base orthogonale des fonctions de Walsh. Finalement, la précision des méta-modèles à base de fonctions de Walsh semble outrepasser celle des méthodes de Krigeage pour des problèmes combinant jusqu'à 25 variables de décision [154]. Ces modèles ont récemment été utilisés dans un contexte d'optimisation et ont fait l'objet de plusieurs contributions exposées au chapitre 6.

Finalement, les sections précédentes ont permis de présenter une variété d'études et de techniques pour l'optimisation d'une large gamme de problèmes. Aussi, à des fins de mise en application, la section suivante présente les problèmes de référence les plus couramment rencontrés dans la littérature.

1.5 Problèmes de référence

La variété des techniques d'optimisation rapportées dans ce chapitre sont généralement mises à l'épreuve pour résoudre des problèmes théoriques bien connus de la communauté scientifique. Cette section introduit une liste de quatre problèmes de référence. Les deux premiers problèmes, à savoir les problèmes du *OneMax* et du voyageur de commerce, sont des problèmes communs et bien connus de la littérature en optimisation. Les deux derniers problèmes, à savoir les problèmes *UBQP* et les paysages κ , constituent plus particulièrement les problèmes de référence pour les travaux rapportés au chapitre 6 de ce mémoire.

1.5.1 Problème *OneMax*

Le problème *OneMax* constitue un problème d'optimisation combinatoire unimodal et linéaire [49]. Il est formellement défini par :

$$f(x) = \sum_{i=1}^n x_i \quad (1.19)$$

avec x une solution représentée sous la forme d'une chaîne binaire, n la dimension de la chaîne binaire et x_i la valeur de la variable à la position i dans la chaîne. Ainsi, la solution optimale est obtenue lorsque la chaîne binaire est constituée uniquement de 1. L'initialisation du problème consiste généralement à fixer l'ensemble des variables à zéro ou à un, aléatoirement dans la chaîne. L'opérateur de variation typique pour cette classe de problèmes est celui de l'inversion de variable, plus couramment désigné par le terme anglais *bitflip*, qui consiste à inverser la valeur d'une variable dans la solution.

1.5.2 Problème du voyageur de commerce

Le problème du voyageur de commerce représente l'un des problèmes d'optimisation les plus célèbres de la littérature [17]. A partir d'une liste de villes et de la liste des distances qui les séparent deux à deux, la résolution du problème consiste à déterminer le plus court chemin qui visite chaque ville, une et une seule fois, et qui termine dans la ville de départ. La Figure 1.6 en propose une illustration. Malgré la simplicité de son énoncé, il s'agit d'un problème d'optimisation NP-difficile. En effet, le nombre de chemins à considérer explose avec la typologie combinatoire du problème : pour une liste de n villes, il existe $\frac{1}{2}(n-1)!$ chemins candidats pour les relier⁴. Cette classe de problème a été largement déclinée selon une multitude de variantes dans la littérature [77], et transposée pour de nombreuses problématiques du monde réel telles que des problèmes de logistique, d'optimisation du mouvement de machines ou même de séquençage du génome [4].

4. A partir de 71 villes, le nombre de trajets possibles est supérieur au nombre d'atomes dans l'univers observable!

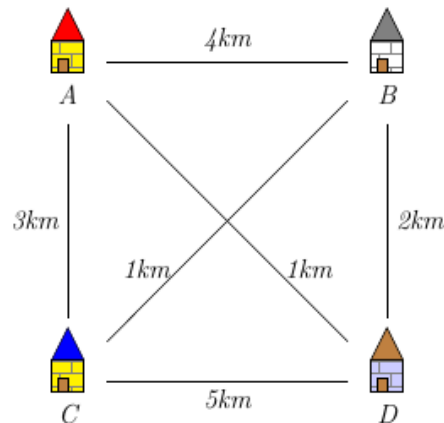


FIGURE 1.6 – Une instance du problème du voyageur de commerce. La solution optimale est obtenue avec le trajet ACBDA (7km) [37].

1.5.3 *Unconstrained Binary Quadratic Programming*

Le problème de programmation binaire sans contrainte, plus communément désigné par *Unconstrained Binary Quadratic Programming* (UBQP) en anglais, est formellement défini par la minimisation de la fonction :

$$f(x) = \sum_{i=1}^n \sum_{j=1}^n q_{ij} x_i x_j \quad (1.20)$$

avec x une solution représentée sous la forme d'une chaîne binaire composée de n variables de décision, x_i la $i^{\text{ème}}$ variable dans la chaîne binaire et $Q = (q_{ij})$ une matrice symétrique et carrée de dimensions $n \times n$, initialisée avec des coefficients aléatoires. Dans un contexte d'optimisation, cette matrice est typiquement creuse : certains coefficients sont nullifiés selon une loi de probabilité de densité, tandis que d'autres se voient attribuer un coefficient choisi uniformément dans l'intervalle $[-100, 100]$ [103]. Bien que simplistes dans leurs définitions, ces fonctions représentent des défis d'optimisation difficiles, et englobent une grande variété de problèmes combinatoires tels que la gestion du trafic, le positionnement d'infrastructures, l'allocation de ressources, ou même l'analyse économique [88]. Compte tenu de leur nature NP-difficile et de leur large gamme d'application, les problèmes UBQP motivent continuellement la communauté

scientifique à proposer de nouvelles métaheuristiques, principalement basées sur la recherche tabou [16].

1.5.4 Paysages NK

Les paysages NK, ou *NK-landscapes* (NKL) en anglais, sont des modèles introduits par Kauffman [85] afin d'étudier l'évolution de génomes. Ils définissent des problèmes d'optimisation dont la dimension et la rugosité des paysages de fitness associés peuvent être paramétrées, respectivement à l'aide des paramètres n et k . La qualité d'une solution x , représentée sous la forme d'une chaîne binaire, est évaluée par la somme de fonctions de contributions g_i . Ces fonctions tiennent compte de la valeur de chaque variable x_i et des valeurs des k autres variables interagissant avec x_i . Le problème d'optimisation consiste à maximiser la fonction ainsi définie :

$$f(x) = \frac{1}{n} \sum_{i=1}^n g_i(x_i, x_{i_1}, \dots, x_{i_k}) \quad (1.21)$$

Généralement, les fonctions de contribution g_i assignent une valeur réelle de l'intervalle $[0, 1[$ pour chaque interaction possible de x_i avec les k variables. Les fonctions g_i sont définies aléatoirement selon une probabilité uniforme. Pour chaque variable x_i , les indices des k variables i_1, \dots, i_k sont choisis uniformément parmi les $(n - 1)$ variables restantes, autres que x_i . Par conséquent, les paysages NK décrivent des problèmes d'optimisation dont la difficulté croît lorsque le paramètre k augmente. Ils offrent ainsi une variété de paysages d'optimisation, allant des géométries lisses aux géométries les plus vallonnées. Par exemple, les fonctions sont linéaires lorsque $k = 0$, alors qu'elles sont quadratiques lorsque $k = 1$. Compte tenu de ces caractéristiques, les paysages NK peuvent modéliser de manière abstraite un large éventail de problèmes d'optimisation du monde réel [162].

Résumé

Ce premier chapitre donne une définition formelle des problèmes d'optimisation, en se concentrant sur les problèmes d'optimisation combinatoire. Dans un premier temps, il introduit plusieurs typologies d'optimisation, selon la nature de l'espace de recherche, la quantité d'information disponible au cours de la recherche, ou la définition analytique de la fonction objectif. Dans un second temps, il introduit un échantillon de métaheuristiques à solution unique ou à population de solutions, qui représentent une variété d'algorithmes de recherche pour la résolution de problèmes d'optimisation de diverses typologies. Dans un troisième temps, il définit l'étude des paysages de fitness, ainsi que certains algorithmes d'échantillonnage et des métriques pour l'analyse, la description et l'interprétation des caractéristiques intrinsèques aux problèmes d'optimisation. La quatrième section introduit la méthodologie de l'optimisation assistée par méta-modèle pour la résolution de problèmes difficiles. Enfin, la dernière section présente quatre problèmes d'optimisation combinatoire de référence parmi les plus communs dans la littérature. Finalement, ce premier chapitre introduit les fondations des travaux en optimisation qui seront utilisées et approfondies dans les chapitres suivants du présent mémoire.

Problèmes d'optimisation en mobilité urbaine

Introduction

Ce deuxième chapitre présente diverses approches pour la simulation des flux urbains, introduit une variété de problèmes d'optimisation visant à améliorer la mobilité urbaine, puis expose l'importance de l'expertise urbaniste pour la conception de techniques d'optimisation robustes. Finalement, sans perte de généralisation, ce chapitre se concentre sur les problématiques de mobilité retenues par les experts urbanistes de Calais, à savoir l'optimisation du réglage des feux de signalisation et l'optimisation du positionnement des arrêts de bus.

Selon l'Organisation des Nations Unies, plus de soixante pour cent de la population mondiale habitera dans des zones urbaines d'ici les trois prochaines décennies [148]. Ce taux en constante augmentation pose des défis grandissants dans les villes modernes, tels que la conception d'un réseau routier fiable et durable ou la planification de systèmes de transports en commun efficaces. Ces enjeux visent à éviter les situations de congestion du trafic tout en garantissant un niveau de service satisfaisant aux citoyens. Cependant, l'extension, le renouvellement ou la réorganisation de la voirie peut comporter de nombreuses contraintes : un manque d'espace, des budgets financiers considérables, ou par-

fois des restrictions environnementales. Par conséquent, les villes cherchent avant tout à améliorer la mobilité de leurs voyageurs en faisant un meilleur usage des infrastructures urbaines déjà existantes. Aussi, les experts urbanistes ont fréquemment recours à des outils de simulation numérique avant d'entamer la restructuration ou le réaménagement des espaces urbains.

2.1 Simulation des flux urbains

La simulation est une discipline de plus en plus populaire dans le domaine des sciences. Les chercheurs abordent ainsi une variété de problèmes en utilisant la puissance de calcul informatique pour concevoir des environnements d'expérimentation artificiels. Ceux-ci permettent de tester des modèles scientifiques afin de prouver ou de réfuter leur faisabilité et leur justesse. La puissance de calcul des machines actuelles rend possible la simulation d'environnements expérimentaux à un rythme bien plus rapide que n'importe quel environnement réel. Ainsi, dans le cadre de la validation du modèle et compte tenu de ses hypothèses simplificatrices, les expériences menées dans le milieu simulé fournissent généralement des résultats avec quelques minutes, heures, jours, voire semaines, mois ou années en avance par rapport à ce que la même expérience donnerait dans la réalité.

La modélisation des flux de circulation fait partie des systèmes les plus étudiés dans le domaine de la simulation informatique. Les chercheurs ont mis au point différentes approches pour simuler les flux de trafic, qui peuvent généralement être classifiées en deux catégories : les approches macroscopiques et les approches microscopiques. La simulation macroscopique se base généralement sur des modèles de dispersion statistique et simule le trafic urbain comme un tout, comparable à la simulation de fluides [53, 92]. A l'opposé, la simulation microscopique modélise le comportement individuel de chaque entité dans le flux. La Figure 2.1 propose une illustration de simulation microscopique. Elle offre ainsi une simulation beaucoup plus précise et détaillée du trafic au niveau individuel, aux dépens du coût calculatoire nécessaire. Ces techniques sont majoritairement fondées sur des systèmes multi-agents [50, 126, 12, 129, 73, 91], bien qu'elles se basent parfois sur des systèmes de particules [138] ou



FIGURE 2.1 – Simulation microscopique multi-agents des véhicules motorisés, des piétons et des feux de signalisation dans un quartier de Calais. La simulation est réalisée avec le simulateur Vissim [157].

sur des automates cellulaires [139]. Il existe également d'autres approches de simulation qui combinent les caractéristiques des systèmes macroscopiques et microscopiques [165, 79, 124].

En outre, il est de plus en plus fréquent de rencontrer dans la littérature des travaux nécessitant la simulation du trafic urbain à un niveau de précision proche de la réalité, en particulier pour les simulations qui nécessitent une modélisation réaliste de la dynamique des agents en mouvement. En effet, les simulateurs de trafic microscopiques actuels modélisent des comportements physiques simplifiés pour décrire le mouvement des véhicules. D'une manière générale, la simulation des trajectoires latérales lors d'un changement de voie ou la modélisation du comportement à l'intérieur de la voie, comme le maintien de la trajectoire et la négociation des virages, sont physiquement peu plausibles. Aussi, les simulations qui sont en mesure d'intégrer ces dynamiques avec un tel niveau de précision sont généralement qualifiées de submicroscopiques. Ces simulations, parfois couplées à des outils de réalité virtuelle, sont généralement

menées pour l'analyse d'une zone spécifique du réseau routier (par exemple pour l'étude précise d'un carrefour ou d'un passage à niveau), ou pour modéliser précisément la physique et le comportement des véhicules autonomes [115, 94].

Finalement, ces divers outils de simulation numérique permettent de reproduire les flux urbains typiques d'une ville, ou d'en simuler de nouveaux. Les simulations résultantes peuvent ainsi être employées pour l'optimisation d'une variété de critères nécessaires à l'amélioration de la mobilité urbaine, comme introduit par la section suivante.

2.2 Optimisation de la mobilité urbaine

L'amélioration de la mobilité urbaine peut être concrétisée par une multitude d'approches, telles que le réglage optimal des feux de signalisation, la planification efficace des transports en commun, la conception d'infrastructures et de véhicules intelligents, ou la promotion des modes de mobilité douce. Cette section rapporte les problématiques les plus fréquemment rencontrées dans la littérature et les travaux associés.

2.2.1 Réglage des feux de signalisation

Une stratégie pour l'optimisation de la mobilité urbaine réside dans le réglage optimal des feux de signalisation. La recherche des paramétrages les plus efficaces contribue non seulement à réguler les flux de circulation urbaine, mais aussi à réduire l'empreinte carbone et les émissions polluantes de chaque voyageur. En outre, un meilleur réglage des feux de signalisation peut également induire une atténuation de la vitesse des conducteurs dans certaines zones sensibles de la ville, améliorant ainsi la sécurité des piétons.

La recherche du réglage optimal des feux de signalisation motive le développement d'un grand nombre de travaux qui sont appliqués pour des villes du monde entier, comme rapporté dans la section 2.4.1. Ces travaux peuvent être classifiés selon deux catégories, conformément aux typologies introduites dans la section 1.1.2. D'une part, les méthodes hors ligne ciblent l'optimisation des feux de signalisation préprogrammés : chaque feu est réglé par un cycle fixe,

déterminé au préalable à partir d'un flux typique de voyageurs (par exemple estimé sur la base d'un comptage ou d'un recensement). D'autre part, les méthodes en ligne considèrent l'optimisation dynamique des réglages en tenant compte de l'information sur le trafic en temps réel (par exemple à l'aide d'un flux de vidéo-surveillance). Bien que ces méthodes en ligne s'avèrent efficaces à petite échelle, elles sont généralement difficiles à transposer à l'échelle d'une ville entière [160, 114, 54]. En outre, la grande majorité des feux de circulation est encore paramétrée par des réglages fixes [9, 55, 121, 132, 137, 106]. Aussi, les contributions présentées dans la section 2.4.1 ainsi qu'aux chapitres 3, 4 et 5 se concentrent essentiellement sur des problèmes d'optimisation hors ligne pour la recherche du réglage optimal des feux de signalisation préprogrammés, selon la vraisemblance du trafic urbain dans les villes étudiées.

2.2.2 Planification des transports en commun

Une autre stratégie pour l'optimisation de la mobilité réside dans la planification et la gestion efficaces des systèmes de transport en commun, tels que les métros, les trams ou les bus. La conception judicieuse de ces systèmes est la clé pour offrir aux citoyens un mode de transport plus attractif que la voiture personnelle. En effet, l'enjeu des experts en aménagement urbain est de proposer des durées et des coûts de voyage globaux suffisamment attrayants pour séduire les usagers des autres modes de transport. Une planification efficace des systèmes de transports en commun permet généralement de décongestionner le trafic automobile et d'obtenir des bienfaits dérivés, comme l'amélioration de la sécurité routière ou la réduction de la pollution sonore. Par ailleurs, l'amélioration de l'aménagement des transports publics est particulièrement recommandée pour atténuer les impacts environnementaux liés aux déplacements urbains, et peut également contribuer à revitaliser ou à renouveler l'intérêt dans certains quartiers de la ville. Par exemple, l'implantation d'une station de transit dans une zone où il n'en existait pas encore attirera sans doute de nouveaux usagers vers les transports en commun.

De nombreux travaux de la littérature proposent de résoudre ces problématiques en suivant une variété d'approches, telles que la gestion efficace du parc

automobile et de ses ressources [158, 80, 155, 59, 43], la planification judicieuse des itinéraires, des horaires et des fréquences des transports [28, 29, 95], ou le positionnement optimal des stations de transit (comme rapporté plus précisément dans la section 2.5). Chacun de ces défis est supposé tenir compte des éventuels changements de la demande, de l'interaction avec le trafic privé, des variables opérationnelles (taille du parc, fréquence des transports, budgets des opérateurs) et des caractéristiques socio-démographiques de chaque zone de l'agglomération urbaine.

2.2.3 Conception d'infrastructures et de véhicules intelligents

Lorsque la planification des transports en commun ne suffit pas à combler la demande des usagers, les citoyens se reportent généralement sur l'utilisation de leurs véhicules particuliers. Cela entraîne de nombreux effets de bord dans les agglomérations urbaines, tels que la recrudescence de situations de congestion du trafic, l'augmentation de la pollution atmosphérique, ou la hausse du nombre d'accidents. De nombreux travaux de la littérature se concentrent toujours sur la recherche de solutions appropriées et acceptables à ces problèmes, notamment en ce qui concerne la gestion des embouteillages. Ainsi, les technologies novatrices en terme de véhicules intelligents ont été largement étudiées afin de surmonter ces problèmes [31]. En effet, ce type de véhicule perçoit son environnement proche et fournit des informations à un ordinateur de bord afin d'assister le conducteur, voire de contrôler directement le véhicule dans un fonctionnement optimal.

Le véhicule intelligent représente une solution qui peut prétendre à résoudre diverses questions de mobilité en améliorant la fluidité du trafic urbain et la sécurité des citoyens [19]. Le développement de ces véhicules est généralement fondé sur trois composants : des capteurs, un ordinateur de bord, et un actuateur. Les capteurs fournissent des informations de perception, notamment des informations sur l'état du véhicule et sur son environnement proche. L'ordinateur de bord embarque des algorithmes d'optimisation et de décision qui déterminent les mouvements que le véhicule doit effectuer, sur la base des informations recueillies par les capteurs. Enfin, l'actuateur exécute l'ordre envoyé



FIGURE 2.2 – Illustration de véhicules intelligents et des communications entre ces véhicules, issue de [31].

par l'ordinateur de bord afin de déplacer le véhicule. Parfois, des systèmes de communication sont embarqués dans le véhicule intelligent, afin de favoriser l'émergence d'une coopération avec d'autres véhicules environnants. La Figure 2.2 en propose une illustration. Certaines des solutions étudiées par ces travaux sont déjà disponibles sur le marché grand public, telles que l'aide au stationnement, le régulateur de vitesse adaptatif, l'alerte de franchissement de ligne ou le système d'évitement de collisions [20]. Dans ce contexte, le monde de l'ingénierie et de la recherche se mobilise davantage pour la conception de véhicules et de carrefours autonomes [26]. Ces méthodes sont souvent considérées comme la solution future pour la régulation optimale du trafic automobile [47, 166].

2.2.4 Développement des modes de mobilité douce

La mobilité douce comprend toutes les formes de transports non motorisés qui utilisent uniquement l'énergie humaine. Bien qu'il n'existe pas de définition unique, la mobilité douce (déplacements à pied, à vélo et autres déplacements non motorisés) est une mobilité *zéro impact*, qui se veut une alternative à l'utili-

sation de la voiture particulière. Il s'agit d'un concept global de déplacements durables qui visent à accroître la qualité de vie en milieu urbain, notamment en diminuant les niveaux de congestion du trafic, de bruit, et de pollution atmosphérique. En général, la mobilité douce et sa promotion dans les politiques urbaines sont liées au problème des rejets en émissions polluantes dans la ville, générés par la circulation des véhicules motorisés par un moteur thermique. Une telle promotion est également basée sur la conviction que l'augmentation de la mobilité douce peut réduire la circulation des voitures particulières, notamment pour les trajets courts. Ainsi, l'attention croissante portée aux préoccupations environnementales a conduit de nombreux espaces urbains à développer des infrastructures et des services spécifiques dédiés à la mobilité douce.

C'est particulièrement le cas des stations de vélos en libre-service. Ces initiatives sont développées pour fournir aux citoyens les chaînons manquants dans le système de transport en commun déjà en vigueur dans l'espace urbain. Dans un réseau idéal de vélos en libre-service, l'emplacement des stations est généralement choisi de manière à ce que les ramassages et les dépôts soient équitablement répartis entre les stations. Cela permet d'éviter des opérations de rééquilibrage coûteuses et de maintenir un niveau élevé de satisfaction des usagers [25]. La mise en oeuvre d'un système efficace de vélos en libre-service et dont les stations sont bien situées représente une problématique d'optimisation bien connue de la littérature. En effet, la demande en stations de vélos est influencée par de multiples facteurs liés à la topologie des zones urbaines et à la planification des réseaux de transports publics déjà établie. À cette fin, de nombreux travaux de la littérature proposent une variété d'approches pour l'optimisation du positionnement des systèmes de vélos en libre-service en tenant compte de ces multiples facteurs d'influence [120, 105] et de données issues de systèmes d'informations géographiques [57].

2.3 Connaissance experte et robustesse des méthodes d'optimisation

Les métaheuristiques, telles que les techniques de recherche locale ou les algorithmes évolutionnaires, forment un ensemble de méthodes pertinentes pour la résolution de tels défis d'optimisation qui ont recours à des simulations numériques. Ces techniques doivent tenir compte des caractéristiques des zones urbaines spécifiques à chaque problème, afin de satisfaire au mieux les objectifs fixés par les experts en aménagement urbain. Cependant, la sélection d'un algorithme d'optimisation adapté à une ville et à ses besoins peut représenter une tâche difficile du point de vue d'un urbaniste, qui n'est pas spécialiste des techniques d'optimisation. En effet, les résultats obtenus peuvent parfois différer selon la topologie des zones urbaines ou selon la technique d'optimisation employée. En pratique, la robustesse d'un algorithme d'optimisation réside dans son paramétrage, dont l'automatisation est un sujet qui suscite l'intérêt [72, 106]. En outre, ces réglages sont généralement guidés par une connaissance experte et consacrés à des perspectives à long terme ou à des situations spécifiques. Par conséquent, même pour un expert en optimisation, les performances des algorithmes peuvent se révéler difficiles à comparer lorsque ceux-ci sont appliqués à des cas d'études hétérogènes. En d'autres termes, il n'est pas évident qu'un algorithme correctement paramétré et appliqué à un problème spécifique conserve son efficacité sur un autre problème, en particulier pour les questions de mobilité urbaine qui portent sur des villes en perpétuelle évolution. Par ailleurs, la simulation et l'optimisation des flux urbains représentent des tâches coûteuses et complexes. C'est typiquement le cas pour les techniques d'optimisation évolutionnaires, qui nécessitent un budget calculatoire d'autant plus important qu'il y a de configurations à simuler. C'est pourquoi les urbanistes se limitent généralement à l'étude et à l'optimisation de petites parties spécifiques de la zone urbaine (par exemple quelques axes routiers ou un quartier), parfois avec un nombre restreint d'objectifs.

Conformément aux stratégies d'amélioration de la mobilité urbaine retenues par les experts en aménagement de la ville de Calais, les deux sections

suivantes se concentrent sur l'optimisation du réglage des feux de circulation et sur l'optimisation du positionnement des arrêts de bus.

2.4 Optimisation du réglage des feux de circulation

Sans perte de généralisation, cette section se concentre sur la première stratégie retenue par les experts urbanistes de Calais pour l'amélioration de la mobilité urbaine, à savoir l'optimisation du réglage des feux de signalisation d'une ville.

2.4.1 Panorama des approches pour l'optimisation

Un critère de qualité récurrent pour l'optimisation du réglage des feux de signalisation est représenté par le temps de parcours des voyageurs dans la ville : un trajet de courte durée sous-entend que les conditions de déplacement de l'utilisateur sont fluides. Ce critère est parfois associé à des perspectives environnementales [56]. En effet, la fluidification du trafic évite les situations de congestion et contribue ainsi à réduire la consommation en carburant et les rejets en émissions polluantes des citadins. Comme évoqué précédemment, de nombreux algorithmes d'optimisation peuvent être employés pour la recherche du réglage optimal des feux de signalisation. Ceux-ci sont généralement associés à un outil de simulation qui fournit une estimation de la fluidité du trafic. Toutefois, ces tâches de simulation sont réputées pour leur nature boîte-noire, coûteuse en temps et en ressources de calcul. La résolution de ces problèmes d'optimisation difficile réside alors dans l'exploration efficace d'un espace de recherche vaste, tout en requérant au minimum l'outil de simulation.

Les algorithmes évolutionnaires permettent d'appréhender cette typologie d'optimisation difficile par leur capacité à faire évoluer une population de solutions qui couvre rapidement une large surface de l'espace de recherche. Ces algorithmes permettent généralement d'atteindre des réglages plus robustes que ceux établis par les procédures par défaut des urbanistes ou des simulateurs. Cela est démontré pour la ville de Chester en Angleterre, où un flux représentant environ 22 000 véhicules par heure est simulé avec le logiciel *Saturn* [147]. Le même constat est établi dans les travaux de Medina *et al.* [137], où l'optimisation

du réglage des feux est effectuée en considérant une situation de trafic typique d'un quartier de Saragosse en Espagne, composé de sept intersections. Une autre approche consiste à utiliser des algorithmes à base de logique floue, plus communément désignés par le terme *Fuzzy Logic Controllers* en anglais. Ces techniques décrivent des modèles de décision dont la définition est généralement basée sur des instructions formulées verbalement. Ainsi, le réglage des feux évolue en utilisant uniquement des informations et des connaissances locales sur le flux ou la densité du trafic. Par exemple, les travaux de Chiu *et al.* considèrent un réseau composé de six voies traversées par un flux aléatoire de véhicules [35]. Plus récemment, Odeh *et al.* ont couplé de tels contrôleurs avec une technique évolutionnaire [121]. La méthode consiste à faire évoluer une population de règles de décisions, afin d'en sélectionner le meilleur ensemble au fil du processus d'optimisation. Les auteurs simulent des flux de véhicules sur un modèle composé de quatre voies et obtiennent un réglage robuste des feux de signalisation [122]. Une utilisation de la programmation génétique peut également s'avérer efficace : un travail récent, inspiré des travaux en biologie portant sur les modifications épigénétiques, se révèle concluant pour la régulation d'un flux de circulation simulé avec une loi de Poisson sur un réseau routier composé de neuf intersections [132]. Finalement, il n'est pas inhabituel de rencontrer des approches quelque peu *insolites* parmi la profusion de littérature traitant du sujet. Par exemple, des algorithmes basés sur le comportement collectif de systèmes auto-organisés, aussi qualifiés par le terme de colonies d'abeilles, se montrent convaincants pour la recherche du réglage optimal des feux sur un réseau routier fictif composé de neuf intersections [55].

2.4.2 Définition du problème d'optimisation

La formulation des problèmes d'optimisation liés au réglage optimal des feux de signalisation est similaire pour la grande majorité des travaux de la littérature. Sans perte de généralisation, la suite de cette section considère le contexte des travaux de Armas *et al.* [7], et définit les caractéristiques du problème d'optimisation sous-jacent.

Fonction objectif

La durée du trajet des citoyens est l'une des premières informations concrètes à interpréter du point de vue des experts en mobilité urbaine. La durée moyenne des trajets au cours d'une journée est l'une des principales mesures de la congestion dans la ville. Aussi, les experts aspirent à minimiser cette quantité qui représente la fonction objectif f du problème d'optimisation sous-jacent.

Modélisation d'une solution

Un système de feux de signalisation est modélisé par trois propriétés : le cycle, le délai, et les phases. Les phases définissent la séquence des feux. Elles sont configurées indépendamment pour chaque axe routier qui compose l'intersection (typiquement les axes nord-sud et ouest-est). Elles définissent essentiellement la durée pendant laquelle les feux d'un axe affichent un signal vert, orange ou rouge. La somme des ces durées respectives définit la durée d'un cycle du système. Enfin, le délai définit la durée d'attente avant que les feux d'un système ne commencent leur cycle. Ce réglage est particulièrement utile pour synchroniser les feux connexes à un même axe routier, afin d'y favoriser la création d'ondes vertes (lorsque les feux d'un même axe passent au vert successivement). Formellement, un système de feux de signalisation F est encodé par :

$$F_h = (C_h, \Theta_h, \Phi_{h,1}, \dots, \Phi_{h,r}) \quad (2.1)$$

avec h l'identifiant du système, C_h la durée de cycle, Θ_h l'instant du délai et $\Phi_{h,1}, \dots, \Phi_{h,r}$ les temps de vert pour les r axes routiers qui composent l'intersection. Cette modélisation est illustrée à la Figure 2.3. Ces réglages sont contraints par diverses restrictions et équations [9] : par exemple, les feux d'un axe ne devraient pas passer au vert tant que les feux de l'axe transversal ne sont pas passés au rouge, et inversement. En conséquence, une solution candidate au problème d'optimisation, autrement dit un réglage potentiel des systèmes de feux de signalisation de la ville étudiée, est encodée comme un vecteur de réglages :

$$x = (F_1, F_2, \dots, F_n) \quad (2.2)$$

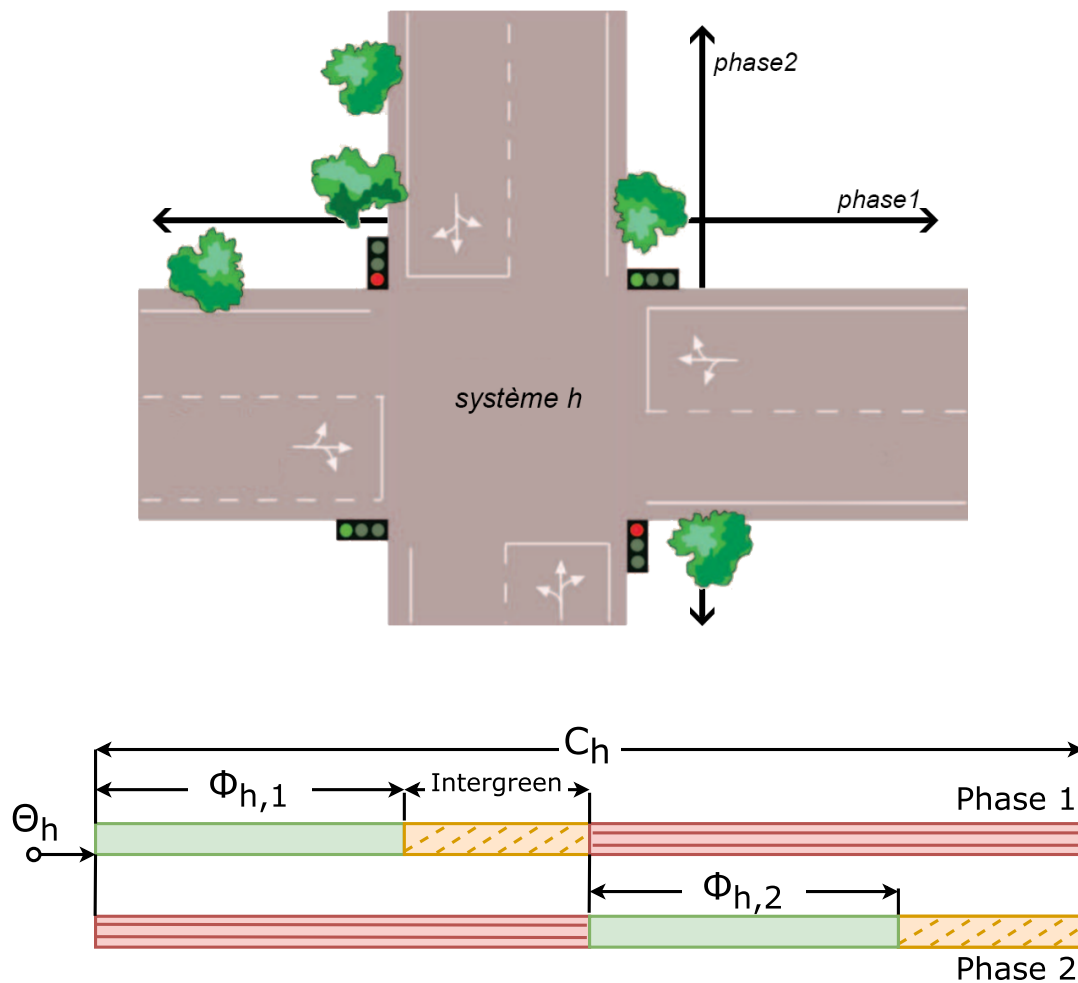


FIGURE 2.3 – Modélisation d'un système de feux de signalisation h , avec deux phases (deux axes routiers) [166].

avec n le nombre total de systèmes de signalisation dans le réseau routier. Les experts urbanistes aspirent finalement à trouver les réglages optimaux x^* qui minimisent le temps de trajet moyen des voyageurs :

$$x^* = \operatorname{argmin} f \quad (2.3)$$

Opérateurs de variation

La modélisation d'un système de feux de signalisation définie par l'Equation 2.1 suggère le recours à trois opérateurs de variation. Ces opérateurs altèrent respectivement les temps de vert, de cycle ou de délai d'un système, en les augmentant ou les réduisant de quelques secondes. En d'autres termes, la variation d'une solution consiste à altérer l'une des trois caractéristiques d'un système F_h qui la compose (une variable de décision). Le choix de l'opérateur de variation à appliquer dépend de taux de modifications fixés au préalable [7]. Dans la suite du manuscrit, ces taux sont désignés par P_{cycle} , P_{vert} et $P_{\text{délai}}$, respectivement pour les opérateurs de variation des temps de cycle, de vert, ou de délai. Par exemple, si ces taux sont fixés à $P_{\text{cycle}} = 0.5$, $P_{\text{vert}} = 0.3$ et $P_{\text{délai}} = 0.2$, alors pour le (ou les) système(s) de feux de signalisation sélectionné(s) pour subir une variation, l'algorithme de recherche choisira d'altérer respectivement leurs temps de cycle, de vert ou de délai selon les probabilités $\frac{1}{2}$, $\frac{3}{10}$ ou $\frac{1}{5}$.

2.5 Optimisation du positionnement des arrêts de bus

Sans perte de généralisation, cette section se concentre sur la seconde stratégie retenue par les experts urbanistes de Calais pour l'amélioration de la mobilité urbaine, à savoir l'optimisation du positionnement des arrêts de bus dans une ville.

L'efficacité d'un système de transport en commun réside dans sa capacité à garantir un niveau de service satisfaisant à ses usagers [8]. La littérature abonde de travaux de recherche traitant de la conception et du déploiement de tels systèmes. Un défi récurrent est de parvenir à ce que les stations des transports en commun soient positionnées judicieusement. Le panorama proposé dans cette section se concentre essentiellement sur l'optimisation du positionnement des arrêts de bus. Leur localisation représente un facteur déterminant qui affecte directement les temps de parcours des passagers [151, 171]. Certains travaux en étudient également les coûts sociaux, les retombées économiques ou les impacts environnementaux [164, 136].

Il existe une grande variété de méthodes pour la recherche du positionnement optimal des arrêts. D'une part, les modèles de couverture, tels que les polygones de Thiessen [171] ou les diagrammes de Voronoi [172] permettent de tenir compte de données complémentaires pendant le processus d'optimisation. Par exemple, ces techniques peuvent servir à caractériser les densités de population des zones urbaines autour d'un arrêt potentiel. D'autre part, la discrétisation du problème d'optimisation s'avère également intéressante. Ibeas *et al.* proposent de diviser l'itinéraire d'une ligne de bus en segments de longueurs égales [76] : chaque lien entre les segments représente un point d'arrêt potentiel pour le bus. Le problème d'optimisation ainsi discrétisé est résolu à l'aide d'un algorithme itératif de recherche de motifs [71]. Dans le même esprit, Furth *et al.* considèrent chaque intersection du réseau routier comme un point d'arrêt potentiel [51]. Ils utilisent alors un algorithme de programmation dynamique pour déterminer les positions optimales des arrêts de bus.

La définition formelle des problèmes d'optimisation liés au positionnement des arrêts de bus diffère fortement selon l'approche envisagée. Par exemple, la représentation du problème selon des polygones de Thiessen établis sur la base des densités de population n'impliquera pas les mêmes méthodes de résolution qu'une représentation discrétisée sur la base du réseau routier, comme illustré par la Figure 2.4. Aussi, dans un souci de généralisation, la définition spécifique des problèmes d'optimisation étudiés dans le contexte des travaux de cette thèse ne sera établie qu'à partir du chapitre 6.

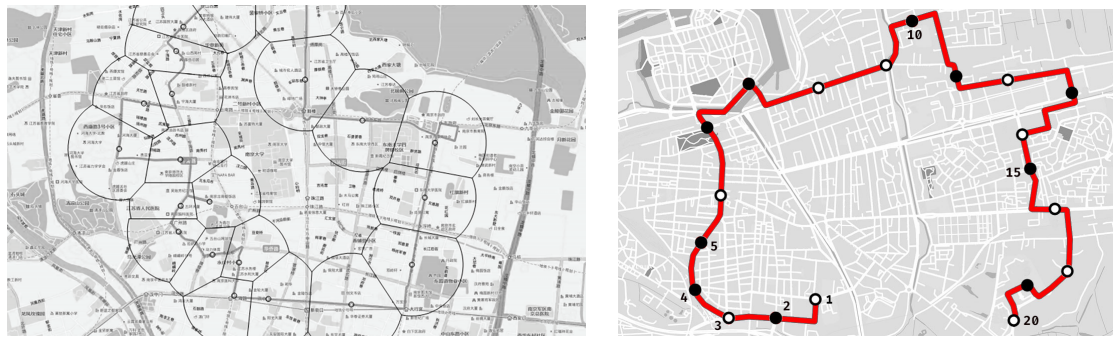


FIGURE 2.4 – A gauche, les arrêts de bus sont positionnés selon des densités de population modélisées par des polygones de Thiessen [171]. A droite, la ligne de bus est discrétisée en segments de tailles égales où chaque point de liaison représente un arrêt potentiel [98].

Résumé

Ce deuxième chapitre présente diverses approches pour la simulation des flux urbains, telles que la simulation macroscopique, microscopique et submicroscopique. Il introduit différentes problématiques associées à l'optimisation de la mobilité urbaine, telles que le réglage des feux de signalisation, la planification des transports en commun, le développement d'infrastructures et de véhicules intelligents, ainsi que la promotion des modes de mobilité douce. En outre, ce chapitre montre l'importance de l'expertise urbaniste pour le réglage et la conception de techniques d'optimisation fiables et robustes, applicables à une variété d'espaces urbains à travers le monde. Enfin, les deux dernières sections se concentrent sur les problématiques retenues par les experts urbanistes de Calais, à savoir l'optimisation du réglage des feux de signalisation et l'optimisation du positionnement des arrêts de bus, en offrant pour chacune de ces sections un panorama des techniques d'optimisation employées dans la littérature pour leur résolution. Finalement, ce deuxième chapitre conclut la première partie de ce mémoire. Celle-ci pose les fondements des problèmes d'optimisation et identifie les travaux état-de-l'art pour des problèmes d'optimisation académiques ou appliqués à la mobilité urbaine. La seconde partie du mémoire poursuit avec les diverses contributions proposées dans le contexte de cette thèse.

Deuxième partie

Contributions

SIALAC : Famille de scénarios de mobilité

Introduction

Ce chapitre propose une méthodologie pour la modélisation des flux urbains. Il expose tout d'abord les motivations qui ont mené à cette approche de modélisation, avant d'introduire le développement d'une variété de cas d'études afin de faire face à des situations de trafic diverses et hétérogènes. Ces travaux ont fait l'objet d'une publication dans une conférence internationale [99].

Le panorama des méthodes présentées aux sections 2.4.1 et 2.5 révèle que la grande majorité des techniques dédiées à l'optimisation de la mobilité urbaine repose sur des études de cas spécifiques. D'une manière générale, ces techniques appliquent un algorithme d'optimisation spécialement conçu ou paramétré pour répondre à une situation particulière. La plupart du temps, les performances de l'algorithme en question sont validées par rapport à des réglages experts ou aux performances du même algorithme configuré selon d'autres paramètres. Aussi, ce panorama met en perspective un besoin de comparaison avec les autres techniques de la littérature. En effet, il semble difficile de concevoir ou de tester un nouvel algorithme d'optimisation de la mobilité urbaine et de déterminer la qualité de ses performances, indépendamment de la spécificité des flux de

voyageurs et des zones urbaines étudiées.

En outre, chacun de ces axes de recherche nécessite une modélisation initiale des flux de circulation dans la ville. Comme introduit à la section 2.1, la modélisation du trafic urbain est fréquemment abordée par une approche microscopique. Les flux de trafic ainsi modélisés sont le résultat des interactions entre les divers plans de mobilité des agents, autrement dit les emplois du temps des citoyens. D'une manière générale, pour des scénarios de mobilité réels, les flux urbains à modéliser peuvent être constitués de milliers, voire de millions de voyageurs. Par conséquent, cela rend la tâche de la collecte d'informations pour chaque voyageur difficile à réaliser : en effet, dans un souci de modélisation microscopique, il serait utopique de collecter individuellement les plans de mobilité de chaque agent composant le flux. Aussi, les experts en modélisation des flux urbains se reposent généralement sur des données agrégées fournies par la ville, telles que des données de recensement de la population, des enquêtes origine-destination ou des données de comptages récoltées sur les principaux axes de déplacement dans la zone urbaine [156]. Néanmoins, l'information est parfois incomplète, obsolète, ou pas disponible pour certains quartiers de l'agglomération étudiée. Aussi, une solution peut consister à générer des populations synthétiques en tirant profit des données agrégées disponibles, et en les extrapolant pour les zones où l'information est insuffisante [1, 116, 14].

3.1 Méthodologie

Dans ce contexte, nous proposons la conception du SIALAC benchmark (*Scenario Investigations of Agents Localization for Algorithm Conception*), applicable à un espace urbain dont le réseau routier est connu. Le benchmark, dont la conception est détaillée à la section 3.2.3, fournit un ensemble de scénarios de mobilité, comme le suggère la première étape de l'organigramme illustré par la Figure 3.1. Ces scénarios visent à modéliser des situations de circulation vraisemblables dans la ville, selon diverses configurations (heures de pointe, jours ouvrables, week-ends, pôles d'attraction). Ainsi, le benchmark fournit un portefeuille de scénarios qui peuvent être utilisés comme données d'entrée dans un simulateur de trafic urbain. Par conséquent, nous espérons remédier aux problématiques éta-

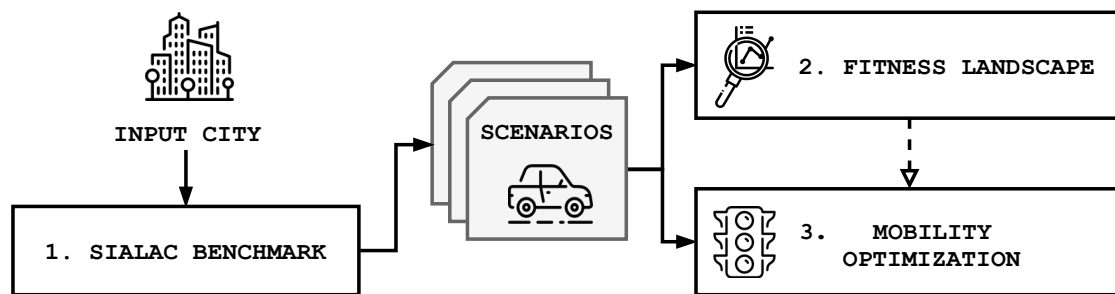


FIGURE 3.1 – Organigramme de la méthodologie.

blies dans l’introduction de ce chapitre, notamment comprendre les propriétés des problèmes de mobilité et comparer les performances de divers algorithmes d’optimisation appliqués à une variété de cas d’étude de la mobilité urbaine.

3.2 SIALAC Benchmark

Sans perte de généralisation, les travaux présentés dans cette section considèrent le réseau routier de la ville de Calais [131]. Nous introduisons le simulateur de flux urbains utilisé, ainsi que le processus de définition des scénarios de mobilité.

3.2.1 Simulateur et état d’équilibre

La modélisation vraisemblable des flux urbains d’une ville est simulée par le système multi-agents *MATSim* [73]. La Figure 3.2 en illustre une exécution. *MATSim* exige comme paramètres d’entrée le modèle d’un réseau routier et les plans de mobilité initiaux d’un ensemble d’agents. Ces plans de mobilité peuvent être interprétés comme l’ensemble des emplois du temps des citoyens : ils définissent leurs domiciles, leurs lieux de travail ou d’activité, ainsi que les créneaux horaires auxquels ils sont susceptibles de se déplacer. Les flux de trafic sont ensuite simulés de manière microscopique. A chaque itération de la simulation, les agents essaient individuellement divers itinéraires pour accomplir leurs impératifs, jusqu’à ce que le simulateur atteigne le principe d’équilibre de Wardrop [38]. L’état d’équilibre est défini comme une condition stable dans

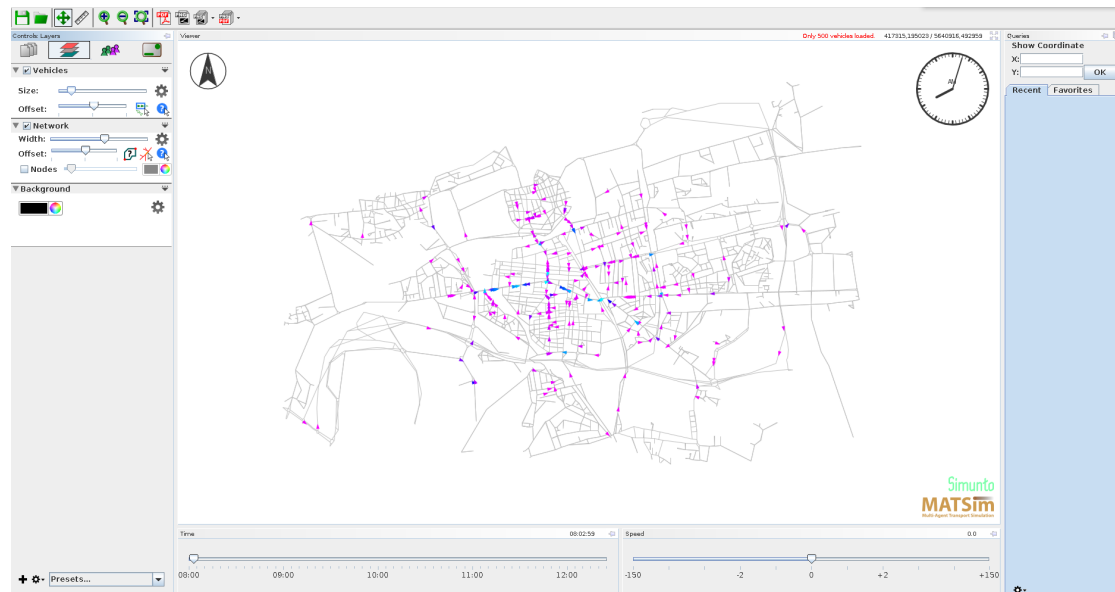


FIGURE 3.2 – Simulation des flux urbains sur le réseau routier de Calais. Chaque triangle symbolise un agent qui se déplace sur le réseau.

laquelle aucun voyageur ne peut améliorer son temps de parcours en changeant unilatéralement d'itinéraire. Cet état nécessite des ressources informatiques substantielles et requiert généralement des temps de calcul de l'ordre de l'heure. Par conséquent, les experts en optimisation précalculent généralement cet état d'équilibre et l'utilisent afin d'initialiser l'algorithme de recherche et d'éviter des temps de calcul superflus [8, 7, 65, 64].

3.2.2 Réseau routier et durée des trajets

Le réseau routier est l'infrastructure sur laquelle les agents peuvent se déplacer. Dans *MATSim*, ce réseau est représenté sous la forme de noeuds et de liens. Du point de vue de la théorie des graphes, cette représentation peut être assimilée à un graphe statique, connecté et dirigé, composé de sommets et d'arêtes. Les noeuds peuvent être interprétés comme des points de décision dans l'espace et sont généralement situés aux jonctions des liens. Les liens représentent des segments de route et définissent des caractéristiques intrinsèques à chaque tronçon, telles que sa longueur, sa capacité de débit, son nombre de voies ou la vitesse maximale qui y est autorisée.

Les scénarios du benchmark considèrent le plan de mobilité d'un agent comme un aller-retour entre un noeud d'origine (*home*) et un noeud d'activité (*activity*). Le noeud d'origine représente le point de départ d'un agent dans la simulation : généralement son domicile, mais parfois une entrée de la ville telle qu'une bretelle d'autoroute. Le noeud d'activité représente la destination d'un agent et peut symboliser un lieu de travail, de loisir ou d'études. Dans un contexte d'optimisation, la fluidité du trafic induite par un scénario de mobilité est définie par le temps de parcours moyen des agents :

$$f(x) = \frac{1}{\#A} \sum_{a \in A} \left(t[M(x), a, \text{home} \rightarrow \text{activity}] + t[M(x), a, \text{activity} \rightarrow \text{home}] \right) \quad (3.1)$$

avec x la représentation d'une solution pour le problème d'optimisation étudié (par exemple, un réglage des feux de signalisation ou un positionnement des arrêts de bus), A l'ensemble des agents simulés, $M(x)$ la simulation de la mobilité des agents selon la configuration x , et t le temps en secondes nécessaire à un agent a dans la simulation M pour accomplir son plan. Ces durées sont calculées à partir des simulations de *MATSim* et sont principalement estimées à partir des caractéristiques intrinsèques des liens susmentionnées. Il convient de préciser que les agents se déplacent à la vitesse maximale autorisée par le lien qu'ils empruntent, tant qu'ils ne sont pas ralentis par d'autres agents ou arrêtés par des feux de circulation.

3.2.3 Définition des scénarios de mobilité

Les scénarios du SIALAC benchmark modélisent la topologie d'une zone urbaine en y localisant des *clusters* : des quartiers résidentiels, des sites d'activité, ou des entrées et sorties de la ville. Les plans de mobilité sont ensuite générés individuellement pour chaque agent. La liste détaillée des scénarios du benchmark est fournie dans la section 3.2.4.

La modélisation de la topologie d'une zone urbaine se base sur les noeuds du réseau routier qui la constituent. Nous supposons que ces noeuds représentent les points d'origine ou de destination de la population à simuler. Dans un premier temps, nous divisons la zone urbaine en une grille de cellules rectangulaires :

plus cette grille est fine, plus la répartition des agents est précise. Chaque cellule de la grille contient ainsi quelques noeuds du réseau routier. Nous pouvons ensuite sélectionner une cellule de la grille pour y établir la source d'un cluster, par exemple pour localiser un quartier résidentiel ou un site d'activité. Chaque cellule avoisinante se voit ainsi attribuer une probabilité d'appartenir à ce cluster. Cette probabilité diminue de manière exponentielle en fonction de la distance par rapport à la cellule source, en suivant une distribution normale. La variance de la distribution normale influe directement sur la répartition spatiale du cluster dans la zone urbaine. Par conséquent, cette modélisation établit une matrice de densité de probabilités qui permet d'affecter aléatoirement des noeuds d'origine et de destination pour les agents de la simulation. Enfin, les agents se voient attribuer une heure de départ ainsi qu'une durée d'activité. Les heures de départ sont fixées de manière aléatoire et uniforme entre 7 heures et 9 heures, ce qui correspond aux heures de pointe du matin à Calais. La durée de l'activité est fixée à quatre heures une fois que l'agent est arrivé à sa destination, puis celui-ci retourne à son noeud d'origine. En suivant l'expertise des urbanistes, la figure 3.3 donne un exemple de quatre clusters *home*, qui peuvent être interprétés comme trois quartiers résidentiels et une entrée de la ville (une bretelle d'autoroute).

3.2.4 Liste des scénarios

La définition des plans de mobilité pour la conception des scénarios du benchmark est basée sur des données de recensement fournies par les experts urbanistes de la mairie de Calais [156, 116, 1]. Ces données comprennent des informations telles que la localisation des quartiers résidentiels, des principaux points d'entrée et de sortie de la ville, des secteurs d'activité attractifs, ainsi que l'emplacement exact des systèmes de feux de signalisation et des arrêts des transports en commun de la ville. Par conséquent, nous appliquons le processus décrit dans la section 3.2.3 pour synthétiser des plans de mobilité vraisemblables. Ces diverses études de cas constituent les scénarios de référence du SIALAC benchmark qui seront utilisés dans la suite des travaux rapportés par ce manuscrit. Les scénarios sont caractérisés par quatre paramètres :

- Le *nombre d'agents* représente la densité de population simulée dans un

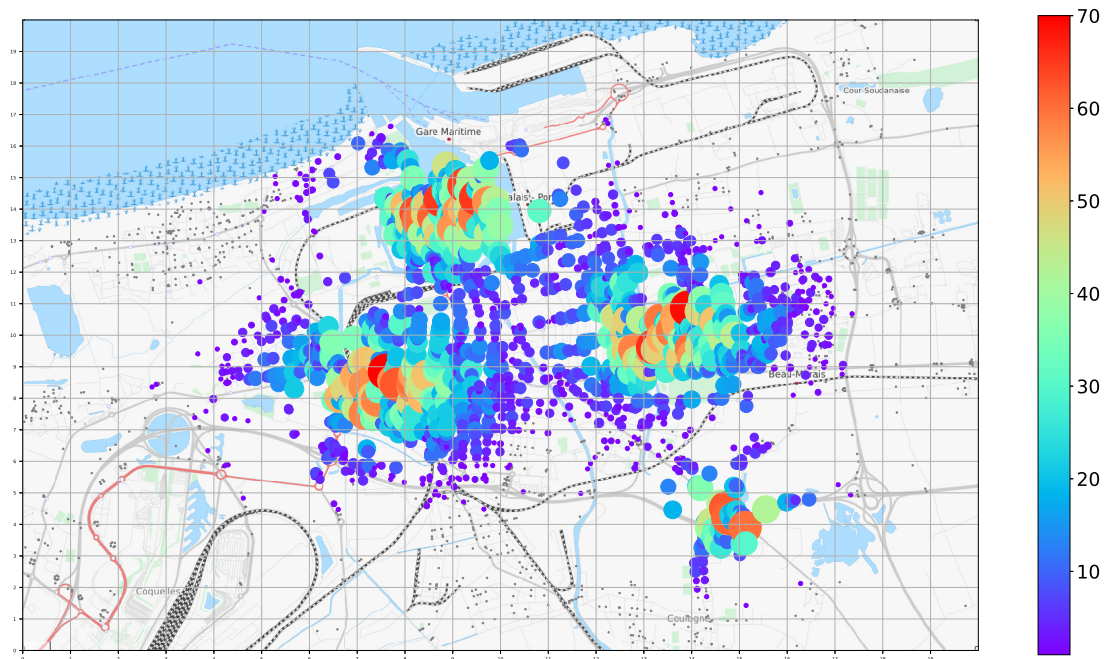


FIGURE 3.3 – Modélisation de quatre clusters *home* dans Calais. La couleur indique le nombre d'agents au départ d'un noeud lors d'une simulation.

scénario. Ce paramètre influe particulièrement sur la congestion du réseau routier. Il varie entre 5000 et 20 000 agents, qui représentent respectivement 6% et 24% de la population de Calais. Ces taux sont proches des déplacements réels des Calaisiens, respectivement les week-ends et les jours ouvrables.

- Les noeuds d'*origine* sont distribués à l'intérieur de clusters spécifiques ou uniformément dans la ville. Cette distribution permet de simuler différentes configurations de l'espace urbain, par exemple pour modéliser des villes monocentriques ou polycentriques. La distribution uniforme est la référence de base pour la répartition de la population dans la ville lorsque la connaissance experte est peu ou pas disponible.
- Les noeuds d'*activité* sont distribués selon le même principe que les noeuds d'*origine*. Cependant, la distribution uniforme des zones d'activité dans l'ensemble de l'espace urbain n'est pas pertinent (les lieux attractifs sont généralement situés dans des zones distinctes). Aussi, les noeuds d'activité ne sont distribués qu'à l'intérieur de clusters spécifiques.

TABLEAU 3.1 – Valeur des paramètres pour les scénarios du SIALAC benchmark.

Nombre d'agents	{5, 10, 15, 20} × 10 ³
Distribution des noeuds d'origine	{1 cluster, 4 clusters, uniforme}
Distribution des noeuds d'activité	{1 cluster, 4 clusters}
Systemes de feux de signalisation	{50%, 75%, 100%}

- Enfin, les *systemes de feux de signalisation* peuvent être activés ou désactivés pendant les simulations. Le ratio des systemes activés varie entre 100%, 75% et 50% du nombre total de systemes sur le reseau routier. Les systemes de feux en peripherie de la ville sont les premiers à être désactivés quand ce parametre est réduit : il ne subsiste ainsi que les feux de signalisation du centre de la ville lorsque seuls 50% d'entre eux sont activés. L'utilisation de ce paramétrage sera profitable à l'analyse des paysages de fitness exposée dans le chapitre 4. La Figure 3.4 illustre l'influence de ce parametre pour la ville de Calais.

Finalement, la combinaison de ces 4 paramètres permet la conception de 72 scénarios de mobilité, comme résumé par le Tableau 3.1. Un exemple de scénario est illustré à la Figure 3.5.

3.3 Valorisation du benchmark

Comme le suggèrent les deuxième et troisième étapes de l'organigramme illustré par la Figure 3.1, nous souhaitons analyser les paysages de fitness induits par les scénarios du SIALAC benchmark, et employer divers algorithmes d'optimisation pour résoudre une variété de problèmes de mobilité tels que l'optimisation du réglage des feux de signalisation ou l'optimisation du positionnement des arrêts de bus. Chacune de ces considérations sera traitée respectivement dans les chapitres 4, 5 et 6 de ce mémoire. Il convient de préciser que l'étude des paysages de fitness n'est pas nécessaire dans un contexte d'optimisation. Cependant, nous montrerons dans la suite de ce manuscrit comment bénéficier d'une telle connaissance automatisée, qui peut parfois suppléer la connaissance experte, afin de concevoir des techniques de résolution efficaces.

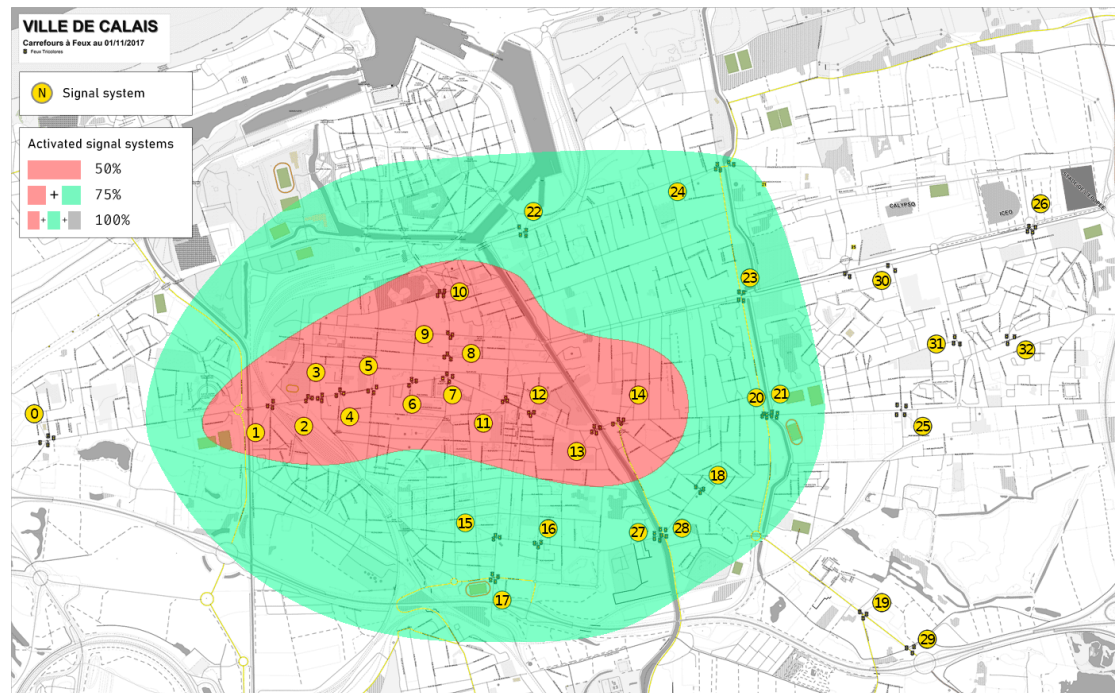


FIGURE 3.4 – Emplacement des 33 systèmes de feux de signalisation dans le réseau routier de Calais. Ces systèmes peuvent être activés ou inactivés dans les simulations, selon les scénarios du benchmark. Pour un ratio fixé à 50%, seuls les systèmes de la zone rouge sont activés (14 systèmes au total). S’ajoutent les systèmes de la zone verte quand 75% des systèmes sont activés (25 systèmes au total). Enfin, tous les systèmes de la ville sont activés lorsque le ratio est fixé à 100% (33 systèmes au total).

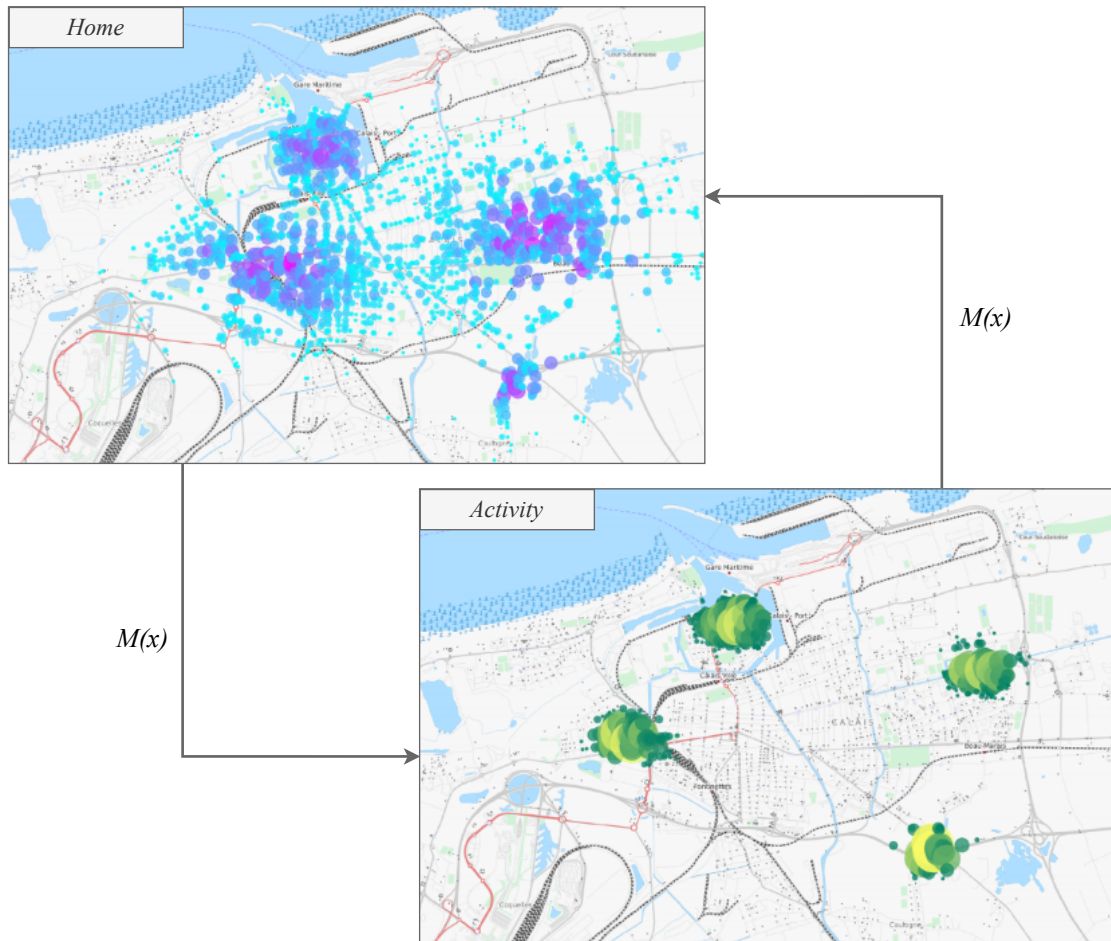


FIGURE 3.5 – Exemple d’un scénario modélisant la distribution de 20 000 agents dans Calais, selon quatre clusters *home* et quatre clusters *activity*. Les déplacements de la population sont simulés par le système multi-agents.

Résumé

Ce troisième chapitre introduit le besoin de comparaison des divers algorithmes d'optimisation présentés par le chapitre précédent, ainsi que les difficultés rencontrées pour la modélisation microscopique des flux urbains lorsque peu d'information est disponible. Ces motivations ont mené au développement du SIALAC benchmark, dont la conception, les paramétrages et les divers scénarios de mobilité sont explicités par la deuxième section de ce chapitre. Finalement, la multitude de cas d'étude proposée par le benchmark permet de mener des travaux approfondis, tels que l'analyse des paysages de fitness ou la conception et la comparaison d'une variété d'algorithmes d'optimisation. Ainsi, ce benchmark permet de tester la robustesse des algorithmes d'optimisation sur une variété de scénarios du benchmark afin de détecter les variables importantes pour le problème d'optimisation du réglage des feux de signalisation.

Modèle d'importance pour l'optimisation du réglage des feux de signalisation

Introduction

Ce chapitre propose d'automatiser certaines connaissances expertes urbanistes, sur la base du benchmark établi dans le chapitre 3. Ces travaux sont essentiellement consacrés à l'optimisation du réglage des feux de signalisation. Pour cela, ce chapitre établit un modèle d'importance pour les variables de décision du problème d'optimisation, en s'appuyant sur de multiples analyses des paysages de fitness, induits par les scénarios du benchmark. Ces travaux ont fait l'objet d'une publication dans une conférence internationale [99] ainsi que dans une revue scientifique [101].

La connaissance experte en aménagement urbain est un atout essentiel pour l'amélioration de la mobilité des citoyens. En pratique, cette expertise permet de guider les techniques d'optimisation vers des solutions robustes qui satisfont au mieux les besoins d'une ville. Toutefois, dans un contexte visant à répondre aux besoins changeants des populations et à des villes en perpétuelle évolution, il ne semble pas concevable de réquérir constamment le jugement et l'expertise des urbanistes pour actualiser les travaux de recherche déjà établis. Par consé-

quent, nous proposons d'automatiser cette connaissance experte, sur la base du benchmark établi dans le chapitre 3.

4.1 Modèle d'importance

Intuitivement, l'importance d'une variable quantifie sa capacité à influencer un critère de qualité. Par exemple, dans le cas d'une régression multilinéaire avec un prédicteur normalisé, cette importance peut être quantifiée par la valeur du coefficient de la régression, ou bien dans le cas d'un classifieur de type *Random Forest*, les variables importantes sont généralement celles qui contribuent le plus à la diminution de l'impureté [6]. Dans le contexte d'un problème d'optimisation boîte noire, nous supposons qu'une variable de décision est importante lorsque son altération influe sur la qualité de la solution qu'elle constitue, au sens de la fonction objectif.

Formellement, soit $op_i(x)$ l'opérateur de mutation qui altère la $i^{\text{ème}}$ variable de décision d'une solution x . Pour un seuil $\tau > 0$, la variable i est importante si :

$$|f(op_i(x)) - f(x)| > \tau \quad (4.1)$$

L'importance des variables peut être estimée au cours de plusieurs marches aléatoires qui modifient une variable de décision à chaque pas, tel qu'introduit par l'Algorithme 3. Comme rapporté à la section 2.4.2, l'ensemble des réglages possibles pour les feux de signalisation est contraint par de multiples restrictions et équations [9]. Autrement dit, bien que l'espace de recherche soit vaste, les diverses restrictions qui s'y appliquent assurent que l'ensemble des marches aléatoires donne un aperçu statistique et cohérent du problème d'optimisation.

Dans cette section, une marche aléatoire de longueur L fournit une séquence de solutions (x_1, x_2, \dots, x_L) , où pour tout $t \in [1, L - 1]$, il existe une variable de décision i telle que :

$$x_{t+1} = op_i(x_t) \quad (4.2)$$

L'importance de la variable i est alors définie par le nombre (ou la proportion) de différences significatives dans l'évaluation de la fonction objectif pour deux

solutions successives de la marche :

$$I_\tau(i) = \#\left\{x_t : t \in [1, L-1] \text{ et } \left| f(\text{op}_i(x_t)) - f(x_t) \right| > \tau \right\} \quad (4.3)$$

Afin de mesurer l'importance des variables sur l'ensemble des marches, la valeur de la fonction objectif est normalisée pour chaque marche aléatoire de manière à obtenir des moyennes et des variances similaires, respectivement égales à 0 et 1. La fonction objectif normalisée est définie par :

$$\hat{f}(x) = \frac{f(x) - E[f(x_t)]}{\sqrt{\text{Var}[f(x_t)]}} \quad (4.4)$$

avec $E[f(x_t)]$ et $\text{Var}[f(x_t)]$ la moyenne et la variance de la fonction objectif au cours de la marche.

4.2 Analyse expérimentale

4.2.1 Protocole expérimental

Nous étudions les 72 scénarios du SIALAC benchmark sur le modèle du réseau routier de Calais, constitué de 2720 noeuds et de 6049 liens. La surface simulée représente une superficie de 35 km². Dans un premier temps, l'état d'équilibre de Wardrop est précalculé pour chaque scénario sans tenir compte de l'activité des feux de signalisation, comme établi à la section 3.2.1. Nous spécifions ensuite les emplacements des 33 intersections signalisées, illustrées par la Figure 4.1. Les états d'équilibre atteints au préalable sont utilisés pour l'initialisation des simulations ultérieures, nécessaires au processus d'optimisation. Les taux de mutation sont fixés à 50%, 30% et 20%, respectivement pour les densités de probabilité P_{cycle} , P_{vert} et $P_{\text{délai}}$ détaillées à la section 2.4.2. L'espace des réglages possibles pour les feux de signalisation est contraint par les spécifications suivantes : les durées des cycles sont comprises dans l'intervalle [40, 140] secondes, les délais sont compris dans l'intervalle [0, 30] secondes et les temps de verts sont compris dans l'intervalle [15, 125] secondes. L'opérateur de variation consiste à

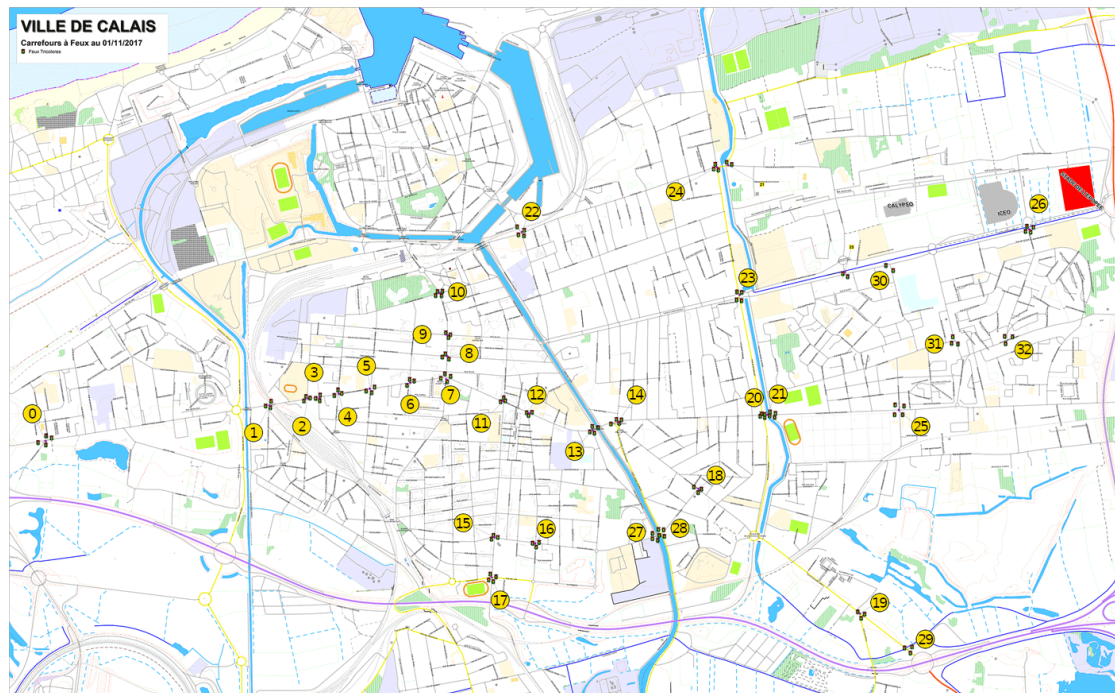


FIGURE 4.1 – Emplacement des 33 systèmes de feux de signalisation dans le réseau routier de Calais.

augmenter ou à réduire de $s \in [1, 10]$ secondes la durée spécifique de l'un de ces réglages selon l'opérateur de variation sélectionné, tout en veillant à respecter les contraintes susmentionnées.

4.2.2 Temps de trajet des voyageurs

Afin d'étudier les structures des paysages de fitness, nous effectuons vingt marches aléatoires de cinquante pas¹ pour chaque scénario du benchmark, conformément à l'Algorithme 3. A chaque pas de la marche, un unique système de feux de signalisation est altéré par un opérateur de mutation. Nous observons ainsi l'évolution de la fonction objectif tout au long de l'exécution de l'algorithme. La Figure 4.2 affiche les durées médianes du trajet des voyageurs, calculées à partir des séquences de solutions fournies par l'ensemble des marches aléatoires, pour chaque scénario du benchmark.

1. Le paramétrage de ces marches est choisi afin d'atteindre un intervalle de confiance à 10% dans le calcul des mesures statistiques de la section 4.2.3.

Une première observation de ces graphiques montre que les temps de trajet augmentent généralement en fonction du nombre d'agents simulés et du nombre de systèmes de feux de signalisation activés. Toutefois, une observation plus approfondie des graphiques conduit aux réflexions suivantes :

- Les temps de trajet les plus courts s'observent généralement pour les scénarios impliquant un ou quatre clusters d'origine et un unique cluster d'activité.
- Lorsque 5000 ou 10 000 agents sont simulés, les temps de trajet les plus longs s'observent pour les scénarios impliquant une répartition uniforme des noeuds d'origine et un unique cluster d'activité. En revanche, pour 15 000 et 20 000 agents, les temps de parcours les plus longs concernent les scénarios impliquant un cluster d'origine et quatre clusters d'activité.
- Il n'y a pas de différence notable pour les temps de trajet médians lorsque 50% ou 75% des feux de signalisation de la ville sont activés.
- Entre 50% (ou 75%) et 100% des feux activés, les temps de trajet restent quasiment identiques lorsque les agents se dirigent vers un unique cluster d'activité, indépendamment de la répartition des noeuds d'origine. En revanche, pour les mêmes ratios d'activation des feux, les temps de parcours augmentent beaucoup lorsque les agents se déplacent vers quatre clusters d'activité, indépendamment de la répartition des noeuds d'origine.

Aussi, ces divers axes de réflexion mènent aux interprétations suivantes. Tout d'abord, ces observations suggèrent que les systèmes de feux du centre de la ville ont une emprise non négligeable sur les temps de parcours des voyageurs et influent particulièrement sur la congestion du réseau routier. Par ailleurs, l'augmentation du nombre de clusters d'activité ou la répartition uniforme des agents dans la ville semblent accroître les situations de croisements entre les voyageurs, et par conséquent les temps de trajet médians. En outre, sauf pour le cas particulier de la distribution uniforme des noeuds d'origine, un unique cluster d'activité semble généralement favoriser la fluidité du trafic dans la ville. Finalement, ces observations révèlent que la répartition de la destination des agents selon quatre clusters d'activité augmente inévitablement la durée du trajet, tandis que la répartition des agents dans quatre clusters d'origine a plutôt tendance à fluidifier le trafic par rapport à la répartition uniforme.

Ces observations peuvent conduire les experts urbanistes à diverses interprétations. Il est évident que le nombre de voyageurs dans la ville a un impact sur la durée des trajets. Néanmoins, certains phénomènes de congestion n'apparaissent pas lorsque le nombre de voyageurs est faible. Par exemple, avec 5000 agents, il n'est pas évident d'établir qu'un scénario de mobilité soit plus propice à la congestion qu'un autre. En outre, la durée médiane du trajet augmente avec le nombre de clusters d'activité, puisque ceux-ci semblent davantage propices aux croisement des voyageurs dans différentes parties de la ville. Enfin, les clusters d'origine semblent avoir moins d'impact sur la durée des trajets par rapport aux clusters d'activité, bien qu'une répartition de la population dans plusieurs clusters d'origine ait tendance à fluidifier les déplacements. Ainsi, du point de vue de l'urbaniste, l'étude de la fluidité du trafic selon une variété de scénarios de mobilité peut fournir une information précieuse afin de comprendre la dynamique d'une ville.

4.2.3 Rugosité des paysages de fitness

L'analyse expérimentale établie par la section précédente montre que la variété des scénarios du benchmark induit des temps de trajet très différents selon les conditions de circulation étudiées. Du point de vue de l'expert urbaniste, la mesure de la fluidité du trafic semble fournir des informations hétérogènes selon les topologies de l'espace urbain. Toutefois, du point de vue de l'expert en optimisation, l'hétérogénéité des réponses de la fonction objectif n'implique pas nécessairement des difficultés différentes pour résoudre les problèmes d'optimisation sous-jacents. A partir des marches aléatoires établies sur les différents scénarios, la rugosité des paysages de fitness peut être estimée en utilisant l'autocorrélation de la fonction objectif ou en évaluant l'entropie maximale des séquences de Vassilev selon plusieurs valeurs de ϵ , échantillonnées dans l'intervalle $[0, 10]$ comme introduit par la section 1.3. Ces mesures sont présentées dans le tableau 4.1.

Etonnamment, la longueur moyenne d'autocorrélation et l'entropie maximale des séquences de Vassilev sont très similaires. Pour les 72 scénarios, les longueurs d'autocorrélation varient entre 7 et 10 et il n'y a pas de différence

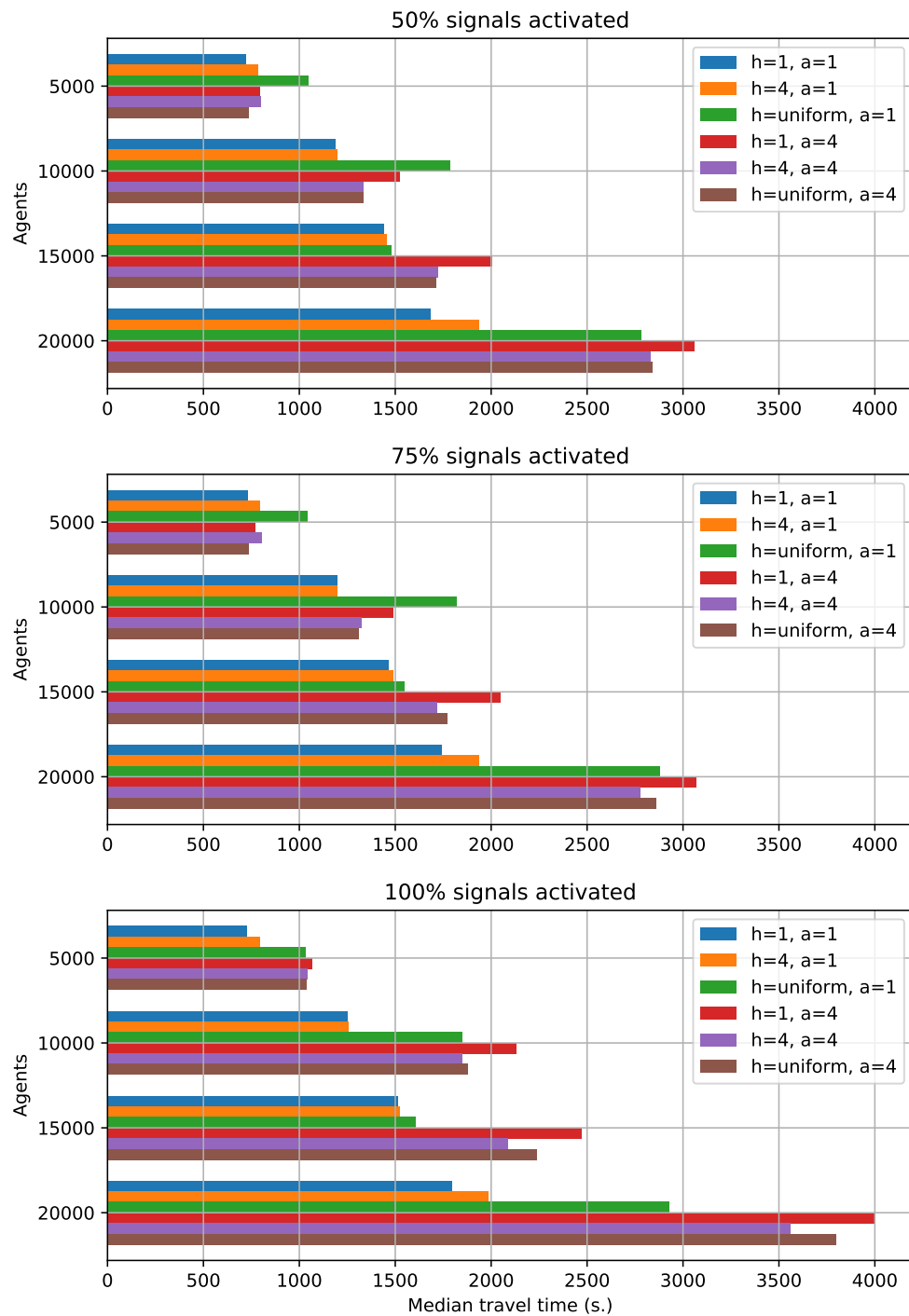


FIGURE 4.2 – Temps de trajet médian (en secondes) pour les scénarios du SIALAC benchmark, estimés sur la séquence des solutions fournies par les marches aléatoires (h : nombre de clusters d'origine, a : nombre de clusters d'activité).

TABLEAU 4.1 – Longueur moyenne d'autocorrélation (ℓ_{ACF}) et entropie maximale des séquences de *Vassilev* (H_{\max}) pour chaque scénario du benchmark, calculées sur la base des séquences de solutions fournies par les marches aléatoires, avec 100% des systèmes de feux de signalisation activés (h : nombre de clusters d'origine, a : nombre de clusters d'activité).

Agents	h	a	ℓ_{ACF}	H_{\max}	Agents	h	a	ℓ_{ACF}	H_{\max}
5000	1	1	8.4	0.81	15 000	1	1	8.7	0.84
		4	7.3	0.82			4	8.8	0.81
	4	1	7.4	0.81		4	1	8.4	0.83
		4	7.2	0.82			4	7.9	0.81
	uniform	1	8.9	0.82		uniform	1	7.7	0.83
		4	8.1	0.79			4	8.3	0.82
10 000	1	1	8.2	0.83	20 000	1	1	7.3	0.84
		4	8.0	0.82			4	8.1	0.85
	4	1	9.2	0.82		4	1	8.8	0.83
		4	8.5	0.82			4	7.9	0.82
	uniform	1	9.2	0.84		uniform	1	8.6	0.84
		4	8.5	0.84			4	7.6	0.83

significative selon un test de Mann-Whitney à 5% de précision. Bien que ces longueurs soient courtes et évoquent des géométries abruptes, cette similitude dans la rugosité des paysages de fitness suggère que la difficulté des problèmes d'optimisation sous-jacents est analogue. En d'autres termes, favoriser une classe d'algorithmes pour résoudre de tels problèmes d'optimisation n'est pas une tâche évidente. Cette observation est renforcée par les mesures d'entropie maximale qui plafonnent à 0,84 pour chaque scénario. En effet, la figure 4.3 illustre l'exécution d'une marche aléatoire pour deux scénarios du benchmark. Les tracés se ressemblent pour chaque scénario : la fonction objectif affiche des bonds abrupts qui reflètent des changements de grandes amplitudes dans la durée du trajet des voyageurs. Ces graphiques et les résultats de l'autocorrélation suggèrent que le modèle autorégressif n'est pas adapté pour mesurer la rugosité des paysages de fitness. En outre, ces perturbations semblent indiquer que certaines variables de décision sont plus importantes que d'autres, puisque leur altération entraîne des changements radicaux dans l'évaluation de la fonction objectif, comme le montre la section suivante.

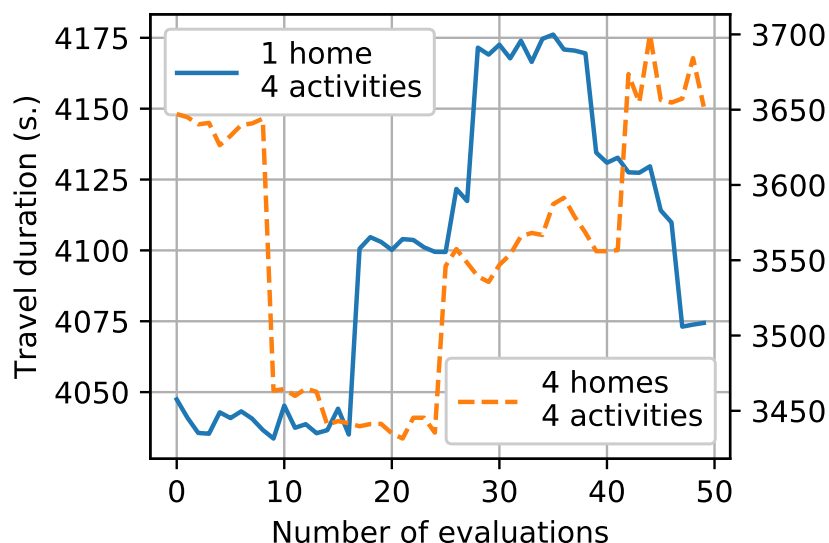


FIGURE 4.3 – Deux marches aléatoires pour deux scénarios du benchmark impliquant 20,000 agents.

4.2.4 Importance des variables

La Figure 4.4 trace la valeur absolue des différences de la fonction objectif :

$$\delta_t = \hat{f}(x_{t+1}) - \hat{f}(x_t) \quad (4.5)$$

calculées à partir des séquences de solutions fournies par les marches aléatoires, tous les scénarios du benchmark confondus. La distribution est unimodale en échelle logarithmique. Cependant, la queue de la distribution est large pour les valeurs élevées de $|\delta_t|$. Cela permet de définir un seuil significatif pour détecter les variables importantes à travers les différents scénarios. Le seuil τ est défini de manière à ce que 10% de la quantité $|\delta_t|$ soit considérée comme une différence significative dans la fonction objectif.

Le Tableau 4.2 recense ainsi la moyenne et l'écart-type du nombre de différences significatives induites par l'altération des variables de décision (autrement dit les systèmes de feux de signalisation), sur l'ensemble des scénarios du benchmark. Nous déduisons de ces résultats que les systèmes de feux ont une importance élevée lorsque cette moyenne est supérieure à 2 (entre autres les

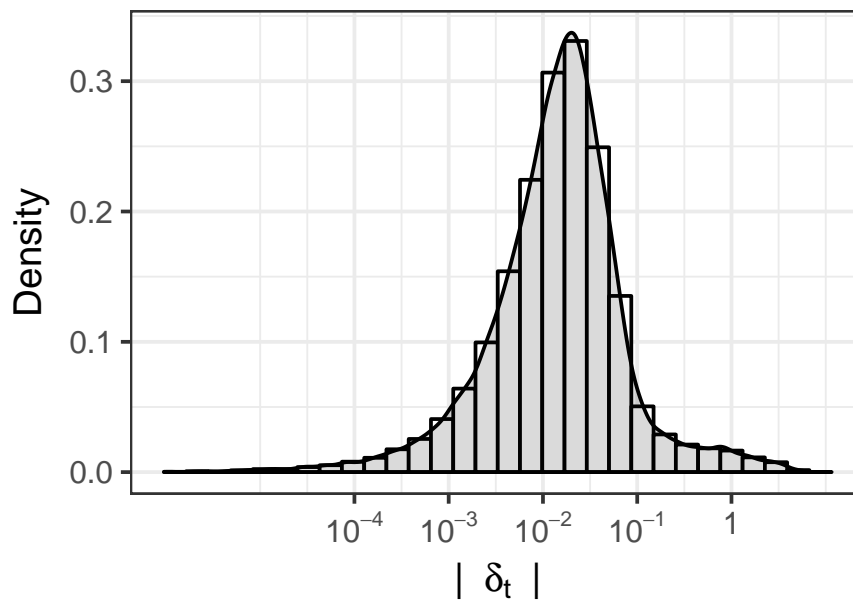


FIGURE 4.4 – Distribution des $|\delta_t|$ avec $\delta_t = \hat{f}(x_{t+1}) - \hat{f}(x_t)$ la différence normalisée des temps de trajet. L'échelle est logarithmique.

systèmes numéro 2, 8, 11, 13, 14 et 29), ou une importance faible lorsqu'elle est inférieure à 1 (par exemple les systèmes numéro 15, 16, 30, ou 31). Cependant, les écarts-types sont élevés et souvent supérieurs aux valeurs moyennes. Ce constat suggère que les systèmes de feux de signalisation importants ne sont pas les mêmes selon les scénarios, comme semble le confirmer la Figure 4.5 lorsque l'importance des variables est calculée individuellement pour chaque scénario. Par ailleurs, nous avons établi dans la section 4.2.2 qu'il n'y avait pas de différence notable dans la durée médiane des trajets pour des scénarios activant 50% ou 75% des feux de signalisation dans la ville : cette observation peut s'expliquer par l'implication de systèmes de faible importance pour ces configurations. En revanche, le passage à des scénarios modélisant la totalité des systèmes de feux de la ville nécessite l'activation des systèmes de signalisation périphériques dont l'importance semble élevée (tels que les systèmes numéro 22 ou 29). Aussi, la durée des trajets augmente inévitablement pour ces scénarios.

La Figure 4.6 illustre l'importance moyenne des variables en fonction de la configuration des scénarios de mobilité. Par exemple, en haut, l'importance des

TABLEAU 4.2 – Moyenne (μ) et écart-type (σ) de l'importance des systèmes de feux de signalisation sur l'ensemble des scénarios du benchmark, avec 100% des systèmes activés.

id.	0	1	2	3	4	5	6	7	8	9	10
μ	0.67	1.96	3.04	2.62	1.75	1.58	1.62	2.67	3.83	2.12	2.46
σ	0.75	1.31	1.84	1.65	1.33	1.04	1.15	1.7	1.4	1.72	1.53
id.	11	12	13	14	15	16	17	18	19	20	21
μ	4.92	1.67	4.38	3.62	0.67	0.79	0.83	1.21	0.75	1.21	1.75
σ	2.77	0.9	2.43	2.53	0.99	0.96	1.07	0.96	0.83	1.	1.56
id.	22	23	24	25	26	27	28	29	30	31	32
μ	2.42	0.75	0.58	0.58	1.88	0.71	0.46	3.58	0.46	0.33	0.38
σ	1.41	0.83	0.91	0.64	2.15	0.84	0.76	3.3	0.64	0.75	0.56

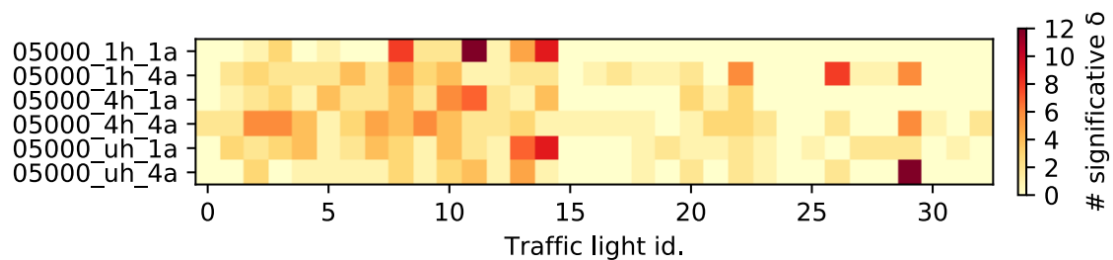


FIGURE 4.5 – Importance des variables sur les scénarios impliquant 5000 agents et différentes configurations des clusters origine et destination.

variables est calculée pour tous les scénarios impliquant 5000 à 20 000 agents. Nous constatons que l'importance d'un système de feux peut varier d'une configuration à l'autre. Par exemple, l'importance de l'intersection numéro 10 vaut environ 4 pour 5000 agents, alors qu'elle chute à moins de 2 pour 15 000 agents. Ceci pourrait impliquer pour les experts urbanistes de modifier le réglage de certains feux de la ville en fonction de la quantité de voyageurs qui s'y déplace (par exemple, lors des heures creuses ou des heures de pointe). Le paramètre du benchmark qui influe le plus sur l'importance des feux de signalisation est le nombre de clusters d'activité. Par exemple, pour l'intersection numéro 29, l'importance passe d'environ 6 pour quatre clusters d'activité, à presque 0 pour un unique cluster. En outre, ce constat semble inciter les experts urbanistes à étudier de près l'ajout d'un nouveau secteur d'activité, voire de préférer le développement de secteurs déjà existants, afin de garantir une fluidité de trafic satisfaisante. Finalement, la rugosité et donc la difficulté des problèmes d'optimisation sous-jacents semblent similaires pour tous les scénarios de mobilité. Les variables de décision qui ont le plus d'impact sur la fonction objectif varient selon les topologies étudiées. Néanmoins, les systèmes de feux ayant une importance élevée pour tous les scénarios du benchmark confondus (dédits du Tableau 4.2) sont clairement identifiables sur ces graphiques : cela est notamment le cas des pics remarquables pour les systèmes numéro 11, 13, 14 ou 29. Finalement, ces résultats obtenus par une approche guidée par les données et la simulation confirment certaines connaissances expertes.

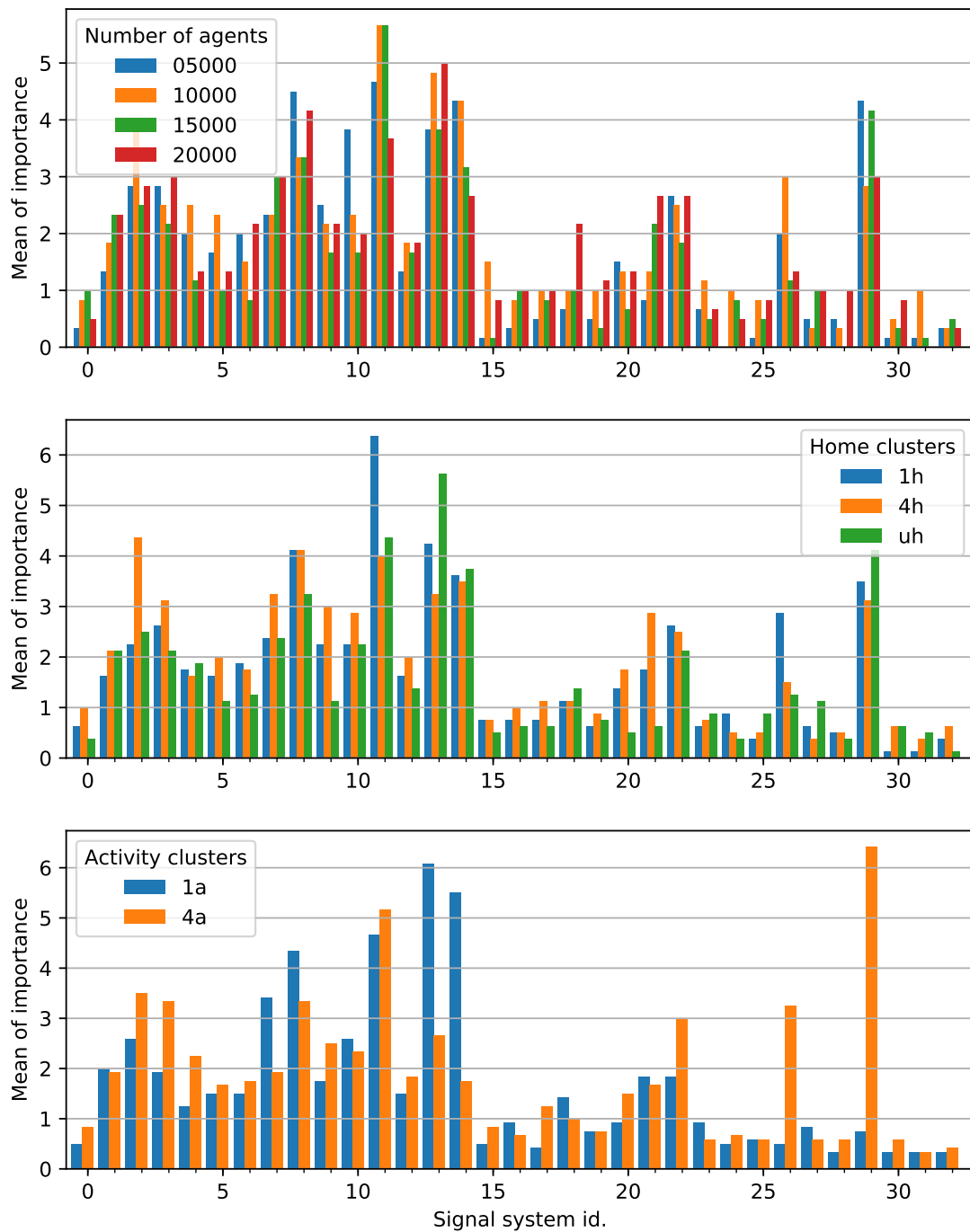


FIGURE 4.6 – Importance moyenne des systèmes de feux de signalisation selon les principaux paramètres du benchmark, avec 100% des systèmes activés.

Résumé

Dans ce quatrième chapitre, nous proposons d'automatiser certaines connaissances expertes urbanistes. Nous analysons les paysages de fitness induits par les scénarios du SIALAC benchmark à l'aide de multiples marches aléatoires, en considérant le temps de parcours moyen des voyageurs comme fonction objectif. La rugosité des paysages, estimée par l'autocorrélation de la fonction objectif et par l'entropie maximale des séquences de Vassilev, est similaire pour tous les scénarios du benchmark. Toutefois, l'importance des variables, estimée selon le nombre de différences significatives engendrées dans la fonction objectif à chaque itération d'une marche, varie fortement d'un scénario à l'autre. En d'autres termes, il ne semble pas exister de scénario de mobilité plus favorable pour l'optimisation qu'un autre. Toutefois, la recherche de réglages robustes semble reposer sur des variables de décision spécifiques, qui diffèrent selon les scénarios étudiés. Ainsi, le chapitre suivant propose de bénéficier de l'analyse des paysages de fitness et de l'interprétation du modèle d'importance induites par les scénarios du SIALAC benchmark afin de concevoir un algorithme d'optimisation adaptatif pour le réglage des feux de signalisation.

Algorithme adaptatif pour l'optimisation du réglage des feux de signalisation

Introduction

Ce chapitre introduit un algorithme d'optimisation adaptatif baptisé *Bandit Descent*, pour l'optimisation du réglage des feux de signalisation dans la ville de Calais. La technique tire parti de l'étude préliminaire des paysages de fitness induits par les scénarios du SIALAC benchmark, afin de guider l'algorithme de recherche vers des solutions de haute qualité, tout en nécessitant un budget calculatoire restreint. Une analyse expérimentale approfondie propose de comparer les performances de diverses techniques d'optimisation sur la base des scénarios de mobilité générés par le benchmark. Enfin, la robustesse de l'algorithme proposé est mise à l'épreuve pour l'optimisation du réglage des feux de signalisation de la ville de Quito, en Equateur. Ces travaux ont fait l'objet d'une publication dans une revue scientifique [101].

L'étude des paysages de fitness induits par les scénarios du SIALAC benchmark permet d'établir un modèle d'importance pour les variables de décision impliquées dans un problème d'optimisation de la mobilité urbaine. Cette connaissance automatisée favorise une meilleure compréhension des espaces urbains, et

peut servir de référence pour le développement de techniques d'optimisation efficaces. Dans un contexte d'optimisation boîte noire, nous ciblons principalement la conception d'algorithmes capables de converger rapidement vers des solutions robustes avec un faible budget en ressources de calcul. En d'autres termes, il est préférable que la technique de recherche requiert un minimum d'évaluations de la simulation coûteuse. Cependant, le panorama des travaux présenté à la section 2.4.1 révèle que la sophistication de la majorité des algorithmes de la littérature pourrait ne pas répondre à ces critères. Bien que les algorithmes évolutionnaires, l'optimisation par essaims de particules ou même la programmation génétique permettent la recherche de solutions robustes, le fondement de ces méthodes nécessite de faire évoluer un ensemble d'individus qui induit un besoin de simulation élevé, et par conséquent un budget calculatoire substantiel. En outre, la typologie combinatoire et la taille de l'espace de recherche associées aux problèmes d'optimisation de la mobilité urbaine ne s'accordent pas bien avec des techniques de résolution classiques telles que l'évolution différentielle – plutôt appropriée pour manipuler des vecteurs de nombres réels – ou la recherche tabou – qui nécessiterait l'exploration d'un voisinage de taille conséquente avant d'aboutir à une convergence satisfaisante.

5.1 *Bandit Descent*

Il a été établi dans le chapitre précédent que certains feux de signalisation influent particulièrement sur la fluidité de la circulation. Seulement, définir la mutation d'une variable de décision selon son importance représente un problème ouvert. En effet, nous pourrions considérer qu'une variable importante ne doit pas être altérée sous l'hypothèse que sa modification entraînerait potentiellement une grande dégradation, ou à l'inverse considérer que la modification d'une variable peu importante ne serait pas utile pour améliorer l'objectif. Aussi, nous proposons un algorithme adaptatif qui tient compte de l'état de la recherche pour modifier ou non les variables selon leur importance. La technique, basée sur une stratégie de bandits manchots et implémentée par l'Algorithme 5, sélectionne automatiquement les feux de signalisation les plus propices à la perturbation du critère de qualité et espère, de façon optimiste, que cette

perturbation soit améliorante au sens de la fonction objectif. Les opérateurs de mutation sont appliqués aux feux sélectionnés selon des densités de probabilités fixes, à savoir 50%, 30% et 20%, respectivement pour les densités de probabilité P_{cycle} pour la variation des temps de cycle, P_{vert} pour la variation des temps de vert et $P_{\text{délai}}$ pour la variation du délai avant le début du cycle (comme détaillé à la section 4.2.1). Cet algorithme s’inspire des travaux de Liu *et al.* [104] introduits dans la section 2.4.2.

Deux éléments essentiels doivent être définis dans le cadre d’une stratégie de bandits : le nombre de bras et la fonction de récompense. Par exemple, une application naïve des travaux de Liu pour cette classe de problème d’optimisation considérerait autant de bras de bandits qu’il y a de systèmes de signalisation dans la ville. Cependant, il convient de préciser que la convergence de la stratégie de bandit ralentit quand le nombre de bras augmente. Afin de réduire ce nombre potentiellement élevé, nous proposons de séparer les feux de signalisation de la ville en plusieurs groupes, désignés par G (*system groups*) dans l’Algorithme 5. Par exemple, du point de vue d’un expert urbaniste, ces groupes peuvent être définis selon la position géographique des systèmes, alors qu’ils peuvent être définis selon leur importance au sens de la fonction objectif du point de vue d’un expert en optimisation. Chaque groupe de feux est alors considéré comme un bras de bandit. Par conséquent, à chaque étape de l’algorithme, une politique UCB sélectionne l’un des groupes puis altère les réglages des feux de signalisation appartenant à celui-ci. Cette étape est assimilée à la ligne 4 de l’Algorithme 5. L’opérateur de mutation est stochastique, comme détaillé dans la section 1.2.2. Contrairement aux travaux rapportés par Liu, nous faisons l’hypothèse que la mutation d’un groupe de feux est pertinente lorsque celle-ci a un impact non négligeable (positif ou négatif) sur la fonction objectif. Aussi, la récompense d’un groupe est calculée par la différence absolue du critère de qualité induite après une mutation :

$$r = |f(x) - f(x')| \quad (5.1)$$

avec f la fonction objectif, x la solution actuelle et x' la solution après l’opération de mutation. La moyenne empirique de cette quantité définit la partie d’exploitation pour chaque groupe dans la formule UCB. De cette façon, l’algorithme est

encouragé à altérer le groupe de feux qui perturbe le plus la fonction objectif, en l'augmentant ou en la diminuant. En effet, nous supposons que les groupes qui modifient fortement la fluidité de la circulation doivent être essentiels pour l'optimisation. C'est pourquoi nous concevons un algorithme qui vise à concentrer les opérateurs de variation sur ces groupes et espérons que la méthode converge rapidement vers des réglages satisfaisants pour la globalité des feux de signalisation de la ville.

Algorithme 5 : *Bandit Descent*

```
Input :  $G$  (system groups)
1  $x \leftarrow$  initialize random solution
2 while stopping criterion not met do
3    $g \leftarrow$  select best group in  $G$  according to bandit strategy
4    $x' \leftarrow$  mutate on variable from  $g$ 
5   if  $f(x') < f(x)$  then
6      $x \leftarrow x'$ 
7   end
8   update rewards
9 end
```

5.2 Analyse expérimentale

Nous recherchons des réglages de haute qualité pour les feux de signalisation, tout en tenant compte du caractère boîte noire et coûteux des simulations impliquées dans le processus d'optimisation. Aussi, nous estimons dans la suite de ce chapitre qu'un algorithme d'optimisation est efficace s'il converge vers un réglage des feux de la ville satisfaisant, tout en requérant un budget calculatoire minimal. En outre, les réglages trouvés à l'issue du processus de recherche constitueront probablement des optima locaux au sens de la fonction objectif. Cependant, dans un contexte d'optimisation boîte noire impliquant des opérations de simulation coûteuses, nous ne cherchons pas nécessairement à trouver le réglage associé à l'optimum global, mais visons plutôt à atteindre un optimum de bonne qualité, même s'il est local, en un minimum d'évaluations de la fonction objectif.

5.2.1 Protocole expérimental

Nos expériences comparent les performances de cinq algorithmes d'optimisation. Le premier algorithme consiste en une méthode *hill-climbing* stochastique (voir l'Algorithme 1), et servira de référence de base pour les comparaisons. Le deuxième algorithme implémente une méthode évolutionnaire élitiste issue de [7], qui manipule une taille de population de 20 individus, génère 20 nouveaux individus à chaque nouvelle génération, et conserve les 10 meilleurs individus (élites) d'une génération à l'autre. Cet algorithme a déjà été employé pour l'optimisation des réglages des feux de signalisation dans d'autres villes du monde, et servira de référence pour la comparaison des performances de l'algorithme adaptatif que nous proposons dans ce chapitre. Enfin, les trois derniers algorithmes sont basés sur des stratégies de bandits qui diffèrent par la façon dont leurs bras et leurs récompenses sont définis (conformément à l'Equation 1.6). Le premier algorithme de bandit tient compte du contexte des travaux de Liu introduits par la section 1.2.2 : ainsi, il y a autant de bras de bandits qu'il y a de systèmes de feux de signalisation dans la ville. La stratégie de *Bandit Descent* vise à réduire ce nombre de bras potentiellement élevé en séparant les feux de signalisation de la ville en différents groupes. Deux moyens sont envisagés pour établir ces groupes. Comme le ferait probablement un urbaniste, la première méthode considère trois groupes géographiques qui réunissent les systèmes de signalisation voisins dans l'espace urbain : par exemple, les systèmes appartenant à un même axe routier ou à un même quartier. La deuxième méthode consiste à séparer les feux de signalisation en trois groupes selon leur importance, déduite des analyses du chapitre 4 : le premier groupe est constitué des systèmes critiques (dont l'importance est supérieure à 2), le deuxième groupe est constitué des systèmes dont l'importance est modérée (estimée entre 1 et 2) et le dernier groupe est constitué des systèmes quasi neutres (dont l'importance est inférieure à 1).

Chacun des cinq algorithmes d'optimisation est exécuté 40 fois sur les scénarios du benchmark impliquant 5000 agents. Une exécution complète représente un budget de 2000 simulations et nécessite environ 15 heures de calcul dans un environnement mono-thread, comme rapporté par la section 5.4.3. En suivant les travaux d'Armas *et al.* [7], nous sélectionnons aléatoirement quatre systèmes

de feux lors de la phase de mutation, afin de faciliter les comparaisons et la discussion dans la section 5.3. Concernant les algorithmes de bandits, la stratégie de sélection détermine tout d’abord un groupe de feux, puis quatre systèmes appartenant au groupe sélectionné sont altérés aléatoirement. Le paramètre d’exploration R de l’Equation 1.6 est empiriquement fixé à 2.

5.2.2 Performances des algorithmes d’optimisation

La Figure 5.1 illustre la minimisation du temps de trajet des voyageurs pour divers scénarios du benchmark, en fonction du budget consacré à l’évaluation de la fonction objectif. Dans ce contexte d’optimisation, une évaluation correspond à une simulation du scénario de mobilité. En outre, les résultats rapportés pour l’algorithme évolutionnaire tiennent compte de la qualité du meilleur individu de la génération. À première vue, chaque algorithme d’optimisation converge vers une qualité de réglage similaire, respectivement au scénario sur lequel ils sont appliqués. Cette convergence est analysée plus en détail par la section 5.4.2. Cependant, il convient de remarquer que les diverses techniques de recherche convergent vers des réglages de haute qualité à des vitesses différentes. Dans un contexte d’optimisation à faible budget, l’algorithme évolutionnaire semble être le moins compétitif : en effet, un tel algorithme nécessite de faire évoluer un ensemble de solutions et requiert davantage d’appels au simulateur pour atteindre des réglages satisfaisants qu’il y a d’individus dans la population. En outre, il semble que le croisement ou le parallélisme implicite de la population ne permette pas de converger plus vite qu’une recherche locale itérative. D’une manière générale, les méthodes *hill-climbing* stochastiques obtiennent des réglages d’une qualité satisfaisante avec un budget calculatoire limité. Cependant, il semble que le regroupement des systèmes de feux de signalisation en fonction de leur importance mène à des réglages de qualité supérieure dès les premières évaluations. Étonnamment, le regroupement géographique des systèmes de feux est généralement aussi efficace qu’une simple méthode *hill-climbing* stochastique, voire moins efficace en ce qui concerne le scénario impliquant une répartition uniforme des clusters d’origine et un unique cluster d’activité (*uh-1a*). Enfin, l’utilisation d’un bras de bandit pour chaque système de feux ralentit consi-

dérablement la vitesse de convergence de l'algorithme d'optimisation, comme l'illustrent les performances désignées par *BD-full* dans les graphiques. Cela peut s'expliquer par le fait que l'algorithme requiert l'exploration d'un trop grand nombre de groupes de feux avant de pouvoir déterminer lequel doit être exploité en priorité.

La vitesse de la convergence de la méthode *Bandit Descent* basée sur l'importance des variables est mise en évidence par le Tableau 5.1. Pour cela, nous supposons que la durée de trajet estimée à l'aide du simulateur et des hypothèses du plan de déplacement est précise à 10%. Nous énumérons ensuite les meilleurs réglages trouvés par les algorithmes d'optimisation pour chaque scénario, toutes exécutions confondues. Ainsi, nous déterminons le nombre moyen d'évaluations nécessaires pour qu'un algorithme d'optimisation converge à 10% près du meilleur réglage connu pour le scénario en question. Le gain de l'algorithme adaptatif par rapport à la méthode *hill-climbing* stochastique ou par rapport à un algorithme évolutionnaire élitiste est confirmé par les tests de Kruskal-Wallis [24]. Ceux-ci démontrent la différence statistique entre les réglages trouvés par ces deux méthodes dès les premières évaluations. Il convient de préciser que les algorithmes d'optimisation s'exécutent dans un temps de calcul équivalent, qui est environ 10 000 fois inférieur au temps de calcul consacré à une simulation, comme le rapporte la section 5.4.3. Toutefois, l'efficacité de la méthode *Bandit Descent* est plus ou moins perceptible selon le scénario auquel elle est appliquée. Par exemple, le gain de la méthode est moins important pour le scénario *4h-1a*, alors qu'il est clairement remarquable dès les premières évaluations pour le scénario *uh-1a*. Ces performances hétérogènes peuvent s'expliquer par le choix de la valeur du paramètre d'exploration R .

5.3 Analyse des méta-paramètres de *Bandit Descent*

Nous revenons sur la stratégie de sélection et le choix des groupes de feux de signalisation pour la méthode *Bandit Descent*, ainsi que sur son paramétrage. Nous approfondissons l'analyse des réglages optimaux présentés dans la section précédente, puis nous menons l'ensemble de la méthodologie des chapitres 3 et 4 pour l'optimisation de la mobilité et le réglage des feux de signalisation d'une

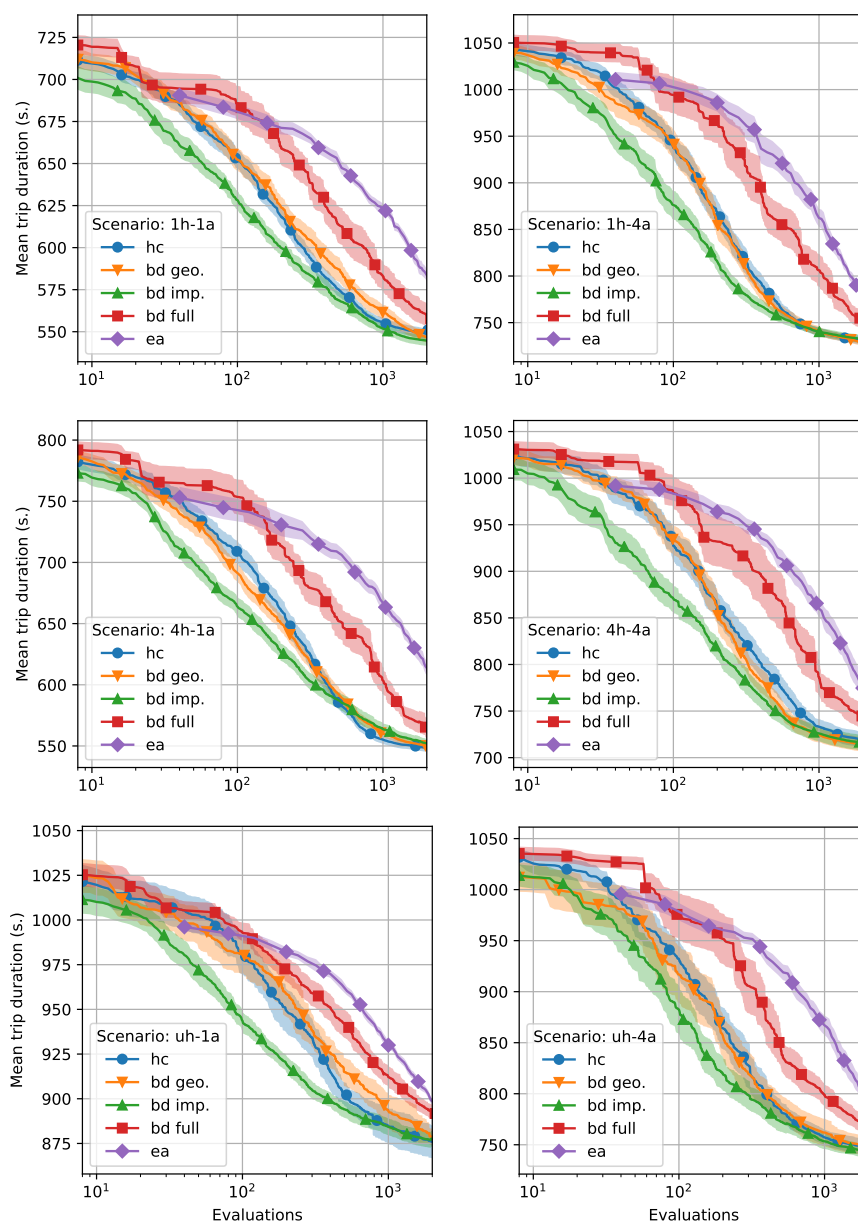


FIGURE 5.1 – Ville de Calais. Minimisation du temps de trajet de 5000 agents, avec 100% des feux de signalisation activés (h : clusters d'origine, a : clusters d'activité, u : répartition uniforme). Cinq algorithmes d'optimisation sont comparés : une méthode *hill-climbing* stochastique (*hc*), un algorithme évolutionnaire élitiste (*ea*) et trois algorithmes *Bandit Descent* (*bd*) paramétrés selon trois groupes géographiques (*geo.*), basés sur l'importance (*imp.*), ou des groupes définis individuellement pour chaque système (*full*). L'échelle est logarithmique en abscisse.

TABLEAU 5.1 – Ville de Calais. Nombre moyen d'évaluations (simulations) nécessaires afin d'atteindre des réglages à 10% près des réglages optimaux, pour une méthode *hill-climbing* stochastique (HC), un algorithme évolutionnaire élitiste (EA) et l'algorithme *Bandit Descent* avec trois groupes de feux basés sur l'importance des variables (BD). Une valeur en gras indique que la performance de la méthode en question est significativement meilleure par rapport à celles des autres méthodes du tableau pour le même scénario, conformément au H-test de Kruskal-Wallis à 5% de précision.

Scénario	Evaluations		
	HC	EA	BD
1h-1a	246	1400	165
1h-4a	362	1680	236
4h-1a	400	2000	315
4h-4a	372	1600	218
uh-1a	146	280	63
uh-4a	308	1240	196

autre ville du monde.

5.3.1 Stratégie de sélection et groupes de feux

Comme établi par la section 5.2.2, les performances de la méthode *Bandit Descent* semblent parfois similaires à celles d'autres techniques d'optimisation, selon le scénario du benchmark étudié. Cette similarité peut s'expliquer par le choix du paramètre d'exploration R fixé arbitrairement pour les stratégies UCB et qui est probablement mal adapté à certains scénarios ou groupes de feux. La Figure 5.2 étudie en détail le scénario *uh-1a* pour 5000 agents et illustre que la valeur du paramètre R peut être affinée de manière à produire des comportements différents de l'algorithme. D'une part, une trop petite valeur conduit l'algorithme à favoriser l'exploitation. Cela conduit à une minimisation rapide du critère de qualité dès les premières évaluations, aux dépens du risque de converger rapidement vers un optimum local de faible qualité. D'autre part, une valeur élevée de R favorise l'exploration. Le comportement de la méthode de bandit s'apparente alors à celui d'une méthode *hill-climbing* stochastique. Ainsi, l'avantage de l'algorithme adaptatif proposé devient plus perceptible lorsque

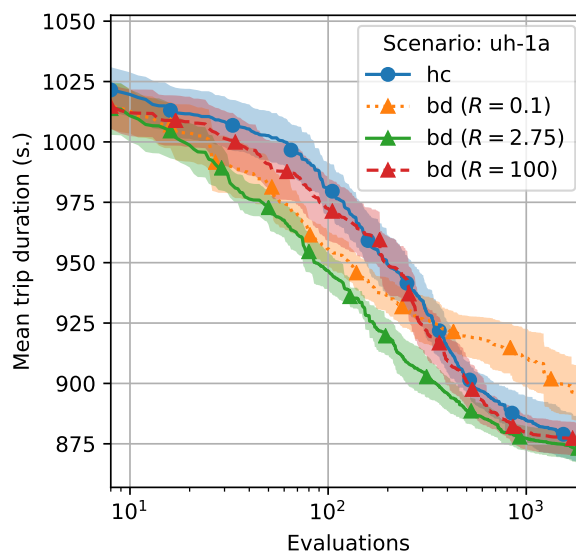


FIGURE 5.2 – Ville de Calais. Performances de la méthode *Bandit Descent* (*bd*) basée sur l'importance des variables, pour plusieurs paramètres d'exploration R , selon un scénario spécifique du benchmark (*uh-1a*). L'optimisation par *hill-climbing* stochastique (*hc*) sert de référence pour la comparaison.

la valeur du paramètre R est choisie judicieusement. Par ailleurs, la méthode *Bandit Descent* proposée utilise une stratégie ucb simple, sans gestion de fenêtre de temps ni stratégie de redémarrage [48], qui pourraient certainement encore améliorer l'efficacité de la technique.

En outre, à paramétrage équivalent pour les stratégies de sélection, nous montrons que les groupes de feux selon un critère géographique conduisent la plupart du temps à des performances plus faibles que les groupes basés sur l'importance des variables. Selon le scénario de mobilité étudié, cette performance est parfois moins bonne que celle d'une simple méthode *hill-climbing* stochastique. Bien que cela puisse sembler contre nature à un expert urbaniste, la détection et le regroupement automatiques des feux de signalisation d'une ville selon leur importance au sens de la fonction objectif semble être plus efficace pour atteindre des réglages de haute qualité. Autrement dit, ce n'est pas tant la stratégie *Bandit Descent* qui conduit à de bonnes performances, mais surtout la façon dont les groupes de feux sont constitués à l'avance.

5.3.2 Robustesse des solutions

En raison des temps de simulation coûteux, du grand nombre de paramètres à évaluer pour chaque algorithme d'optimisation et des nombreux scénarios de mobilité à étudier, il peut être judicieux de tirer parti de la variété de scénarios du benchmark pour accélérer la convergence des algorithmes d'optimisation. En supposant que certains scénarios partagent les mêmes caractéristiques, par exemple le même nombre de clusters d'origine ou d'activité, les réglages optimaux des feux de signalisation trouvés pour un scénario spécifique pourraient être réutilisés pour répondre à l'optimisation de scénarios similaires. En d'autres termes, l'algorithme de recherche pourrait être initialisé à partir des solutions découvertes dans des scénarios de mobilité analogues, plutôt qu'avec des réglages initiaux totalement aléatoires.

Le Tableau 5.2 indique le nombre moyen¹ d'itérations nécessaires à une simple méthode *hill-climbing* stochastique pour atteindre un réglage de haute qualité lorsque l'algorithme est initialisé avec le réglage optimal associé à un autre scénario de mobilité. Pour rappel, il faut environ un millier d'évaluations pour obtenir un réglage de haute qualité lorsque l'algorithme est initialisé à partir de paramètres aléatoires dans la section 5.2.2. La Figure 5.3 illustre un exemple de cette analyse expérimentale : l'initialisation de l'algorithme de recherche à partir de réglages déjà optimisés conduit généralement à une bonne valeur de la fonction objectif dès les premières évaluations. D'ailleurs, il suffit de quelques évaluations supplémentaires pour converger vers un ensemble de réglages d'une qualité encore supérieure. En effet, nous remarquons que pour seulement deux types de scénarios de $1h-4a$ (respectivement $4h-1a$) vers $Xh-1a$ (respectivement $Xh-4a$) initialiser l'algorithme de recherche avec les réglages optimaux d'un autre scénario induit l'algorithme l'optimisation en erreur et freine sa convergence. En revanche, pour tous les autres scénarios, initialiser l'algorithme de recherche avec les réglages optimaux d'un autre scénario permet, d'une manière générale, de découvrir des réglages de haute qualité en moins de 400 itérations. Finalement, il semble préférable de rechercher les réglages optimaux pour un scénario à

1. Ces moyennes sont établies sur la base de 40 exécutions de l'algorithme d'optimisation pour chaque croisement de scénario du Tableau 5.2.

TABLEAU 5.2 – Ville de Calais. Nombre moyen d'évaluations nécessaires pour atteindre un réglage des feux de signalisation de haute qualité avec 5000 agents, en initialisant l'algorithme d'optimisation avec les meilleurs réglages trouvés pour d'autres scénarios de mobilité. Le symbole ∞ est utilisé lorsque la recherche n'a pas pu aboutir à une solution de bonne qualité.

De \ Vers	1h-1a	1h-4a	4h-1a	4h-4a	uh-1a	uh-4a
1h-1a	–	386.4 ± 32.5	355.1 ± 44.8	349.8 ± 35.3	45.3 ± 14.1	379.3 ± 26.1
1h-4a	∞	–	∞	0 ± 0	∞	0 ± 0
4h-1a	25.3 ± 8.0	∞	–	∞	161.1 ± 51.1	∞
4h-4a	157.7 ± 24.8	40.0 ± 16.9	209.3 ± 28.3	–	292.0 ± 69.4	0 ± 0
uh-1a	268.5 ± 76.0	400.6 ± 20.3	404.6 ± 52.6	325.4 ± 19.7	–	338.7 ± 27.9
uh-4a	323.8 ± 66.5	44.3 ± 19.7	326.2 ± 62.9	0 ± 0	401.0 ± 50.0	–

TABLEAU 5.3 – Ville de Calais. Nombre moyen d'évaluations nécessaires pour parvenir à un réglage des feux de signalisation de haute qualité pour les scénarios impliquant 20 000 agents. La recherche est initialisée avec les meilleurs réglages trouvés pour les scénarios impliquant les mêmes topologies de l'espace urbain et 5000 agents. Le symbole ∞ est utilisé lorsque la recherche n'a pas pu aboutir à une solution de bonne qualité.

1h-1a	1h-4a	4h-1a	4h-4a	uh-1a	uh-4a
407.2 ± 110.8	∞	332.1 ± 83.9	287.7 ± 54.3	291.4 ± 51.5	303.1 ± 51.2

partir d'un scénario ayant au moins un même nombre de clusters (d'origine ou de destination).

Pour approfondir la discussion, nous réutilisons les meilleurs réglages des feux de signalisation trouvés dans des scénarios impliquant 5000 agents pour l'optimisation des mêmes topologies de scénarios avec un nombre d'agents augmenté à 20 000. Ces scénarios nécessitent deux fois plus de temps de calcul pour leur simulation. Comme rapporté par le Tableau 5.3, nous montrons une fois de plus qu'une amélioration est possible, dans la plupart des cas, quand nous initialisons le processus d'optimisation à partir des meilleures solutions trouvées lorsqu'un nombre inférieur d'agents est impliqué. Ces solutions sont toujours performantes lorsque davantage d'agents sont simulés, ce qui permet de gagner un temps de calcul précieux afin d'atteindre des réglages de haute qualité.

Par conséquent, cette brève étude montre que les meilleurs réglages des systèmes de signalisation trouvés lors de l'analyse expérimentale de la section 5.2.2

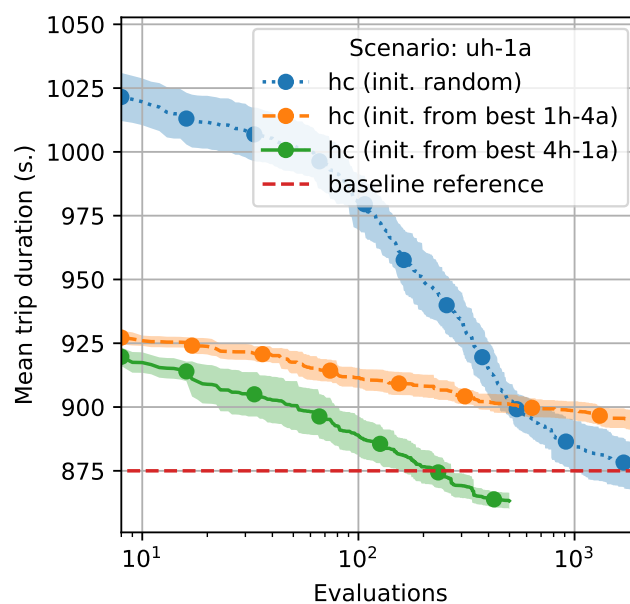


FIGURE 5.3 – Ville de Calais. Minimisation du temps de trajet pour le scénario *uh-1a* et 5000 agents. Les méthodes *hill-climbing* stochastiques sont initialisées aléatoirement ou selon les meilleurs réglages issus d'autres scénarios.

semblent être pertinents lorsqu'ils sont confrontés aux différents scénarios du *SIALAC* benchmark. En outre, cela pourrait suggérer d'envisager des heuristiques multi-objectifs pour aborder plusieurs scénarios du benchmark dans le même processus d'optimisation, afin d'atteindre des réglages des feux de signalisation encore plus robustes. Finalement, la qualité des réglages optimisés trouvés par ces travaux n'est pas évidente à estimer par rapport aux réglages réels appliqués dans la ville de Calais. Ces perspectives de travail sont laissées pour des travaux futurs. De fait, la méthodologie proposée ne vise pas directement cet objectif, mais fournit plutôt aux urbanistes une perspective originale sur leur ville, l'objectif étant d'ouvrir la réflexion en combinant les points de vue des experts en optimisation et en aménagement urbain afin de rechercher les réglages des feux de signalisation les plus efficaces.

Cette première étape de recherche ouvre de nombreuses perspectives allant des applications de mobilité dans le monde réel aux techniques d'optimisation robustes. Le *SIALAC* benchmark est un référentiel ouvert à la communauté

scientifique², qui peut être étendu avec différents réseaux routiers ou des scénarios de mobilité plus précis. Il peut être réutilisé par une large communauté de chercheurs afin d'établir une comparaison plus cohérente entre divers algorithmes d'optimisation et de les améliorer éventuellement. Aussi, nous avons expérimenté notre méthodologie sur la ville de Quito, en Équateur, en suivant et collaborant avec *et al.* [7].

5.4 Application

5.4.1 Application à la ville de Quito, Equateur

Les résultats présentés dans cette section s'appuient sur l'étude de la mobilité dans la ville de Quito, en Equateur [8, 9, 7]. Le réseau routier couvre une zone de 40 km² et comprend 8192 liens et 70 systèmes de feux de signalisation (soit le double du nombre de systèmes impliqués dans le réseau routier de Calais). Nous générons 24 scénarios de mobilité avec le SIALAC benchmark. Ces scénarios impliquent les mêmes paramètres que ceux présentés dans le Tableau 3.1, à la différence que le ratio des systèmes de feux activés est fixé à 100%. Les clusters d'origine et d'activité sont définis en fonction des données de recensement et des destinations attractives de la ville, tels que des campus universitaires, des quartiers d'affaires ou des centres commerciaux.

Le même protocole expérimental que celui décrit par la section 4.2.1 est mené sur ces nouvelles études de cas. Les paysages de fitness et leur robustesse sont estimés à partir d'un ensemble de 20 marches aléatoires de 50 pas sur chaque scénario du benchmark. L'importance des variables est estimée à l'aide de l'Equation 4.3. Comme pour la ville de Calais, certains systèmes de feux de signalisation semblent plus ou moins importants selon le scénario étudié, comme le présentent la Figure 5.4 et le Tableau 5.4. Les feux de signalisation sont considérés comme critiques lorsque leur importance moyenne est supérieure à 3, et comme quasi-neutres lorsqu'elle est inférieure à 2. Une fois de plus, les valeurs des écarts-types sont élevées et souvent proches des valeurs moyennes.

2. Le code source du SIALAC benchmark est disponible en ligne : https://gitlab.com/florianlprt/sialac_benchmark.

TABLEAU 5.4 – Ville de Quito. Moyenne (μ) et écart-type (σ) de l'importance des systèmes de feux de signalisation sur l'ensemble des scénarios du benchmark, avec 100% des systèmes activés.

id.	0	1	2	3	4	5	6	7	8	9
μ	3.29	3.71	2.17	2.92	3.08	4.92	2.25	1.21	1.79	2.
σ	2.34	2.39	1.97	2.48	2.5	2.52	1.88	1.15	1.53	0.82
id.	10	11	12	13	14	15	16	17	18	19
μ	4.17	3.88	2.54	3.42	2.96	2.5	1.5	2.92	3.79	1.33
σ	2.15	2.24	1.76	2.16	2.05	1.78	1.55	1.58	2.5	1.37
id.	20	21	22	23	24	25	26	27	28	29
μ	0.88	1.67	2.67	1.79	2.58	3.54	4.04	1.58	2.25	1.08
σ	1.17	1.49	1.37	1.32	2.58	2.36	2.23	1.29	1.53	1.26
id.	30	31	32	33	34	35	36	37	38	39
μ	0.79	3.25	3.54	4.17	2.83	3.62	4.5	3.58	2.46	1.33
σ	1.15	2.18	2.	2.09	1.7	2.63	1.94	2.33	1.53	1.46
id.	40	41	42	43	44	45	46	47	48	49
μ	1.38	1.42	1.67	1.46	1.96	1.88	2.17	4.04	3.46	3.
σ	1.52	1.47	1.43	1.38	1.49	1.36	1.21	2.42	2.08	2.52
id.	50	51	52	53	54	55	56	57	58	59
μ	3.96	3.29	2.62	2.79	2.29	1.79	1.42	1.46	0.75	1.04
σ	2.35	1.84	1.63	1.98	1.54	1.61	1.41	1.71	1.23	0.98
id.	60	61	62	63	64	65	66	67	68	69
μ	0.58	0.25	1.08	1.21	3.38	1.92	1.67	1.25	1.04	0.92
σ	1.	0.6	1.29	0.91	1.98	1.8	1.95	1.23	1.02	1.08

Cela suggère que les systèmes de signalisation importants ne sont pas les mêmes selon les scénarios de mobilité. Aucune expertise urbaniste de la ville de Quito ne pouvant nous guider, les regroupements de systèmes de feux pour la stratégie *Bandit Descent* sont basés sur cette analyse. Ainsi, la simulation des différents scénarios du SIALAC benchmark et leurs analyses respectives permet à la méthode *Bandit Descent* de découvrir des réglages de haute qualité pour les feux de signalisation pour deux scénarios de mobilité dans la ville de Quito simulant 20 000 agents, comme l'illustre la Figure 5.5. Ces premiers résultats sont calculés selon le même protocole que celui exposé dans la section 5.2.2 pour la ville de Calais.

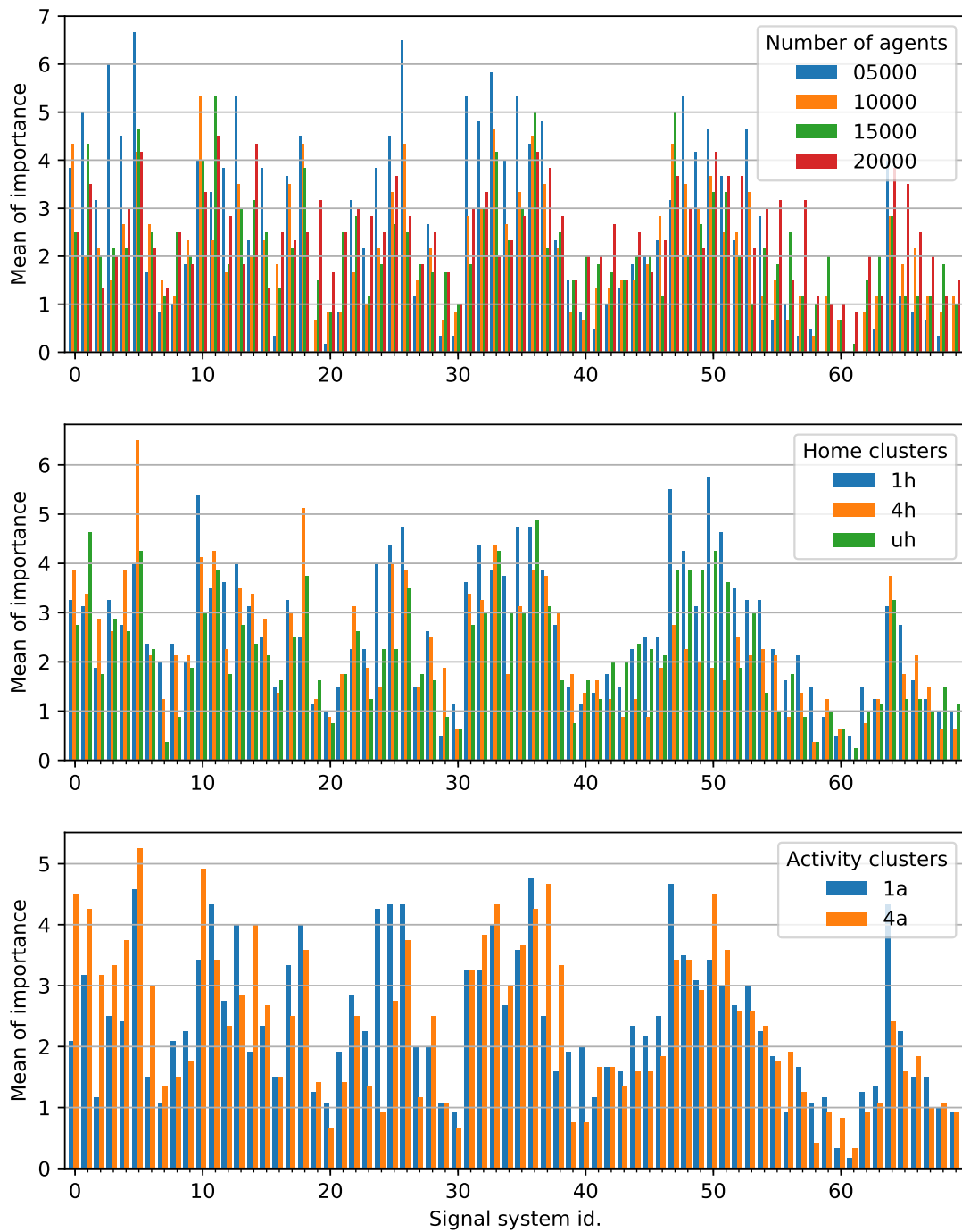


FIGURE 5.4 – Ville de Quito. Importance moyenne des systèmes de feux de signalisation selon les principaux paramètres du benchmark, avec 100% des systèmes activés.

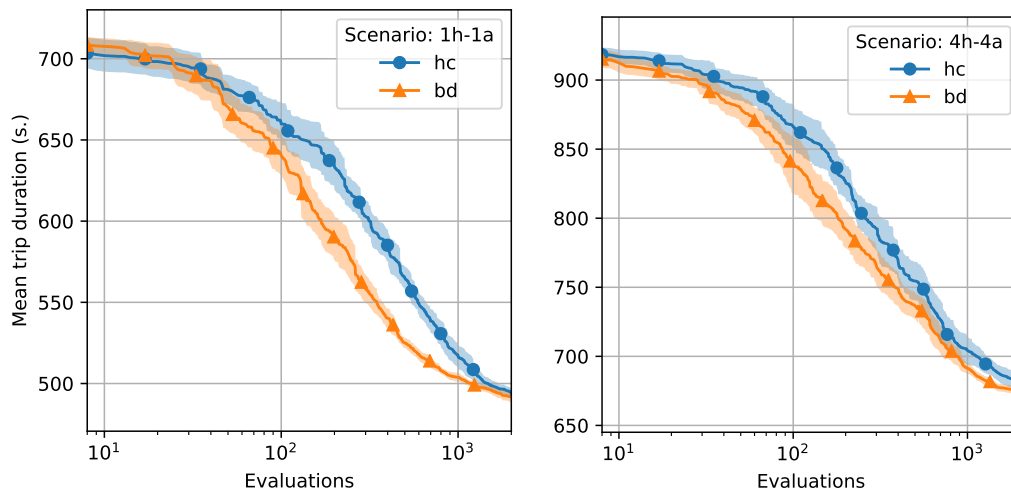


FIGURE 5.5 – Ville de Quito. Minimisation du temps de trajet de 20 000 agents pour deux scénarios de mobilité du SIALAC benchmark. Deux algorithmes d’optimisation sont comparés : une méthode *hill-climbing* stochastique (*hc*) et la méthode *Bandit Descent* avec trois groupes basés sur l’importance des feux de signalisation.

5.4.2 Convergence des algorithmes d’optimisation

Nous prolongeons les exécutions des algorithmes pour l’optimisation du réglage des feux de signalisation d’un scénario du benchmark impliquant 5000 agents. Le protocole expérimental suit les spécifications des sections 4.2.1 et 5.2.1, à la différence que les budgets calculatoires sont doublés à 4000 évaluations de simulateur, de manière à observer la convergence des techniques de recherche. Ces expériences impliquent un temps de calcul d’environ 30 heures pour chaque exécution d’un algorithme d’optimisation dans un environnement *mono-thread*. La Figure 5.6 illustre la minimisation de la durée moyenne des trajets en fonction du nombre d’évaluations, en d’autres termes le nombre d’exécutions de l’outil de simulation. Les résultats sont affichés selon différentes échelles pour faciliter les observations.

L’échelle linéaire montre clairement que chaque algorithme converge vers un ensemble équivalent de réglages optimaux à la fin des 4000 évaluations. Cette affirmation est confirmée par les tests statistiques de Kruskal-Wallis à 5% de précision. Toutefois, l’échelle logarithmique met en évidence les performances

de l'algorithme *Bandit Descent*, basé sur l'importance des feux de signalisation, qui trouve des réglages de haute qualité tout en nécessitant un moindre budget calculatoire, comme évoqué dans la section 5.2.2.

5.4.3 Temps d'exécution des algorithmes d'optimisation

Une itération d'un algorithme d'optimisation comporte deux phases principales. Premièrement, l'algorithme d'optimisation fait évoluer une ou un ensemble de solutions. Deuxièmement, la fonction objectif est évaluée pour ces solutions avec l'outil de simulation. Le Tableau 5.5 et le Tableau 5.6 indiquent les temps moyens d'exécution pour une itération des algorithmes d'optimisation sur chaque scénario du SIALAC benchmark. La comparaison concerne la méthode *hill-climbing* stochastique, l'algorithme évolutionnaire élitiste et la méthode *Bandit Descent*. D'une manière générale, une grande partie sinon la totalité du temps nécessaire à l'exécution d'une itération est consommée par la tâche de simulation, tandis que le temps d'exécution de la phase d'évolution est négligeable (de l'ordre de la milliseconde). En conséquence, l'algorithme évolutionnaire requiert inévitablement beaucoup plus de temps pour compléter une itération, puisqu'il nécessite d'autant plus d'appels au simulateur qu'il y a d'individus à évaluer dans la population. En effet, l'algorithme évolutionnaire élitiste suit les spécifications des travaux d'Armas *et al.* : la taille de la population, le nombre de nouveaux individus à chaque génération et le nombre d'individus d'élite sont fixés respectivement à 20, 20 et 10. Par conséquent, une itération de cet algorithme représente 20×2 évaluations du simulateur, alors qu'une itération de la méthode *hill-climbing* stochastique ou de la méthode *Bandit Descent* n'en nécessite qu'une seule. Ceci explique les temps de calcul par itération très contrastés dans le Tableau 5.5. Il convient de préciser que tous les algorithmes d'optimisation sont comparés avec le même nombre d'évaluations, autrement dit le même budget de temps d'exécution, dans la comparaison proposée à la section 5.2.2.

Finalement, chaque algorithme atteint un ensemble de solutions de qualité équivalente lorsque le budget d'optimisation est étendu, comme présenté dans la section 5.4.2. Ces observations renforcent les arguments en faveur du déve-

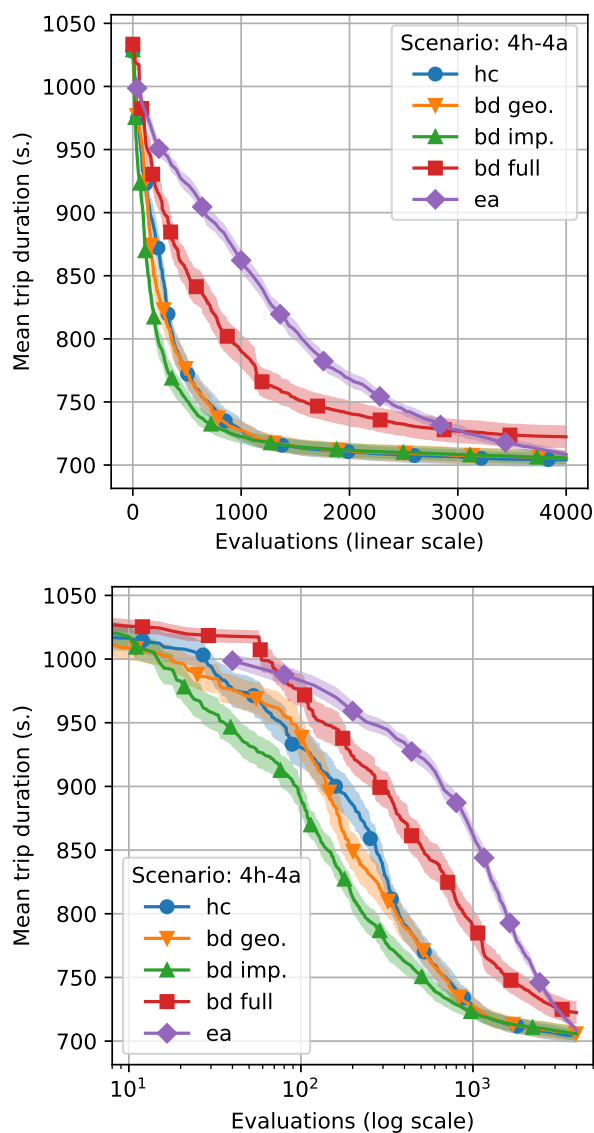


FIGURE 5.6 – Ville de Calais. Minimisation du temps de trajet de 5000 agents, avec 100% des feux de signalisation activés (h : clusters d'origine, a : clusters d'activité). Cinq algorithmes d'optimisation sont comparés : une méthode *hill-climbing* stochastique (hc), un algorithme évolutionnaire élitiste (ea) et trois algorithmes *Bandit Descent* (bd) paramétrés selon trois groupes géographiques ($geo.$), trois groupes basés sur l'importance ($imp.$), ou des groupes définis individuellement pour chaque système ($full$). Le scénario de mobilité est le même pour les deux graphiques. En haut, l'échelle est linéaire en abscisse. En bas, l'échelle est logarithmique en abscisse.

loppement d'algorithmes capables de converger vers des solutions robustes en requérant un minimum d'évaluations de la fonction objectif.

5.4.4 Validation de la méthodologie

Ces performances sont encourageantes pour la généralisation de la méthodologie à diverses villes du monde. Idéalement, pour une ville donnée, un urbaniste peut simuler une variété de scénarios de mobilité à l'aide du SIALAC benchmark, puis analyser ces derniers pour en extraire les systèmes de feux critiques et faciliter la recherche de leurs réglages optimaux. Si les conditions de circulation dans la ville étaient amenées à changer, par exemple suite à l'ouverture de nouvelles zones commerciales ou une délocalisation d'entreprise, les experts de la ville peuvent réitérer la méthodologie proposée pour comprendre la nouvelle dynamique du trafic et reparamétrer les feux de signalisation en conséquence. Le processus d'optimisation hors ligne semble être un choix pertinent lorsqu'il est nécessaire d'améliorer les conditions de circulation avec les infrastructures routières déjà existantes, en attendant la transition vers des technologies plus modernes, telles que les feux de circulation intelligents et actionnés en temps réel par l'état du trafic.

TABLEAU 5.5 – Ville de Calais. Temps moyens d'exécution pour les scénarios de mobilité.

Agents	h	a	Simulation (en s)	Itération (temps cumulé en s)		
				HC	EA	BD
5000	1	1	31.0	31.0	1243.0	31.0
		4	31.7	31.7	1271.2	31.7
	4	1	32.0	32.0	1283.2	32.0
		4	30.9	30.9	1239.1	30.9
	uniforme	1	33.6	33.6	1347.4	33.6
		4	34.0	34.0	1363.4	34.0
10000	1	1	42.4	42.4	1700.2	42.4
		4	46.8	46.8	1876.7	46.8
	4	1	42.5	42.5	1704.6	42.5
		4	45.6	45.6	1828.6	45.6
	uniforme	1	44.7	44.7	1792.5	44.7
		4	48.9	48.9	1960.9	48.9
15000	1	1	50.6	50.6	2029.0	50.6
		4	55.7	55.7	2233.6	55.7
	4	1	50.6	50.6	2029.1	50.6
		4	56.9	56.9	2281.7	56.9
	uniforme	1	50.3	50.3	2017.0	50.3
		4	52.3	52.3	2097.2	52.3
20000	1	1	58.0	58.0	2325.8	58.0
		4	63.8	63.8	2558.4	63.8
	4	1	59.7	59.7	2394.0	59.7
		4	63.0	63.0	2526.3	63.0
	uniforme	1	62.0	62.0	2486.2	62.0
		4	67.8	67.8	2718.8	67.8

TABLEAU 5.6 – Ville de Quito. Temps moyens d'exécution pour les scénarios de mobilité.

Agents	h	a	Simulation (en s)	Itération (temps cumulé en s)		
				HC	EA	BD
5000	1	1	28.0	28.0	1122.8	28.0
		4	29.7	29.7	1191.0	29.7
	4	1	30.1	30.1	1207.0	30.1
		4	28.9	28.9	1158.9	28.9
	uniforme	1	29.6	29.6	1187.0	29.6
		4	29.8	29.8	1195.0	29.8
10000	1	1	33.5	33.5	1343.4	33.5
		4	33.8	33.8	1355.4	33.8
	4	1	34.6	34.6	1387.5	34.6
		4	36.0	36.0	1443.6	36.0
	uniforme	1	34.5	34.5	1383.5	34.5
		4	34.0	34.0	1363.4	34.0
15000	1	1	39.0	39.0	1563.9	39.0
		4	38.6	38.6	1547.9	38.6
	4	1	40.5	40.5	1624.0	40.5
		4	46.1	46.1	1848.6	46.1
	uniforme	1	42.5	42.5	1704.3	42.5
		4	46.2	46.2	1852.6	46.2
20000	1	1	46.6	46.6	1868.7	46.6
		4	48.4	48.4	1940.8	48.4
	4	1	51.4	51.4	2061.1	51.4
		4	50.4	50.4	2021.0	50.4
	uniforme	1	48.4	48.4	1940.8	48.4
		4	49.5	49.5	1985.0	49.5

Résumé

Ce cinquième chapitre propose la conception d'un algorithme d'optimisation adaptatif : le *Bandit Descent*. Cet algorithme utilise une politique de bandit (une technique d'apprentissage par renforcement) et tire parti du modèle d'importance établi au chapitre précédent, afin de concentrer ses efforts calculatoires sur les variables de décision importantes. Ses performances sont comparées à d'autres algorithmes de la littérature ainsi qu'à des configurations imaginées par les experts urbanistes. Une analyse expérimentale approfondie semble montrer que l'algorithme *Bandit Descent* est capable d'atteindre des réglages robustes pour les feux de signalisation, en moins d'itération que nécessaire (et donc de temps de calcul) pour les autres algorithmes comparés. Par ailleurs, la connaissance automatisée semble outrepasser celle des experts urbanistes pour certains scénarios de mobilité. Comme le montrent les résultats sur la ville de Quito en Equateur, ces performances sont encourageantes pour la généralisation de la méthodologie à diverses villes du monde. Idéalement, pour une ville donnée, un urbaniste peut simuler une variété de scénarios de mobilité à l'aide du SIALAC benchmark, puis analyser ces derniers afin d'en extraire les systèmes de feux critiques et faciliter la recherche de leurs réglages optimaux.

Méta-modèles combinatoires pour l'optimisation du positionnement des arrêts de bus

Introduction

Ce chapitre traite de la deuxième stratégie retenue par les experts en aménagement de la ville de Calais pour l'amélioration de la mobilité urbaine, à savoir l'optimisation du positionnement des arrêts de bus. A cette fin, ce chapitre montre que la conception de méta-modèles à partir d'une décomposition en fonctions de Walsh peut être combinée avec des techniques d'optimisation particulièrement efficaces pour des espaces de recherche combinatoires. Dans un premier temps, l'efficacité de la méthode proposée est comparée avec d'autres techniques de la littérature sur des problèmes d'optimisation artificiels. Dans un second temps, ces mêmes modèles de substitution sont employés pour résoudre un problème de mobilité urbaine traitant de l'optimisation du positionnement des arrêts de bus. Ces travaux ont fait l'objet de deux publications dans des conférences internationales [100, 98].

Une grande variété de problèmes d'optimisation repose sur l'évaluation d'une fonction objectif coûteuse en temps de calcul. C'est particulièrement le cas des problèmes d'optimisation de la mobilité urbaine, qui nécessitent une simulation

informatique des flux de trafic pour estimer virtuellement les qualités de diverses configurations pour un espace urbain. Ces tâches de simulation requièrent des durées et des ressources de calcul substantielles et les problèmes d'optimisation associés sont généralement qualifiés de boîtes noires. Une approche pour faire face à cette typologie de problèmes difficiles consiste à élaborer des modèles de substitution, aussi appelés méta-modèles, de la fonction objectif coûteuse afin d'assister le processus d'optimisation. L'optimisation assistée par méta-modèle consiste à optimiser un modèle de substitution rapide à évaluer plutôt que la fonction objectif originale, comme introduit par la section 1.4. La solution optimale au sens du méta-modèle est supposée être prometteuse lorsqu'elle est évaluée par la fonction originale. Cette solution optimale et sa véritable qualité sont ainsi ajoutées à l'échantillon d'apprentissage du modèle de substitution pour l'itération suivante du processus d'optimisation.

6.1 Optimisation assistée par méta-modèle de Walsh

Les travaux rapportés dans ce chapitre introduisent un algorithme d'optimisation assistée par méta-modèle, baptisé *Walsh Surrogate-assisted Optimizer* (*WSaO*). Cet algorithme, spécialement dédié aux espaces de recherche constitués de chaînes binaires, propose d'utiliser une décomposition en fonctions de Walsh pour substituer une fonction objectif coûteuse. En suivant la méthodologie introduite par la section 1.4.1, le modèle de substitution est ainsi optimisé par l'algorithme *Efficient Hill-climbing* (*EH*) [34]. Cette phase d'optimisation correspond alors à la ligne 4 de l'Algorithme 4. La technique *EH* est particulièrement adaptée à l'optimisation des fonctions pseudo-booléennes exprimées sous la forme d'une décomposition en fonctions de Walsh. Sans perte de généralisation, cette phase d'optimisation considère le contexte d'une maximisation, et est implémentée par l'Algorithme 6. Elle consiste à identifier les mouvements améliorants à une distance de Hamming fixée, de manière à concentrer la recherche sur l'altération des variables prometteuses d'une solution. Un score est ainsi attribué à chaque mouvement dans une boule de rayon 1, autrement dit à chaque *bitflip*

possible dans une solution donnée :

$$\delta_m(x) = M(x \oplus m) - M(x) \quad (6.1)$$

avec M le modèle de substitution, x une solution représentée sous la forme d'une chaîne binaire, m un mouvement potentiel et \oplus qui désigne l'opération de *bitflip*. Cette étape peut être assimilée à la ligne 3 de l'Algorithme 6. L'étape suivante consiste à inverser la $m^{\text{ème}}$ variable de x qui maximise l'Equation 6.1, associée aux lignes 5 et 7 de l'Algorithme 6. En fonction de l'interaction des variables dans le méta-modèle, il est démontré que seuls certains scores sont affectés par cette inversion [34]. Par conséquent, l'actualisation des scores après chaque mouvement ne concerne qu'un sous-ensemble de scores spécifiques, calculés à la ligne 8 de l'Algorithme 6. Cette actualisation s'effectue en complexité constante en fonction de la dimension du problème d'optimisation, et en complexité linéaire selon le nombre d'interactions entre les variables dans les sous-fonctions de la décomposition. Ce processus est itéré jusqu'à ce que le budget de calcul soit épuisé ou qu'il n'existe plus de mouvement améliorant δ_i^* , autrement dit que tous les δ_m soient inférieurs ou égaux à zéro. Finalement, si la solution optimale proposée à l'issue de la recherche est déjà dans l'échantillon d'apprentissage du modèle de substitution, alors le nouveau point à ajouter est échantillonné de manière aléatoire afin d'assurer une diversité de solutions dans l'échantillon. Cette étape est associée aux lignes 12 et 13 de l'Algorithme 6. Finalement, une implémentation complète de la méthode *Walsh Surrogate-assisted Optimizer* est proposée par l'Algorithme 7.

6.2 Analyse expérimentale sur des problèmes artificiels

Nous étudions la qualité des méta-modèles basés sur les fonctions de Walsh ainsi que les performances de l'algorithme d'optimisation *WSaO* introduit par la section précédente. L'analyse expérimentale est menée sur deux problèmes d'optimisation artificiels, introduits par la section 1.5. La prochaine section pose

Algorithme 6 : Sélection d'une solution à partir d'un méta-modèle basé sur une décomposition en fonctions de Walsh.

```

1 while restarts budget is not spent do
2    $x \leftarrow$  Random solution
3    $\delta \leftarrow$  Compute scores for each move
4   while iterations budget is not spent do
5      $i^* \leftarrow$  Find  $\arg \max \delta_i$ 
6     if  $\delta_{i^*} > 0$  then
7        $x \leftarrow$  Flip  $i^*$ -th bit in  $x$ 
8        $\delta \leftarrow$  Update affected scores
9     else
10      break
11    $x^* \leftarrow$  Update best solution if  $x$  is an improving solution
12 while  $x^*$  is already known do
13    $x^* \leftarrow$  Sample random solution

```

Algorithme 7 : *Walsh Surrogate-assisted Optimizer (WSaO).*

```

1  $f \leftarrow$  Original (expensive) function
2  $o \leftarrow$  Maximum order for the Walsh functions in the decomposition
3  $S \leftarrow$  Initial sample  $\{(x, f(x)), \dots\}$ 
4 while computational budget is not spent do
5    $W \leftarrow$  Build Walsh decomposition at maximum order  $o$ 
   (see Equation 1.17)
6    $M \leftarrow$  Learn surrogate model by estimating coefficients of  $W$  from  $S$ 
   (sparse linear regression technique is employed)
7    $x \leftarrow$  Optimize  $M$  with Algorithm 6
8    $S \leftarrow S \cup \{(x, f(x))\}$ 

```

le contexte des expérimentations en précisant notamment les valeurs prises par les différents paramètres des algorithmes étudiés.

6.2.1 Protocole expérimental

Nous considérons les problèmes d'optimisation combinatoire $UBQP$ et les paysages NK , tels que définis par les sections 1.5.3 et 1.5.4. Nous étudions ces problèmes pour diverses dimensions $n \in \{10, 25, 50, 100\}$, de petites à bien plus grandes que celles traitées dans les travaux de la littérature [13, 170]. Pour les problèmes $UBQP$, nous suivons les recommandations de Liefoghe *et al.* [103] et fixons la densité de zéros dans la matrice Q à 90%. Pour les paysages NK , nous faisons varier $k \in \{1, 2\}$ qui induit une interaction quadratique entre les variables lorsque $k = 1$, ou cubique lorsque $k = 2$. Pour chaque combinaison de paramètres (4 pour $UBQP$, $4 \times 2 = 8$ pour les paysages NK), nous générons cinq instances du problème d'optimisation correspondant. Le paramètre de régularisation pour la méthode de régression *Lasso* est choisi empiriquement à partir des valeurs prises par les fonctions objectif originales sur la base d'un échantillon de solutions aléatoires. Le paramètre de régularisation α est ainsi fixé à 10^{-1} et 10^{-5} , respectivement pour les problèmes $UBQP$ et les paysages NK . Finalement, nous testons la précision des modèles de substitution par rapport à un ensemble de 1000 solutions générées aléatoirement. Les techniques de régression sont exécutées 20 fois pour chaque instance de problème pour mesurer la précision moyenne des méta-modèles. Afin de mesurer les performances moyennes des techniques d'optimisation assistée par les modèles de substitution, les algorithmes sont redémarrés 100 fois, ce qui définit le budget des lignes 1 et 4 de l'Algorithme 6. L'ordre maximal des fonctions de Walsh et des fonctions multilinéaires est fixé à 2 pour les problèmes $UBQP$ et NK lorsque $k = 1$, et l'ordre est fixé à 3 pour les problèmes NK avec $k = 2$. Les algorithmes et les expériences sont entièrement implémentés en Python¹ et utilisent des bibliothèques d'apprentissage et d'optimisation standard [127].

1. Le code source des méthodes utilisées est disponible en ligne : <https://gitlab.com/florianlprt/wsao>.

6.2.2 Précision des méta-modèles

Nous expérimentons la précision de trois modèles de substitution : la méthode de Krigeage basée sur des processus gaussiens, la base de fonctions multilinéaires, et la base de fonctions de Walsh introduite par la section 1.4. Aucun algorithme d'optimisation n'est utilisé pour ces premières analyses : la précision des méta-modèles est mesurée sur la base de solutions échantillonnées aléatoirement. Les solutions sélectionnées pour l'apprentissage du modèle sont échantillonnées uniformément dans $\{0, 1\}^n$. Les Figures 6.1, 6.2 et 6.3 comparent la valeur absolue de l'erreur moyenne commise par les modèles de substitution sur l'ensemble de validation, en fonction du nombre d'échantillons aléatoires dédiés à leur apprentissage. Pour les deux problèmes d'optimisation étudiés, chaque modèle de substitution converge vers une erreur nulle lorsque la taille de l'échantillon augmente. Toutefois, cette convergence semble s'effectuer à des vitesses différentes selon les méthodes. Dans l'ensemble, la méthode de Krigeage apparaît comme la plus lente des trois. Pour les problèmes $UBQP$ présentés dans la Figure 6.1, la précision des méta-modèles multilinéaires et des méta-modèles de Walsh évolue d'une manière similaire pour des dimensions modérées (inférieures ou égales à 50). Néanmoins, lorsque les problèmes d'optimisation considèrent 100 dimensions, les méta-modèles de Walsh atteignent une erreur strictement nulle en nécessitant un budget d'apprentissage deux fois moins important que celui consacré à l'apprentissage des modèles multilinéaires. D'autre part, les Figures 6.2 et 6.3 montrent des différences significatives dans la vitesse de convergence pour ces mêmes méthodes appliquées aux paysages NK . La taille de l'échantillon nécessaire à l'obtention d'une erreur strictement nulle est sensiblement plus petite pour les méta-modèles de Walsh sur des problèmes en faibles dimensionnalités. D'ailleurs, ces modèles convergent environ deux fois plus rapidement que les méta-modèles multilinéaires lorsque les paysages NK considèrent 50 ou 100 dimensions. Ce contraste est encore plus marqué pour les paysages qui impliquent une interaction cubique entre les variables. Toutefois, il convient de préciser que la taille de l'échantillon nécessaire pour que les méta-modèles de Walsh atteignent une erreur nulle augmente rapidement avec le nombre d'interactions entre les variables, comme l'illustre la Figure 6.4.

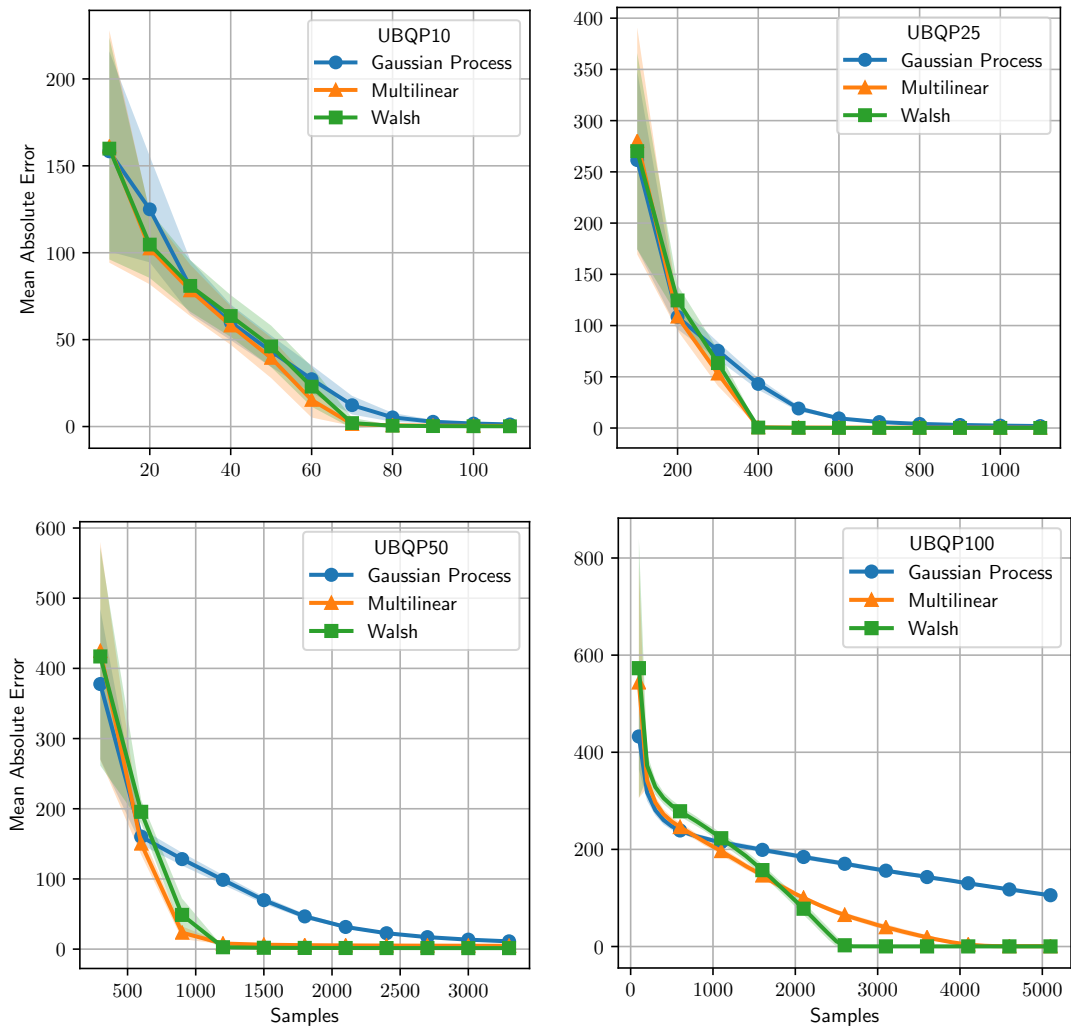


FIGURE 6.1 – Moyenne de l'erreur absolue sur un échantillon de test et intervalles de confiance des méta-modèles sur des problèmes UBQP de diverses dimensions, en fonction du nombre d'échantillons aléatoires dédiés à leur apprentissage.

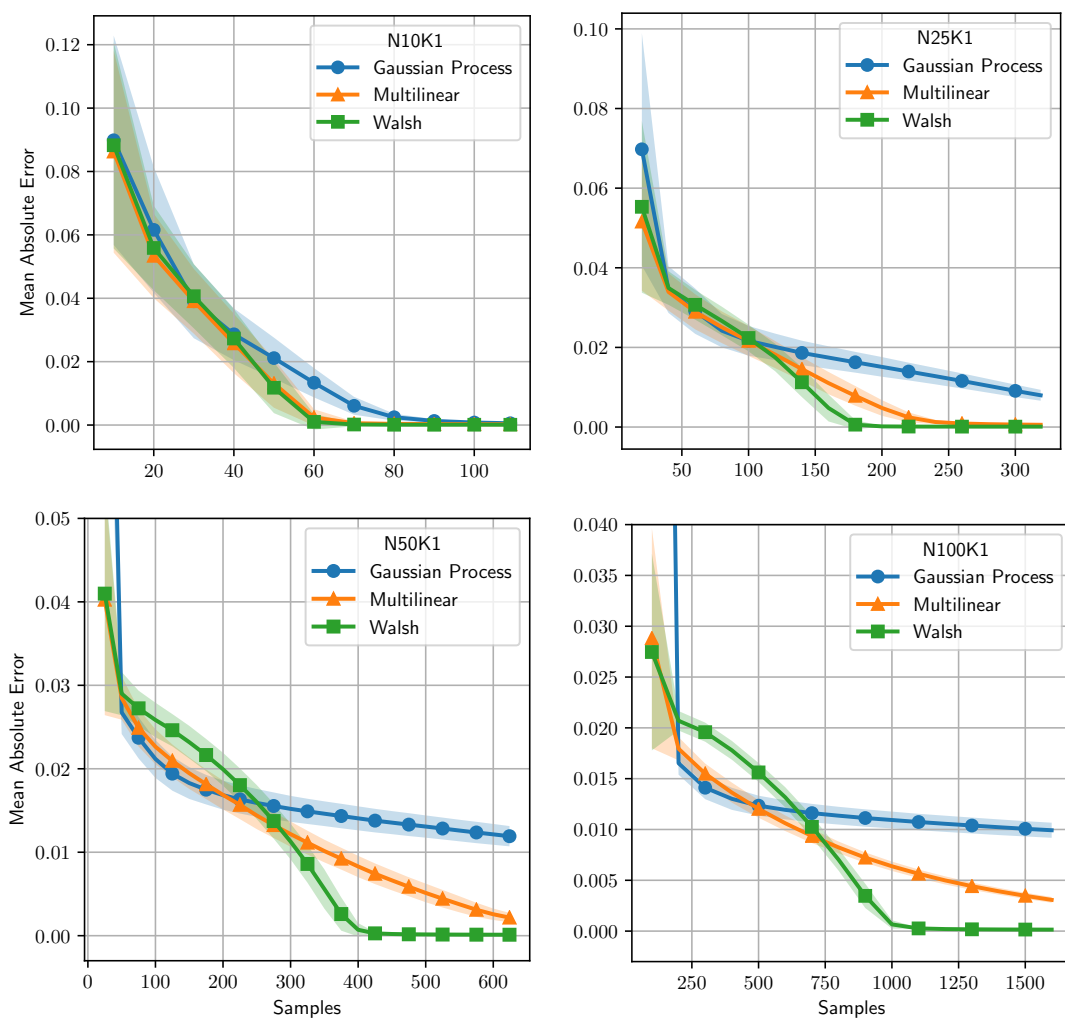


FIGURE 6.2 – Moyenne de l’erreur absolue sur un échantillon de test et intervalles de confiance des méta-modèles sur des paysages n_k de diverses dimensions, en fonction du nombre d’échantillons aléatoires dédiés à leur apprentissage. Le paramètre k est fixé à 1.

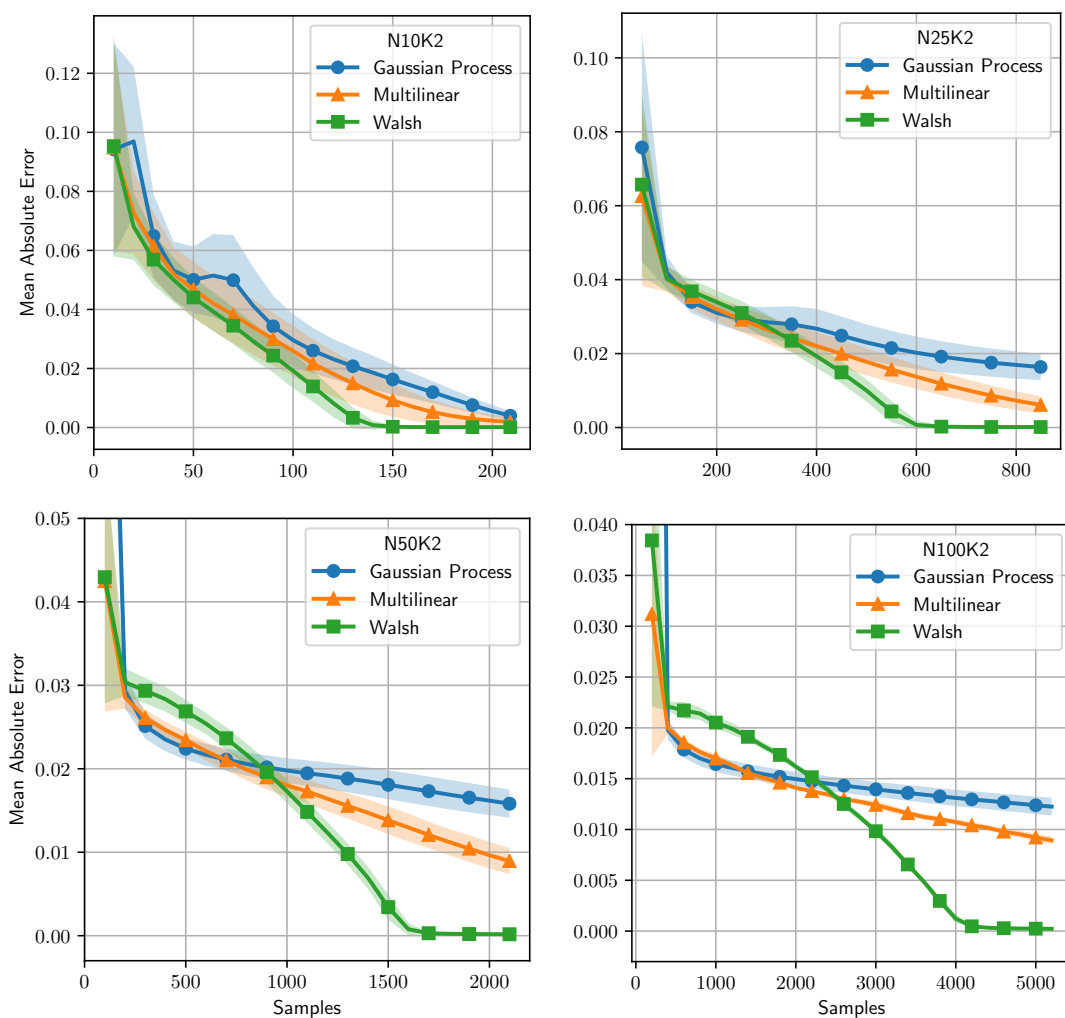


FIGURE 6.3 – Moyenne de l’erreur absolue sur un échantillon de test et intervalles de confiance des méta-modèles sur des paysages \mathcal{N}_k de diverses dimensionnalités, en fonction du nombre d’échantillons aléatoires dédiés à leur apprentissage. Le paramètre k est fixé à 2.

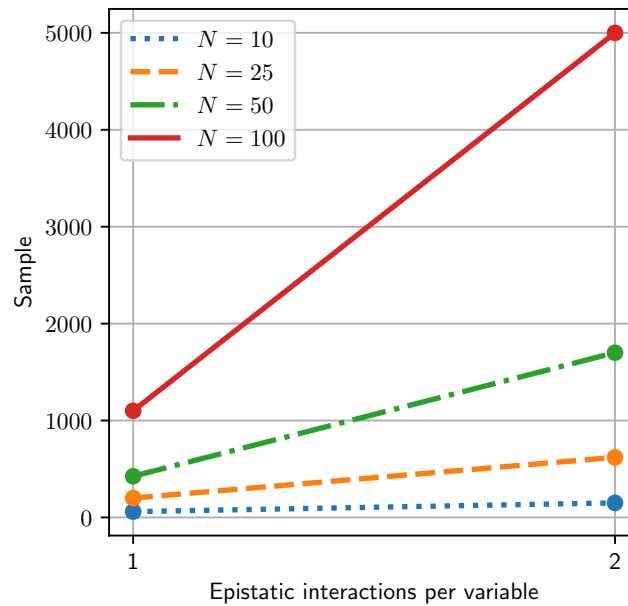


FIGURE 6.4 – Taille d'échantillon requise afin d'atteindre une erreur nulle pour les méta-modèles de Walsh, selon la taille du problème et le nombre d'interactions entre les variables.

6.2.3 Performances en optimisation

Nous déterminons les performances de l'algorithme d'optimisation assistée par un méta-modèle basé sur une décomposition de fonctions de Walsh (*WSaO*), comme implémenté par l'Algorithme 7. Nous comparons ces performances avec celles des algorithmes de la littérature introduits par la section 1.4 : l'*Efficient Global Optimizer* (*EGO*) couplé à une méthode de Krigeage [170] et l'algorithme *BOCS* [13] pour l'optimisation de méta-modèles basés sur une décomposition de fonctions multilinéaires. Nous comparons également les performances de l'Algorithme 6 lorsqu'il est couplé à une décomposition de fonctions multilinéaires apprise avec la technique de régression *Lasso*. Contrairement à l'analyse de précision de la section précédente, la solution ajoutée à l'échantillon d'apprentissage à chaque itération de l'algorithme est maintenant la solution qui maximise (ou minimise selon le problème) le modèle de substitution. En outre, la résolution de problèmes d'optimisation en grandes dimensions impliquant des interactions cubiques entre les variables nécessite des ressources de calcul substantielles.

Aussi, les résultats présentés dans la suite de cette section ne tiennent compte que des problèmes UBQP et des paysages NK où k est fixé à 1, limitant ainsi l'interaction entre les variables à un ordre quadratique. L'ordre maximal des fonctions de Walsh et des fonctions multilinéaires est ainsi fixé à 2. L'analyse des performances en optimisation pour des problèmes dont l'interaction entre les variables est grande est laissée pour des travaux futurs. Toutefois, dans un souci de fournir des résultats exploitables, le Tableau 6.1 enregistre la qualité moyenne des solutions trouvées par les quatre algorithmes d'optimisation en considérant deux budgets d'optimisation distincts. Ainsi, nous définissons les budgets élevés comme approximativement égaux au nombre d'échantillons nécessaires pour que les modèles de substitution les plus précis convergent vers une erreur strictement nulle (conformément aux résultats présentés dans la section 6.2.2), plus environ 50% de cette quantité. Les budgets restreints sont alors définis comme le tiers des budgets élevés. Des tests statistiques sont utilisés pour déterminer la différence significative entre les résultats rapportés.

En ce qui concerne les problèmes UBQP, l'algorithme EGO et un modèle multilinéaire optimisé avec la technique EH fournissent des solutions de moins bonne qualité. Tous deux semblent rester coincés dans des optima locaux. Ce constat est d'autant plus remarquable quand la dimension des problèmes augmente. L'algorithme BOCS semble être très efficace lorsque la dimension des problèmes est faible (inférieure à 25 dimensions), et nécessite très peu d'échantillons pour converger vers des solutions optimales. Cependant, l'algorithme s'arrête radicalement dès que la taille du problème augmente. Cela explique les arrêts brutaux visibles sur les Figures 6.5 et 6.6 : l'algorithme n'a pas pu terminer le processus d'optimisation dans le temps de calcul imparti² pour des problèmes en 50 et 100 dimensions. Finalement, seule la technique d'optimisation proposée par l'Algorithme 6 assistée avec des méta-modèles de Walsh parvient à trouver des solutions de hautes qualités, quelle que soit la dimension du problème étudié.

En ce qui concerne les paysages NK, les performances sont similaires pour chaque algorithme lorsque la dimension est faible, bien que la méthode EGO semble moins efficace. Une fois de plus, l'algorithme BOCS ne passe pas à l'échelle

2. Le temps de calcul imparti pour chacun des algorithmes d'optimisation comparés est limité à 96 heures de calcul parallélisées sur 4 cœurs de CPU

avec la dimension. Enfin, la technique *WSaO* atteint des solutions de haute qualité sur des paysages \mathcal{NK} de grandes dimensions, même avec un budget de calcul restreint : ceux-ci fournissent des solutions de qualités légèrement supérieures à partir de 500 échantillons pour 100 dimensions.

Finalement, il semble que l'algorithme *vocs* offre une optimisation efficace pour les problèmes *UBQP* et les paysages \mathcal{NK} de dimensions modérées (jusqu'à 25). Cependant, ces performances ne passent pas à l'échelle pour des dimensions plus élevées, contrairement aux performances de *WSaO*. D'après notre analyse expérimentale, l'algorithme *vocs* ne peut pas être utilisé lorsque la dimension du problème est supérieure à 50, alors que *WSaO* peut être mis à l'échelle jusqu'à au moins 100 dimensions. Dans l'ensemble, nous constatons que le nombre d'échantillons nécessaires pour atteindre des solutions optimales est plus importante pour les problèmes *UBQP* que pour les paysages \mathcal{NK} . Par exemple en 100 dimensions, 3000 échantillons sont nécessaires à l'obtention d'une solution optimale pour les problèmes *UBQP*, alors qu'il suffit de 1000 échantillons pour les paysages \mathcal{NK} .

6.3 Analyse expérimentale pour le positionnement des arrêts de bus

Inspirés des méthodes introduites par la section 2.5, les travaux de cette section considèrent le problème du positionnement des arrêts de bus comme un problème d'optimisation pseudo-booléenne, avec le temps de trajet des passagers comme étant la fonction objectif à minimiser. Ainsi, une variable binaire est associée à chaque emplacement potentiel des arrêts de bus : la variable est égale à 1 si l'arrêt est activé ou égale à 0 dans le cas contraire. Ceci sera discuté plus en détail dans la section 6.3.1. Il a été établi dans les chapitres précédents que l'étude de la mobilité urbaine repose principalement sur la simulation informatique des flux de trafic. Celle-ci représente généralement un modèle boîte noire au regard des algorithmes d'optimisation, et nécessite quelques minutes voire quelques heures pour obtenir la valeur de la fonction objectif d'une seule simulation [9, 23]. Compte tenu de l'explosion combinatoire de

TABLEAU 6.1 – Qualité moyenne des solutions trouvées par les quatre algorithmes d'optimisation, selon un budget faible ou élevé consacré à la recherche. Les problèmes UBQP sont des problèmes de minimisation. Les paysages NK sont des problèmes de maximisation. Le symbole * signifie que la méthode n'a pas été complétée pour le problème considéré dans le temps imparti. Les moyennes apparaissent en gras lorsque la différence est statistiquement significative avec les tests de Mann-Whitney à 5% de précision, en fonction des problèmes et des budgets de calcul.

Problèmes	Budget faible			
	EGO	Multilinéaire (EH)	WSaO	BOCS
UBQP10	-104.67±27.62	-178.00±0.00	-176.67±5.16	-178.00±0.00
UBQP25	-539.27±71.34	-679.07±50.91	-831.80±30.50	-845.00±0.00
UBQP50	-1727.14±197.87	-2069.60±133.17	-3080.73±70.58	*
UBQP100	*	-3756.50±152.03	-6909.50±120.40	*
N10K1	0.75±0.04	0.80±0.00	0.79±0.01	0.80±0.00
N25K1	0.59±0.01	0.62±0.01	0.62±0.01	0.62±0.00
N50K1	0.70±0.01	0.72±0.01	0.71±0.01	*
N100K1	0.67±0.01	0.69±0.01	0.69±0.01	*
Problèmes	Budget élevé			
	EGO	Multilinéaire (EH)	WSaO	BOCS
UBQP10	-146.27±23.81	-178.00±0.00	-178.00±0.00	-178.00±0.00
UBQP25	-586.13±59.27	-707.73±34.83	-845.00±0.00	-845.00±0.00
UBQP50	-1924.67±237.10	-2161.93±122.73	-3141.00±0.00	*
UBQP100	*	-3811.00±74.95	-7522.00±0.00	*
N10K1	0.80±0.00	0.80±0.00	0.80±0.00	0.80±0.00
N25K1	0.62±0.01	0.63±0.00	0.63±0.00	0.63±0.00
N50K1	0.72±0.00	0.73±0.00	0.73±0.00	*
N100K1	0.67±0.01	0.70±0.00	0.71±0.00	*

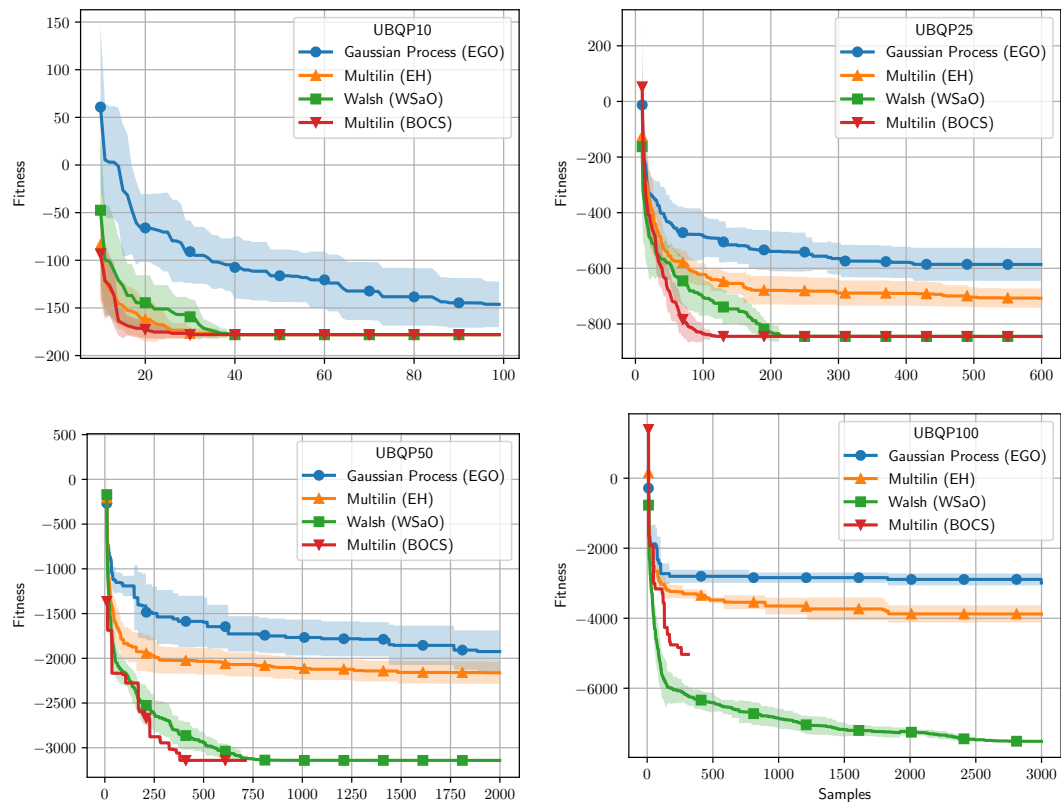


FIGURE 6.5 – Minimisation de la fonction objectif en fonction de la taille de l'échantillon, pour les problèmes UBQP. Quatre algorithmes d'optimisation assistée par méta-modèles sont comparés : la méthode de Krigeage avec l'algorithme EGO, la base multilinéaire avec l'algorithme EH, la base de Walsh avec l'algorithme EH (WSaO), et la base multilinéaire avec l'algorithme BOCS.

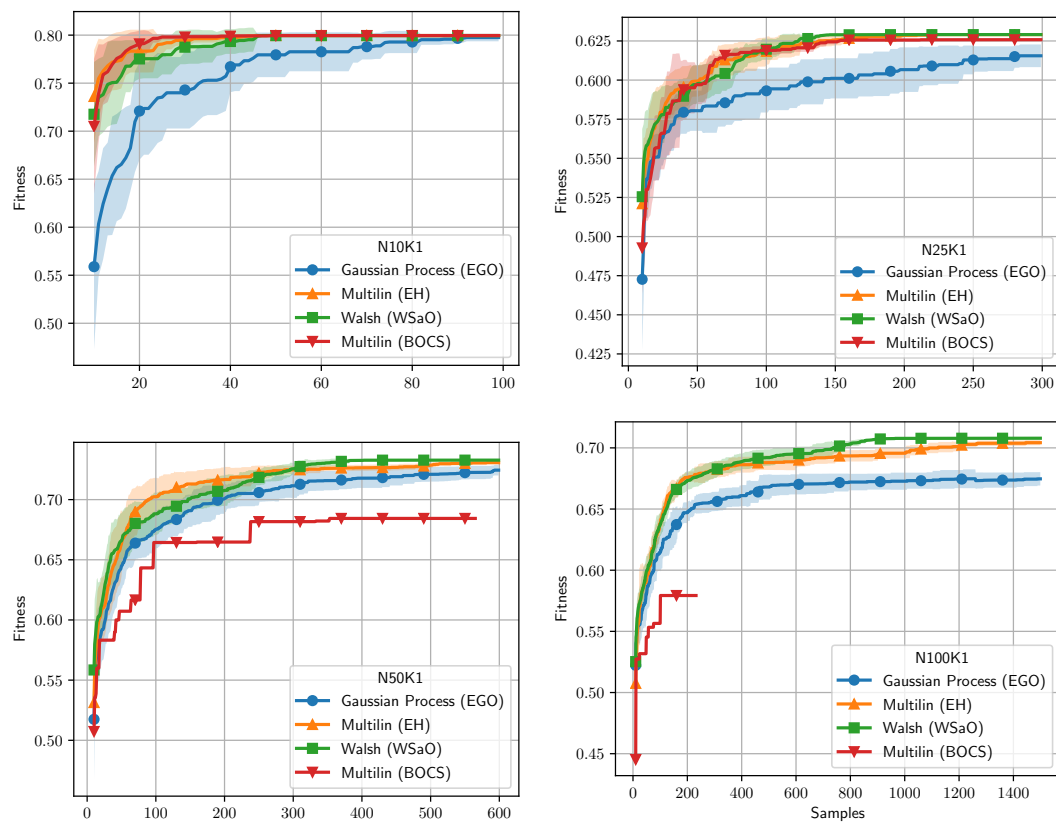


FIGURE 6.6 – Maximisation de la fonction objectif en fonction de la taille de l'échantillon, pour les paysages Nk , avec le paramètre k fixé à 1. Quatre algorithmes d'optimisation assistée par méta-modèles sont comparés : la méthode de Krigeage avec l'algorithme *EGO*, la base multilinéaire avec l'algorithme *EH*, la base de Walsh avec l'algorithme *EH* (*WSaO*), et la base multilinéaire avec l'algorithme *BOCS*.

l'espace de recherche, les experts en optimisation sont donc confrontés à une sérieuse limitation dans leurs capacités à explorer librement des configurations potentielles de l'espace urbain.

Pour relever un tel défi d'optimisation, nous proposons d'implémenter diverses techniques récentes de méta-modélisation spécialement dédiées au domaine combinatoire, afin d'approximer les simulations de trafic urbain. En outre, nous intégrons ces modèles de substitution dans un contexte d'optimisation pour le positionnement des arrêts de bus. Ces travaux sont les premiers à utiliser un algorithme d'optimisation assisté par méta-modèle basé sur les fondements mathématiques des fonctions de Walsh couplés à des techniques d'optimisation particulièrement efficaces [100], afin de résoudre une classe de problème du monde réel. Nous visons à mettre en évidence la précision et les performances d'optimisation de ces méthodes.

6.3.1 Définition du problème et simulation

Nous considérons le problème du positionnement des arrêts de bus comme un défi d'optimisation pseudo-booléenne. Sans perte de généralisation de la méthode, les travaux exposés dans la suite de cette section considèrent l'optimisation d'une ligne de bus régulière de la ville de Calais.

Arrêts de bus

Tous les arrêts de bus potentiels sont définis au préalable sur le réseau routier de la ville. En s'inspirant des travaux de Furth *et al.* [51], les arrêts potentiels sont localisés sur les noeuds d'intersection du réseau et sont numérotés de 1 à n , où n est le nombre total d'arrêts potentiels sur la ligne de bus, autrement dit la dimension du problème d'optimisation. Par conséquent, une configuration possible des arrêts de bus de la ville est représentée par une chaîne binaire $x \in \{0, 1\}^n$. Ainsi, les arrêts de bus ouverts sont associés aux variables de x égales à 1, tandis que les arrêts fermés sont associés aux variables égales à 0. Toutefois, le premier et le dernier arrêt de la ligne sont contraints d'être ouverts. Finalement, seuls les arrêts de bus ouverts sont pris en compte lors des simulations. La Figure 6.7 illustre un positionnement des arrêts de bus sur l'itinéraire étudié.

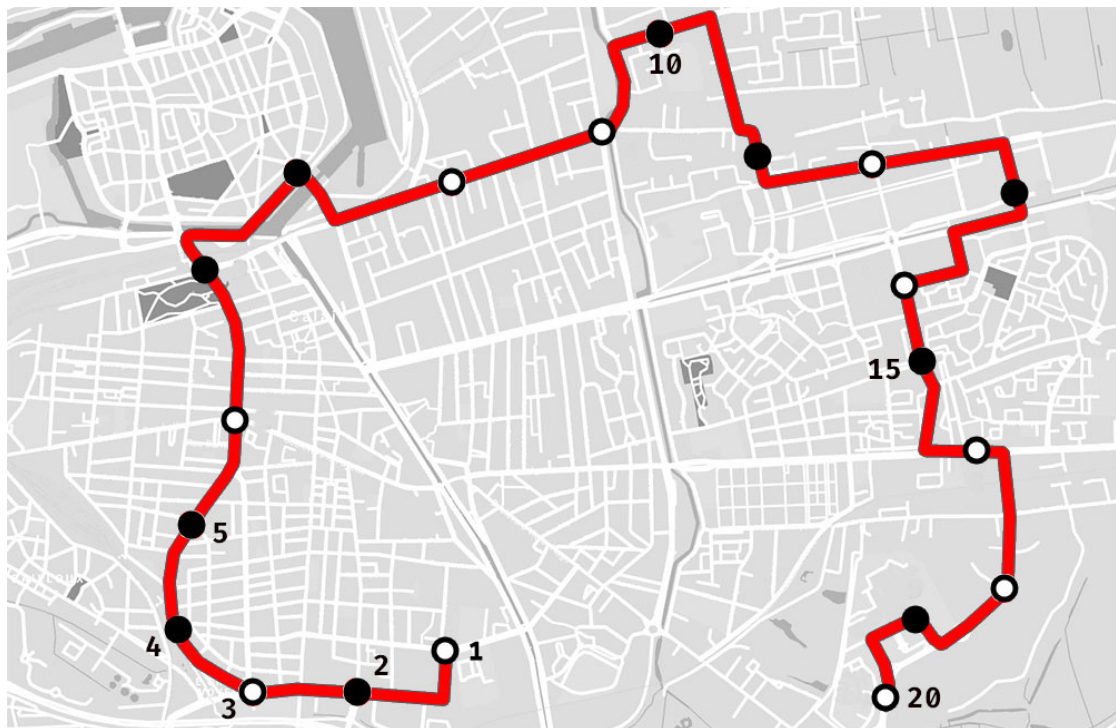


FIGURE 6.7 – Exemple d’une configuration pour des arrêts de bus ouverts (points blancs) ou fermés (points noirs) sur l’itinéraire du bus représenté par la ligne rouge, conformément à la solution $x = 10100100110010101101$.

L’itinéraire de bus complet prend en compte 20 positions d’arrêts potentiels.

Flux urbains

Ces travaux considèrent le système de simulation multi-agents *MATSim* [73]. *MATSim* nécessite comme intrants un modèle du réseau routier [131] et les scénarios de mobilité initiaux pour un ensemble d’agents. Ces scénarios sont générés par le *SIALAC* benchmark, introduit au chapitre 3. Ce dernier permet de synthétiser des plans de mobilité selon diverses topologies de l’espace urbain, en tenant compte des lieux d’habitation, des quartiers d’affaires ou des principaux points d’entrée et de sortie de la ville. Dans le contexte de ces travaux, nous étudions six scénarios impliquant 5000 voyageurs. Un scénario définit un aller-retour entre un lieu d’origine et un lieu d’activité, pour chaque voyageur. Ces derniers sont répartis en fonction de la topologie des six scénarios étudiés. Enfin,

il convient de rappeler qu'une simulation avec *MATSim*, autrement dit une évaluation de la fonction objectif, nécessite environ une minute de calcul dans un environnement mono-thread.

6.3.2 Protocole expérimental

Nous cherchons à minimiser le temps de trajet moyen des voyageurs estimé par *MATSim*, autrement dit la simulation boîte noire, en fonction du positionnement des arrêts de bus dans la ville. Nous utilisons des chaînes binaires composées de 20 variables, correspondant aux 20 emplacements potentiels des arrêts de bus. Nous suivons le même protocole expérimental que celui exposé par la section 6.2.1, à la différence que validons la précision des modèles par rapport à un ensemble de 250 solutions générées aléatoirement. Nous nous limitons aux interactions quadratiques entre les variables pour les méthodes *vocs* et *WSaO*. Les algorithmes et les expériences sont entièrement implémentés en Python, et utilisent des bibliothèques d'apprentissage et d'optimisation standard [127].

6.3.3 Précision des méta-modèles

Nous évaluons la précision de trois modèles de substitution basés sur une méthode de Krigeage, une décomposition en fonctions multilinéaires et une décomposition en fonctions de Walsh. Aucun algorithme d'optimisation n'est impliqué dans ces premières expérimentations : les solutions sélectionnées pour l'apprentissage des méta-modèles sont échantillonnées de façon aléatoire dans $\{0, 1\}^n$. La Figure 6.8 compare l'erreur moyenne absolue des modèles en fonction du nombre d'échantillons aléatoires dédié à leur apprentissage. Bien que le Krigeage semble prometteur dès les premières itérations, l'effort de calcul requis pour une légère amélioration de la précision augmente considérablement au fur et à mesure que le processus d'apprentissage progresse. En outre, la convergence de la précision des méta-modèles basés sur les polynômes multilinéaires ou les polynômes de Walsh est atteinte avec environ 400 échantillons du simulateur pour 3 scénarios du benchmark impliquant un unique cluster d'activité (*1h-1a*, *4h-1a* et *uh-1a*). Pour les scénarios impliquant quatre clusters d'activité (*1h-4a*, *4h-4a* et *uh-4a*), la précision des modèles semble augmenter au-delà de la limite

des 1000 échantillons fixée dans le cadre de ces expérimentations. D'une manière générale, les modèles de substitution basés sur les fonctions de Walsh sont les modèles les plus précis sur l'ensemble des scénarios. En effet, le gain de précision d'une décomposition en fonctions de Walsh par rapport à une décomposition multilinéaire est d'environ 30% à l'issue de chaque processus d'apprentissage.

6.3.4 Performances en optimisation

Nous comparons les algorithmes d'optimisation assistée par méta-modèles introduits par la section 1.4. En outre, nous comparons également les performances d'une décomposition en fonctions multilinéaires couplée à la technique EH [34]. Contrairement à l'analyse de précision de la section précédente, la solution ajoutée à l'échantillon d'apprentissage à chaque itération de l'algorithme représente maintenant la solution qui minimise le modèle de substitution (sauf pour *WSaO* si cette solution appartient déjà à l'échantillon, conformément à l'Algorithme 6). La Figure 6.9 illustre la minimisation du temps de trajet moyen des voyageurs en fonction de la taille de l'échantillon d'apprentissage, pour un scénario spécifique de mobilité. L'algorithme EGO semble converger trop rapidement vers un minimum local. Ce passage à l'échelle inapproprié a déjà été identifié sur des problèmes artificiels dans la section 6.2.3. D'autre part, *WSaO* semble découvrir des configurations de bonnes qualités avec un petit échantillon d'apprentissage, bien qu'un polynôme multilinéaire couplé à l'algorithme BOCS semble sensiblement meilleur lorsque la taille de l'échantillon augmente, comme rapporté par le Tableau 6.2. Cependant, il convient de préciser que la différence entre les deux approches est de l'ordre de la seconde. Aussi, ces deux méthodes semblent prometteuses pour ce scénario de dimension modérée impliquant une ligne de bus et 20 arrêts potentiels. Ces résultats encourageants sont encore en développement et font suite aux premiers résultats obtenus sur des problèmes d'optimisation artificiels [100].

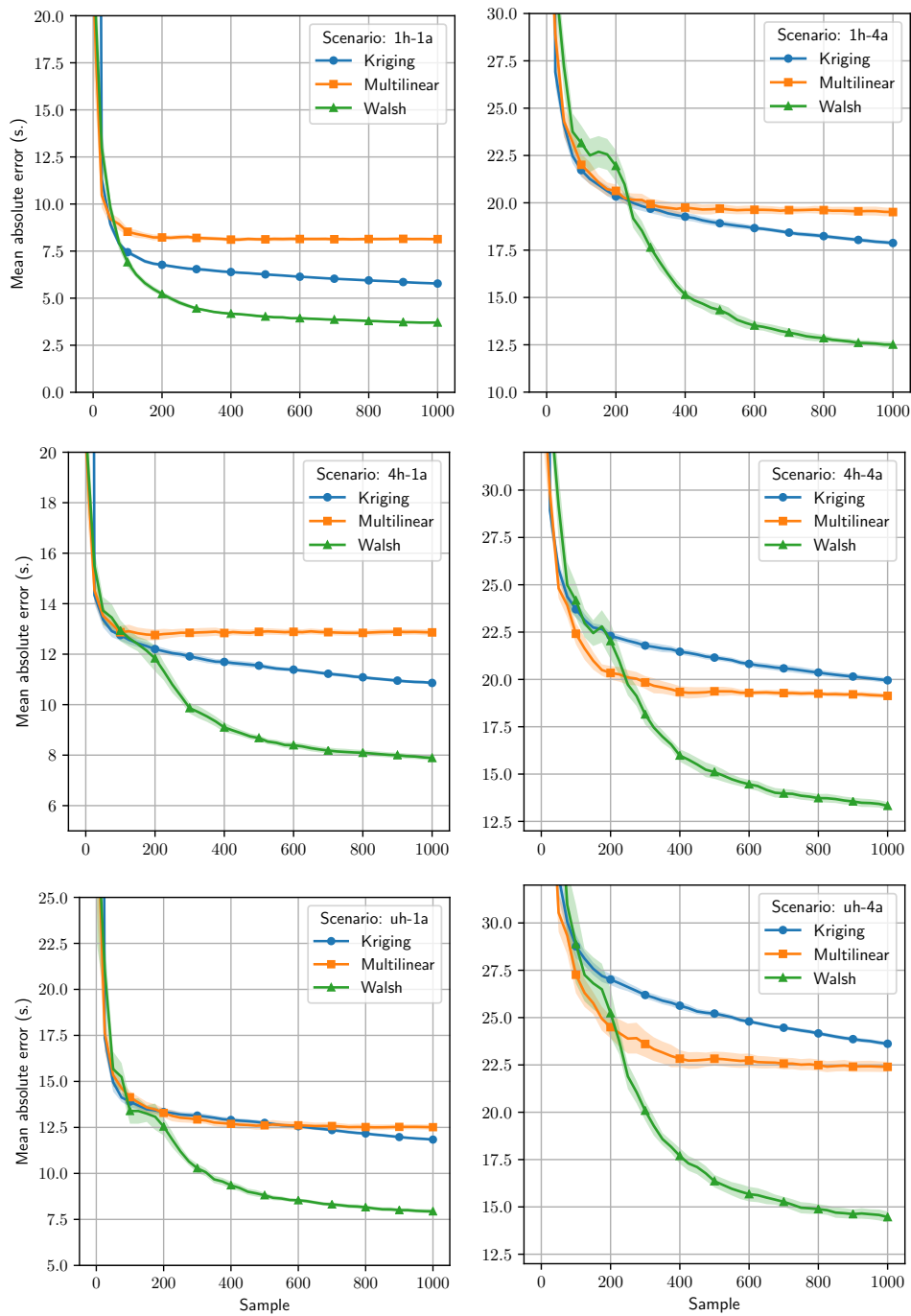


FIGURE 6.8 – Moyenne de l’erreur absolue sur un échantillon de test et intervalles de confiance des méta-modèles pour 6 scénarios du SIALAC benchmark, en fonction du nombre d’échantillons aléatoires dédiés à leur apprentissage (h : clusters d’origine, a : clusters d’activité).

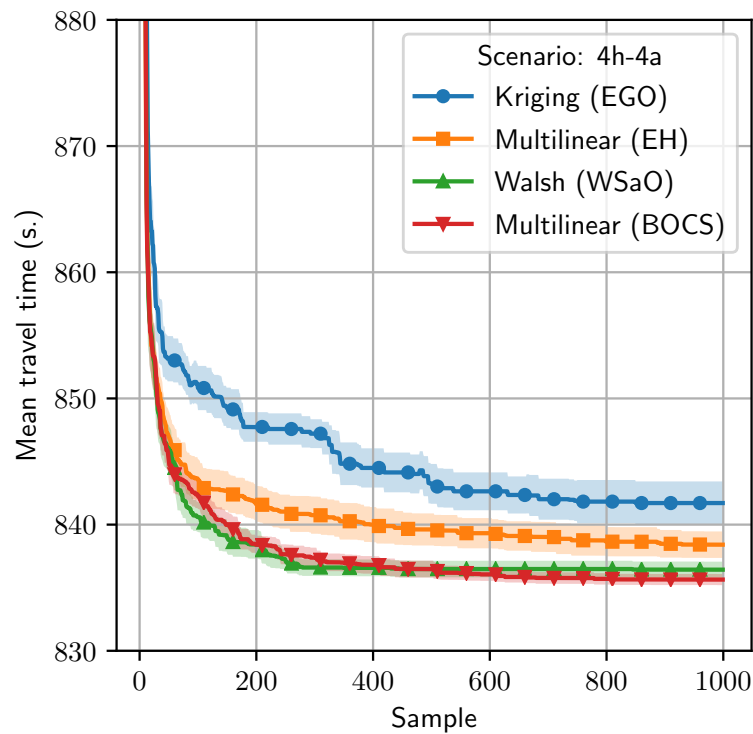


FIGURE 6.9 – Minimisation du temps de trajet des voyageurs en fonction de la taille de l'échantillon, pour un scénario du SIALAC benchmark. Quatre algorithmes d'optimisation assistée par méta-modèles sont comparés : la méthode de Krigeage avec l'algorithme EGO, la base multilinéaire avec l'algorithme EH, la base de Walsh avec l'algorithme EH (WSaO), et la base multilinéaire avec l'algorithme BOCS.

TABLEAU 6.2 – Durée moyenne du trajet des voyageurs (en s) en fonction de la taille de l'échantillon d'apprentissage pour le scénario de SIALAC : 4h-4a. Les valeurs apparaissent en gras lorsqu'elles sont statistiquement significatives selon les tests de Mann-Whitney à 5% de précision.

Echantillons	Kriging (EGO)	Multilinéaire (EH)	Walsh (WSaO)	Multilinéaire (BOCS)
100	850.82±4.00	843.55±4.17	840.59±3.03	842.24±2.58
400	844.49±3.55	839.99±3.56	836.57±1.70	836.81±1.78
1000	841.71±3.27	838.41±2.68	836.43±1.62	835.64±1.13

Résumé

Ce sixième chapitre propose la conception d'un algorithme d'optimisation assistée par un modèle de substitution, fondé sur la base mathématique des fonctions de Walsh : le *Walsh Surrogate-assisted Optimizer (WSaO)*. L'algorithme d'optimisation ainsi conçu tire parti d'une décomposition en fonctions de Walsh pour atteindre des solutions optimales en complexité linéaire. Dans un premier temps, les performances de *WSaO* sont analysées sur des problèmes d'optimisation académiques. La précision des modèles de substitution de Walsh outrepassé celle des autres modèles de la littérature. En outre, la bonne performance de *WSaO* en optimisation est d'autant plus remarquable lorsque la dimension des problèmes d'optimisation est grande. Dans un second temps, les mêmes algorithmes sont transposés et mis à l'épreuve pour résoudre une problématique de mobilité urbaine. Une fois encore, la précision des modèles de Walsh semble excellente, tandis que les performances en optimisation pour le positionnement des arrêts de bus rejoignent les résultats théoriques issus des analyses précédentes, pour des dimensionnalités modérées.

Conclusion

Ce mémoire traite de la modélisation, de la simulation et plus particulièrement de l'optimisation des flux de trafic urbain, afin de résoudre des problèmes de mobilité urbaine. La littérature consacrée aux problèmes d'optimisation et aux algorithmes de recherche est rapportée dans le chapitre 1. Un panorama des problèmes de mobilité urbaine et des techniques de résolution associées est rapporté dans le chapitre 2. Quatre contributions majeures sont présentées dans la deuxième partie de ce mémoire.

Le chapitre 3 introduit le SIALAC benchmark pour l'étude d'une variété de problèmes de mobilité. Ainsi, une centaine de scénarios regroupant diverses caractéristiques et topologies des espaces urbains sont proposés et analysés pour les villes de Calais et de Quito. En s'appuyant sur une étude approfondie des paysages de fitness, le chapitre 4 montre que la structure globale de l'espace de recherche est similaire pour l'ensemble des scénarios du benchmark, et propose une méthodologie basée sur des marches aléatoires pour détecter les variables importantes dans les problèmes d'optimisation. Le modèle d'importance ainsi établi permet de concevoir un algorithme d'optimisation adaptatif, basé sur une stratégie d'apprentissage par renforcement, pour l'optimisation du réglage des feux de signalisation. Par conséquent, le chapitre 5 introduit l'algorithme *Bandit Descent* pour aborder divers scénarios de mobilité aux caractéristiques hétérogènes. L'analyse expérimentale indique que l'apprentissage par renforcement, associé au modèle d'importance des variables, permet d'obtenir des réglages de haute qualité pour un certain nombre de scénarios, tout en nécessitant un faible budget calculatoire et un moindre besoin des connaissances expertes en urbanisme. Enfin, le chapitre 6 introduit un algorithme d'optimisation assistée par méta-modèle. La méthode combine la base orthogonale des fonctions de

Walsh et une technique d'optimisation particulièrement pertinente pour cette modélisation. L'algorithme est comparé à des travaux états-de-l'art et se révèle performant pour des problèmes d'optimisation artificiels impliquant jusqu'à cent dimensions. La méthode proposée est alors appliquée pour la résolution d'un problème d'optimisation de la mobilité urbaine traitant du positionnement optimal des arrêts de bus.

En résumé, les travaux rapportés par ce mémoire combinent à la fois des techniques d'optimisation efficaces en termes de complexité temporelle ou calculatoire, et des techniques d'apprentissage artificiel telles que des méthodes de régression ou d'apprentissage par renforcement. Aussi, ces travaux s'inscrivent dans le domaine de l'intelligence artificielle, puisqu'ils traitent de la résolution automatique d'une variété de problèmes liés à l'amélioration de la mobilité urbaine. En effet, l'approche par les paysages de fitness associée à la technique d'apprentissage par renforcement ne donne pas seulement un algorithme d'optimisation efficace, mais fournit une explication aux experts urbanistes de la manière dont l'algorithme visualise et interprète automatiquement l'espace urbain. De plus, les approches par méta-modèles proposées dans le contexte de ces travaux recherchent des modèles non boîtes noires (contrairement à des réseaux de neurones), adaptés à l'optimisation, et qui pourraient eux-mêmes être analysés à des fins d'explicabilité.

En outre, les travaux de ce manuscrit ouvrent de nombreuses perspectives de recherche pour des travaux futurs. Tout d'abord, les scénarios de mobilité générés par le SIALAC benchmark peuvent être définis avec une plus grande précision afin de modéliser la dynamique d'une variété de villes à travers le monde et d'en étudier les paysages de fitness. Dans un deuxième temps, l'algorithme adaptatif *Bandit Descent* peut être amélioré grâce à l'utilisation de stratégies de redémarrage ou de fenêtres temporelles [48], et avec un paramétrage plus robuste de la constante d'exploration des stratégies UCB. Par ailleurs, le modèle d'importance des variables établi au cours d'une marche aléatoire ouvre la voie à l'intégration d'une détection adaptative des variables importantes pendant le processus d'optimisation. Troisièmement, l'analyse des performances de l'algorithme d'optimisation assistée par un méta-modèle de Walsh pousse à concevoir une version parallèle de la méthode. En effet, les travaux rapportés

n'utilisent qu'une simple recherche locale itérative avec un redémarrage purement aléatoire. L'efficacité de la méthode d'optimisation pourrait encore être améliorée en utilisant un algorithme à population de solution et un croisement basé sur les points d'articulation entre les solutions [32]. En outre, la différence de performances des divers méta-modèles, selon la dimension ou la complexité du problème d'optimisation, suggère de concevoir un ensemble de modèles de substitution qui combinerait les différentes approches. Enfin, il est envisageable d'étendre le problème du positionnement des arrêts de bus à une centaine de dimensions au moins, afin de concevoir des itinéraires de bus plus concis et de confronter les méthodes de la littérature à des problèmes d'optimisation plus difficiles. En effet, une dimension plus élevée permettrait de tirer des conclusions plus claires quant à la décomposition polynomiale la plus appropriée. En outre, ces modèles polynomiaux sont limités ici aux interactions quadratiques entre les variables. L'interaction cubique (ou plus élevée) est envisagée afin de prétendre à la conception de modèles de substitution encore plus précis.

Enfin, les travaux rapportés par ce mémoire visent à susciter la réflexion sur des problèmes de mobilité urbaine, en s'appuyant sur les points de vue des experts urbanistes et des experts en optimisation, afin de parvenir à une mobilité urbaine optimale et durable.

Bibliographie

- [1] Brahim ABOUTAIB, Cyril FONLUPT, Virginie MARION et Sébastien VEREL. « Urban traffic modeling and simulation : Moroccan capital case study ». In : *The 2nd International Conference on Smart Applications and Data Analysis for Smart Cities*. Casablanca, Morocco, juil. 2018.
- [2] Khulood ALYAHYA et Jonathan E ROWE. « Simple Random Sampling Estimation of the Number of Local Optima ». In : *Parallel Problem Solving from Nature*. Springer. 2016, p. 932-941.
- [3] A AMUTHAN et K Deepa THILAK. « Survey on Tabu search meta-heuristic optimization ». In : *International Conference on Signal Processing, Communication, Power and Embedded System*. IEEE. 2016, p. 1539-1543.
- [4] David L APPLGATE, Robert E BIXBY, Vasek CHVATAL et William J COOK. *The traveling salesman problem : a computational study*. Princeton university press, 2006.
- [5] Renato Oliveira ARBEX et Claudio Barbieri da CUNHA. « Efficient transit network design and frequencies setting multi-objective optimization by alternating objective genetic algorithm ». In : *Transportation Research Part B : Methodological* 81 (2015), p. 355-376.
- [6] Kellie J. ARCHER et Ryan V. KIMES. « Empirical Characterization of Random Forest Variable Importance Measures ». In : *Computational Statistics and Data Analysis* 52.4 (2008), p. 2249-2260.
- [7] R. ARMAS, H. AGUIRRE, F. DAOLIO et K. TANAKA. « Traffic signal optimization and coordination using neighborhood mutation ». In : *Congress on Evolutionary Computation*. Juil. 2016, p. 395-402.
- [8] Rolando ARMAS, Hernan AGUIRRE et Kiyoshi TANAKA. « Multi-objective Optimization of Level of Service in Urban Transportation ». In : *Proceedings of the Genetic and Evolutionary Computation Conference*. Berlin, Germany : ACM, 2017, p. 1057-1064.

- [9] Rolando ARMAS, Hernán AGUIRRE, Saúl ZAPOTECAS-MARTÍNEZ et Kiyoshi TANAKA. « Traffic Signal Optimization : Minimizing Travel Time and Fuel Consumption ». In : *Artificial Evolution*. Springer, 2016, p. 29-43.
- [10] Peter AUER, Nicolò CESA-BIANCHI et Paul FISCHER. « Finite-time Analysis of the Multiarmed Bandit Problem ». In : *Mach. Learn.* 47.2-3 (2002), p. 235-256.
- [11] Thomas BÄCK, David B FOGEL et Zbigniew MICHALEWICZ. *Evolutionary computation 1 : Basic algorithms and operators*. CRC press, 2018.
- [12] Michael BALMER, Nurhan CETIN, Kai NAGEL et Bryan RANEY. « Towards truly agent-based traffic and mobility simulations ». In : *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*. IEEE. 2004, p. 60-67.
- [13] Ricardo BAPTISTA et Matthias POLOCZEK. « Bayesian Optimization of Combinatorial Structures ». In : *International Conference on Machine Learning*. 2018, p. 462-471.
- [14] Johan BARTHELEMY et Philippe L TOINT. « Synthetic population generation without a sample ». In : *Transportation Science* 47.2 (2013), p. 266-279.
- [15] Thomas BARTZ-BEIELSTEIN et Martin ZAEFFERER. « Model-based methods for continuous and discrete global optimization ». In : *Applied Soft Computing* 55 (2017), p. 154-167.
- [16] John E BEASLEY. « Heuristic algorithms for the unconstrained binary quadratic programming problem ». In : *London, England* (1998).
- [17] Mandell BELLMORE et George L NEMHAUSER. « The traveling salesman problem : a survey ». In : *Operations Research* 16.3 (1968), p. 538-558.
- [18] Maurizio BIELLI, Massimiliano CARAMIA et Pasquale CAROTENUTO. « Genetic algorithms in bus network optimization ». In : *Transportation Research Part C : Emerging Technologies* 10.1 (2002), p. 19-34.
- [19] Richard BISHOP. « Intelligent vehicle applications worldwide ». In : *IEEE Intelligent Systems and Their Applications* 15.1 (2000), p. 78-81.
- [20] Richard BISHOP. *Intelligent vehicle technology and trends*. 2005.
- [21] Léon BOTTOU. « Stochastic Gradient Descent Tricks ». In : *Neural Networks : Tricks of the Trade : Second Edition*. Sous la dir. de Grégoire MONTAVON, Geneviève B. ORR et Klaus-Robert MÜLLER. Berlin, Heidelberg : Springer Berlin Heidelberg, 2012, p. 421-436.
- [22] GEORGE EP BOX, Gwilym M JENKINS et G REINSEL. « Time series analysis : forecasting and control Holden-day San Francisco ». In : *BoxTime Series Analysis : Forecasting and Control Holden Day* 1970 (1970).

- [23] Juergen BRANKE. « Simulation Optimization Tutorial ». In : *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. ACM. 2018.
- [24] Norman BRESLOW. « A generalized Kruskal-Wallis test for comparing K samples subject to unequal patterns of censorship ». In : *Biometrika* 57.3 (1970), p. 579-594.
- [25] Jan BRINKMANN, Marlin W ULMER et Dirk C MATTFELD. « Inventory routing for bike sharing systems ». In : *Transportation research procedia* 19 (2016), p. 316-327.
- [26] Alberto BROGGI, Michele BUZZONI, Stefano DEBATTISTI, Paolo GRISLERI, Maria Chiara LAGHI, Paolo MEDICI et Pietro VERSARI. « Extensive tests of autonomous driving technologies ». In : *IEEE Transactions on Intelligent Transportation Systems* 14.3 (2013), p. 1403-1415.
- [27] Yan CAO, Guantao CHEN, Guangming JING, Michael STIEBITZ et Bjarne TOFT. « Graph edge coloring : A survey ». In : *Graphs and Combinatorics* 35.1 (2019), p. 33-66.
- [28] Avishai CEDER. « Bus frequency determination using passenger count data ». In : *Transportation Research Part A : General* 18.5-6 (1984), p. 439-453.
- [29] Avishai CEDER et Nigel HM WILSON. « Bus network design ». In : *Transportation Research Part B : Methodological* 20.4 (1986), p. 331-344.
- [30] Vladimír ČERNÝ. « Thermodynamical approach to the traveling salesman problem : An efficient simulation algorithm ». In : *Journal of optimization theory and applications* 45.1 (1985), p. 41-51.
- [31] Bofei CHEN. « A multi-agent based cooperative control model applied to the management of vehicles-trains ». Thèse de doct. 2017.
- [32] Francisco CHICANO, Gabriela OCHOA, Darrell WHITLEY et Renato TINÓS. « Enhancing partition crossover with articulation points analysis ». In : *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM. 2018, p. 269-276.
- [33] Francisco CHICANO, Darrell WHITLEY, Gabriela OCHOA et Renato TINÓS. « Optimizing one million variable NK landscapes by hybridizing deterministic recombination and local search ». In : *GECCO*. 2017, p. 753-760.

- [34] Francisco CHICANO, Darrell WHITLEY et Andrew M. SUTTON. « Efficient Identification of Improving Moves in a Ball for Pseudo-boolean Problems ». In : *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*. GECCO '14. Vancouver, BC, Canada : ACM, 2014, p. 437-444.
- [35] S. CHIU et S. CHAND. « Adaptive Traffic Signal Control Using Fuzzy Logic ». In : *The First IEEE Regional Conference on Aerospace Control Systems*, 1993, p. 122-126.
- [36] Wikimedia COMMONS. *Ant Colony*. 2018.
- [37] Wikimedia COMMONS. *TSP instance*. 2016.
- [38] José R CORREA et Nicolás E STIER-MOSES. « Wardrop equilibria ». In : *Wiley encyclopedia of operations research and management science* (2010).
- [39] Michael Lynn CRAMER. « A representation for the adaptive generation of simple sequential programs ». In : *proceedings of an International Conference on Genetic Algorithms and the Applications*. 1985, p. 183-187.
- [40] Noel CRESSIE. « The origins of kriging ». In : *Mathematical geology* 22.3 (1990), p. 239-252.
- [41] Fabio DAOLIO, Arnaud LIEFOOGHE, Sébastien VEREL, Hernán AGUIRRE et Kiyoshi TANAKA. « Problem Features versus Algorithm Performance on Rugged Multiobjective Combinatorial Fitness Landscapes ». In : *Evolutionary Computation Journal* 25.4 (2017), p. 555-585.
- [42] Gülesin Sena DAŞ, Fatma GZARA et Thomas STÜTZLE. « A review on airport gate assignment problems : Single versus multi objective approaches ». In : *Omega* 92 (2020), p. 102146.
- [43] Luigi DELL'OLIO, Jose Luis MOURA et Angel IBEAS. « Bi-level mathematical programming model for locating bus stops and optimizing frequencies ». In : *Transportation research record* 1971.1 (2006), p. 23-31.
- [44] Stanley N DEMING, Lloyd R PARKER JR et M BONNER DENTON. « A review of simplex optimization in analytical chemistry ». In : *CRC Critical Reviews in Analytical Chemistry* 7.3 (1978), p. 187-202.
- [45] Marco DORIGO, Alberto COLORNI et Vittorio MANIEZZO. *Distributed optimization by ant colonies*. 1991.
- [46] Marco DORIGO, Vittorio MANIEZZO et Alberto COLORNI. « Ant system : optimization by a colony of cooperating agents ». In : *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 26.1 (1996), p. 29-41.

- [47] Kurt DRESNER et Peter STONE. « A multiagent approach to autonomous intersection management ». In : *Journal of artificial intelligence research* 31 (2008), p. 591-656.
- [48] Amaury DUBOIS, Julien DEHOS et Fabien TEYTAUD. « Improving Multimodal Optimization Restart Strategy Through Multi-armed Bandit ». In : *2018 17th IEEE International Conference on Machine Learning and Applications*. IEEE, 2018, p. 338-343.
- [49] L ESHELMAN. « On crossover as an evolutionarily viable strategy ». In : *Proceedings of the Fourth International Conference on Genetic Algorithms*. 1991, p. 61-68.
- [50] Jacques FERBER et Gerhard WEISS. *Multi-agent systems : an introduction to distributed artificial intelligence*. T. 1. Addison-Wesley Reading, 1999.
- [51] Peter FURTH et Adam B RAHBEE. « Optimal Bus Stop Spacing Through Dynamic Programming and Geographic Modeling ». In : *Transportation Research Record* 1731 (2000), p. 15-22.
- [52] Ralph GAKENHEIMER. « Urban mobility in the developing world ». In : *Transportation Research Part A : Policy and Practice* 33.7 (1999), p. 671-689.
- [53] Donald T GANZ et James R MEKEMSON. *Flow profile comparison of a microscopic car-following model and a macroscopic platoon dispersion model for traffic simulation*. Rapp. tech. Institute of Electrical et Electronics Engineers (IEEE), 1990.
- [54] Juntao GAO, Yulong SHEN, Jia LIU, Minoru ITO et Norio SHIRATORI. « Adaptive traffic signal control : Deep reinforcement learning algorithm with experience replay and target network ». In : *arXiv preprint arXiv :1705.02755* (2017).
- [55] K. GAO, Yicheng ZHANG, A. SADOLLAH et Rong SU. « Improved artificial bee colony algorithm for solving urban traffic light scheduling problem ». In : *Congress on Evolutionary Computation*. Juin 2017, p. 395-402.
- [56] J. GARCIA-NIETO, J. FERRER et E. ALBA. « Optimising traffic lights with metaheuristics : Reduction of car emissions and consumption ». In : *2014 International Joint Conference on Neural Networks*. Juil. 2014, p. 48-54.
- [57] Juan Carlos GARCÍA-PALOMARES, Javier GUTIÉRREZ et Marta LATORRE. « Optimizing the location of stations in bike-sharing programs : A GIS approach ». In : *Applied Geography* 35.1-2 (2012), p. 235-246.
- [58] Michel GENDREAU et Jean-Yves POTVIN. « Tabu search ». In : *Search methodologies*. Springer, 2005, p. 165-186.

- [59] Stephen GLAISTER. « Bus deregulation, competition and vehicle size ». In : *Journal of Transport Economics and Policy* (1986), p. 217-244.
- [60] Fred GLOVER. « Future paths for integer programming and links to artificial intelligence ». In : *Computers operations research* 13.5 (1986), p. 533-549.
- [61] David E GOLDBERG. « Genetic algorithms and Walsh functions : Part I, a gentle introduction ». In : *Complex systems* 3.2 (1989), p. 129-152.
- [62] David E GOLDBERG et John Henry HOLLAND. « Genetic algorithms and machine learning ». In : (1988).
- [63] Genetha Anne GRAY, Tamara G KOLDA, Ken SALE et Malin M YOUNG. « Optimizing an empirical scoring function for transmembrane protein structure determination ». In : *INFORMS Journal on Computing* 16.4 (2004), p. 406-418.
- [64] Dominik GREYER. « Extension of a multi-agent transport simulation for traffic signal control and air transport systems ». Thèse de doct. Technische Universität Berlin, Fakultät V - Verkehrs- und Maschinensysteme, 2014.
- [65] Dominik GREYER, Joschka BISCHOFF et Kai NAGEL. « Traffic-actuated signal control : Simulation of the user benefits in a big event real-world scenario ». In : *2nd International Conference on Models and Technologies for ITS, Leuven, Belgium*. 2011, p. 11-12.
- [66] Peter L HAMMER et Ron HOLZMAN. « Approximations of pseudo-Boolean functions ; applications to game theory ». In : *Zeitschrift für Operations Research* 36.1 (1992), p. 3-21.
- [67] Jeongwoo HAN, Michael KOKKOLARAS et Panos Y PAPALAMBROS. « Optimal design of hybrid fuel cell vehicles ». In : *Journal of fuel cell science and technology* 5.4 (2008).
- [68] Nikolaus HANSEN, Steffen FINCK, Raymond Ros et Anne AUGER. « Real-parameter black-box optimization benchmarking 2009 : Noiseless functions definitions ». In : (2009).
- [69] Pierre HANSEN et Brigitte JAUMARD. « Algorithms for the maximum satisfiability problem ». In : *Computing* 44.4 (1990), p. 279-303.
- [70] John Henry HOLLAND et al. *Adaptation in natural and artificial systems*. MIT press, 1975.
- [71] Robert HOOKE et T. A. JEEVES. « “ Direct Search ” Solution of Numerical and Statistical Problems ». In : *J. ACM* 8.2 (avr. 1961), p. 212-229.

- [72] Holger H. Hoos. « Automated Algorithm Configuration and Parameter Tuning ». In : *Autonomous Search*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2012, p. 37-71.
- [73] Andreas HORNI, Kai NAGEL et Kay AXHAUSEN, éd. *Multi-Agent Transport Simulation MATSim*. London : Ubiquity Press, août 2016.
- [74] Yingjie HU, JianQiang HU, Yifan XU, Fengchun WANG et Rong Zeng CAO. « Contamination control in food supply chain ». In : *Proceedings of the 2010 Winter Simulation Conference*. IEEE. 2010, p. 2678-2681.
- [75] Hsien-Kuei HWANG, Alois PANHOLZER, Nicolas ROLIN, Tsung-Hsi TSAI et Wei-Mei CHEN. « Probabilistic analysis of the (1+ 1)-evolutionary algorithm ». In : *Evolutionary computation* 26.2 (2018), p. 299-345.
- [76] Ángel IBEAS, Luigi DELL'OLIO, Borja ALONSO et Olivia SAINZ. « Optimizing bus stop spacing in urban areas ». In : *Transportation Research Part E : Logistics and Transportation Review* 46.3 (2010), p. 446-458.
- [77] K ILAVARASI et K Suresh JOSEPH. « Variants of travelling salesman problem : A survey ». In : *International Conference on Information Communication and Embedded Systems*. IEEE. 2014, p. 1-7.
- [78] L. INGBER. « Very fast simulated re-annealing ». In : *Mathematical and Computer Modelling* 12.8 (1989), p. 967-973.
- [79] Sanjay JAIN et Charles R McLEAN. « A concept prototype for integrated gaming and simulation for incident management ». In : *Proceedings of the 2006 Winter Simulation Conference*. IEEE. 2006, p. 493-500.
- [80] Jan Owen JANSSON. « A simple bus line model for optimisation of service frequency and bus size ». In : *Journal of Transport Economics and Policy* (1980), p. 53-80.
- [81] Vaithilingam JEYAKUMAR et Alexander M RUBINOV. *Continuous Optimization : Current Trends and Modern Applications*. T. 99. Springer Science & Business Media, 2006.
- [82] Yaochu JIN. « Surrogate-assisted evolutionary computation : Recent advances and future challenges ». In : *Swarm and Evolutionary Computation* 1.2 (2011), p. 61-70.
- [83] Donald R JONES, Matthias SCHONLAU et William J WELCH. « Efficient global optimization of expensive black-box functions ». In : *Journal of Global optimization* 13.4 (1998), p. 455-492.
- [84] T. JONES. « Evolutionary Algorithms, Fitness Landscapes and Search ». Thèse de doct. University of New Mexico, Albuquerque, 1995.

- [85] S. A. KAUFFMAN. *The Origins of Order*. Oxford University Press, 1993.
- [86] Pascal KERSCHKE, Mike PREUSS, Simon WESSING et Heike TRAUTMANN. « Low-budget exploratory landscape analysis on multiple peaks models ». In : *Proceedings of the Genetic and Evolutionary Computation Conference 2016*. ACM. 2016, p. 229-236.
- [87] Jin Hee KIM, Yun Kyung BAE et J CHUNG. « Multi-objective Optimization for Sustainable Road Network Design Problem ». In : *Int. Conference on Transport, Environment and Civil Engineering*. 2012, p. 104-108.
- [88] Gary KOCHENBERGER, Jin-Kao HAO, Fred GLOVER, Mark LEWIS, Zhipeng LÜ, Haibo WANG et Yang WANG. « The unconstrained binary quadratic programming problem : A survey ». In : *Journal of Combinatorial Optimization* 28 (juil. 2014).
- [89] Tamara G KOLDA, Robert Michael LEWIS et Virginia TORCZON. « Optimization by direct search : New perspectives on some classical and modern methods ». In : *SIAM review* 45.3 (2003), p. 385-482.
- [90] John R KOZA et John R KOZA. *Genetic programming : on the programming of computers by means of natural selection*. T. 1. MIT press, 1992.
- [91] Daniel KRAJZEWICZ, Jakob ERDMANN, Michael BEHRISCH et Laura BIEKER. « Recent Development and Applications of SUMO - Simulation of Urban MObility ». In : *International Journal On Advances in Systems and Measurements*. International Journal On Advances in Systems and Measurements 5.3&4 (déc. 2012), p. 128-138.
- [92] Reinhart D KÜHNE et Malte B RÖDIGER. « Macroscopic simulation model for freeway traffic with jams and stop-start waves ». In : *Proceedings of the 23rd conference on Winter simulation*. IEEE Computer Society. 1991, p. 762-770.
- [93] Tze Leung LAI et Herbert ROBBINS. « Asymptotically efficient adaptive allocation rules ». In : *Advances in applied mathematics* 6.1 (1985), p. 4-22.
- [94] Oliver LAMOTTE, Stephane GALLAND, Jean-Michel CONTET et Franck GECHTER. « Submicroscopic and physics simulation of autonomous and intelligent vehicles in virtual reality ». In : *2010 Second International Conference on Advances in System Simulation*. IEEE. 2010, p. 28-33.
- [95] Larry J LEBLANC. « Transit system network design ». In : *Transportation Research Part B : Methodological* 22.5 (1988), p. 383-390.
- [96] Yann LECUN, Yoshua BENGIO et Geoffrey HINTON. « Deep learning ». In : *nature* 521.7553 (2015), p. 436-444.

- [97] Jon LEE. *A first course in combinatorial optimization*. T. 36. Cambridge University Press, 2004.
- [98] Florian LEPRÊTRE, Cyril FONLUPT, Sébastien VEREL et Virginie MARION. « Combinatorial Surrogate-Assisted Optimization for Bus Stops Spacing Problem ». In : *Biennial International Conference on Artificial Evolution (EA 2019)*. Mulhouse, France, oct. 2019.
- [99] Florian LEPRÊTRE, Cyril FONLUPT, Sébastien VEREL et Virginie MARION. « SIALAC benchmark : on the design of adaptive algorithms for traffic lights problems ». In : *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. ACM. 2018, p. 288-289.
- [100] Florian LEPRÊTRE, Cyril FONLUPT, Sébastien VEREL et Virginie MARION. « Walsh Functions as Surrogate Model for Pseudo-Boolean Optimization Problems ». In : *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM. 2019.
- [101] Florian LEPRÊTRE, Cyril FONLUPT, Sébastien VEREL, Virginie MARION, Rolando ARMAS, Hernán AGUIRRE et Kiyoshi TANAKA. « Fitness landscapes analysis and adaptive algorithms design for traffic lights optimization on SIALAC benchmark ». In : *Applied Soft Computing* 85 (2019), p. 105869.
- [102] Florian LEPRÊTRE, Fabien TEYTAUD et Julien DEHOS. « Multi-armed bandit for stratified sampling : Application to numerical integration ». In : *TAAI 2017 - Conference on Technologies and Applications of Artificial Intelligence*. Taipei, Taiwan, déc. 2017.
- [103] Arnaud LIEFOOGHE, Sébastien VEREL, Luis PAQUETE et Jin-Kao HAO. « Experiments on local search for bi-objective unconstrained binary quadratic programming ». In : *EMO - 2015 8th International Conference on Evolutionary Multi-Criterion Optimization*. T. 9018. Lecture Notes in Computer Science. Guimarães, Portugal, 2015, p. 171-186.
- [104] Jialin LIU, Diego Perez LIEBANA et Simon M. LUCAS. « Bandit-Based Random Mutation Hill-Climbing ». In : *Congress on Evolutionary Computation*. 2017, p. 2145-2151.
- [105] Junming LIU, Qiao LI, Meng QU, Weiwei CHEN, Jingyuan YANG, Hui XIONG, Hao ZHONG et Yanjie FU. « Station site optimization in bike sharing systems ». In : *2015 IEEE International Conference on Data Mining*. IEEE. 2015, p. 883-888.
- [106] Manuel LÓPEZ-IBÁÑEZ, Jérémie DUBOIS-LACOSTE, Leslie Pérez CÁCERES, Mauro BIRATTARI et Thomas STÜTZLE. « The irace package : Iterated racing for automatic algorithm configuration ». In : *Operations Research Perspectives* 3 (2016), p. 43-58.

- [107] Jihui MA, Yanqing ZHAO, Yang YANG, Tao LIU, Wei GUAN, Jiao WANG et Cuiying SONG. « A model for the stop planning and timetables of customized buses ». In : *PloS one* 12.1 (2017).
- [108] Enes MAKALIC et Daniel F SCHMIDT. « A simple sampler for the horseshoe estimator ». In : *IEEE Signal Processing Letters* 23.1 (2016), p. 179-182.
- [109] Katherine M. MALAN et Andries P. ENGELBRECHT. « A Survey of Techniques for Characterising Fitness Landscapes and Some Possible Ways Forward ». In : *Information Sciences* 241 (2013), p. 148-163.
- [110] Alison L MARSDEN, Jeffrey A FEINSTEIN et Charles A TAYLOR. « A computational framework for derivative-free optimization of cardiovascular geometries ». In : *Computer methods in applied mechanics and engineering* 197.21-24 (2008), p. 1890-1905.
- [111] B Chandra MOHAN et R BASKARAN. « A survey : Ant Colony Optimization based recent research and implementation on several engineering domain ». In : *Expert Systems with Applications* 39.4 (2012), p. 4618-4627.
- [112] Jean-Marc MONTANIER et Nicolas BREDECHE. « Embedded evolutionary robotics : The (1+ 1)-restart-online adaptation algorithm ». In : *New horizons in evolutionary robotics*. Springer, 2011, p. 155-169.
- [113] Alberto MORAGLIO et Ahmed KATTAN. « Geometric generalisation of surrogate model based optimisation to combinatorial spaces ». In : *European Conference on Evolutionary Computation in Combinatorial Optimization*. Springer. 2011, p. 142-154.
- [114] Seyed Sajad MOUSAVI, Michael SCHUKAT et Enda HOWLEY. « Traffic light control using deep policy-gradient and value-function-based reinforcement learning ». In : *IET Intelligent Transport Systems* 11.7 (2017), p. 417-423.
- [115] Freddy Antony MULLAKKAL-BABU, Meng WANG, Bart van AREM, Barys SHYROKAU et Riender HAPPEE. « A Hybrid Submicroscopic-Microscopic Traffic Flow Simulation Framework ». In : *IEEE Transactions on Intelligent Transportation Systems* (2020).
- [116] Kirill MÜLLER et Kay W AXHAUSEN. *Population synthesis for microsimulation : State of the art*. ETH Zürich, Institut für Verkehrsplanung, Transporttechnik, Strassen-und Eisenbahnbau (IVT), 2010.
- [117] Mathieu MUNIGLIA, Sébastien VEREL, Jean-Charles Le PALLEC et Jean-Michel DO. « Massive asynchronous master-worker EA for nuclear reactor optimization : a fitness landscape perspective ». In : *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. ACM. 2017, p. 295-296.

- [118] Mario A MUÑOZ, Yuan SUN, Michael KIRLEY et Saman K HALGAMUGE. « Algorithm selection for black-box continuous optimization problems : A survey on methods and challenges ». In : *Information Sciences* 317 (2015), p. 224-245.
- [119] Diana M NEGOESCU, Peter I FRAZIER et Warren B POWELL. « The knowledge-gradient algorithm for sequencing experiments in drug discovery ». In : *INFORMS Journal on Computing* 23.3 (2011), p. 346-363.
- [120] Eoin O'MAHONY et David B SHMOYS. « Data analysis and optimization for (citi) bike sharing ». In : *Twenty-ninth AAAI conference on artificial intelligence*. 2015.
- [121] S. M. ODEH, A. M. MORA, M. N. MORENO et J. J. MERELO. « A Hybrid Fuzzy Genetic Algorithm for an Adaptive Traffic Signal System ». In : *Advances in Fuzzy Systems* 2015 (2015).
- [122] S. M. ODEH, A. M. MORA, M. N. MORENO et J. J. MERELO. « A Hybrid Fuzzy Genetic Algorithm for an Adaptive Traffic Signal System ». In : *Adv. Fuzzy Sys.* 2015 (2015).
- [123] Rodrigue OEUVRAY. *Trust-region methods based on radial basis functions with application to biomedical imaging*. Rapp. tech. EPFL, 2005.
- [124] Johan Janson OLSTAM, Jan LUNDGREN, Mikael ADLERS et Pontus MATSTOMS. « A framework for simulation of surrounding vehicles in driving simulators ». In : *ACM Transactions on Modeling and Computer Simulation (TOMACS)* 18.3 (2008), p. 1-24.
- [125] Christos H PAPANIMITRIOU et Kenneth STEIGLITZ. *Combinatorial optimization : algorithms and complexity*. Courier Corporation, 1998.
- [126] Praveen PARUCHURI, Alok Reddy PULLALAREVU et Kamalakar KARLAPALEM. « Multi agent simulation of unorganized traffic ». In : *Proceedings of the first international joint conference on Autonomous agents and multiagent systems : part 1*. 2002, p. 176-183.
- [127] F. PEDREGOSA et al. « Scikit-learn : Machine Learning in Python ». In : *Journal of Machine Learning Research* 12 (2011), p. 2825-2830.
- [128] Roneeta PURKAYASTHA, Tanmay CHAKRABORTY, Anirban SAHA et Debarka MUKHOPADHYAY. « Study and Analysis of Various Heuristic Algorithms for Solving Travelling Salesman Problem—A Survey ». In : *Proceedings of the Global AI Congress 2019*. Springer. 2020, p. 61-70.
- [129] Michal RADECKÝ et Petr GAJDOŠ. « Intelligent agents for traffic simulation ». In : *Proceedings of the 2008 Spring simulation multiconference*. 2008, p. 109-115.

- [130] Eric RAMAT et Bertrand VANDORNE. *Plant growth model for decision making support*. Rapp. tech. Université du Littoral Côte d'Opale, et ISA Lille, 2002.
- [131] Frederik RAMM, Christine KARCH et Jochen TOPF. *Geofabrik*. <https://www.geofabrik.de>. Accessed : 2018-01-24. 2007.
- [132] Esteban RICALDE et Wolfgang BANZHAF. « A Genetic Programming Approach for the Traffic Signal Control Problem with Epigenetic Modifications ». In : *Genetic Programming*. Springer, 2016, p. 133-148.
- [133] Hendrik RICHTER et Andries ENGELBRECHT. *Recent advances in the theory and application of fitness landscapes*. Springer, 2014.
- [134] Marcel RIESER. « Adding Transit to an Agent-Based Transportation Simulation : Concepts and Implementation ». Thèse de doct. der Technischen Universität Berlin, 2010.
- [135] Juan P ROMERO, Angel IBEAS, Jose L MOURA, Juan BENAVENTE et Borja ALONSO. « A simulation-optimization approach to design efficient systems of bike-sharing ». In : *Procedia-Social and Behavioral Sciences* 54 (2012), p. 646-655.
- [136] Anthony A. SAKA. « Model for Determining Optimum Bus-Stop Spacing in Urban Areas ». In : *Journal of Transportation Engineering* 127.3 (2001), p. 195-199.
- [137] J. J. SANCHEZ-MEDINA, M. J. GALAN-MORENO et E. RUBIO-ROYO. « Traffic Signal Optimization in La Almozara District in Saragossa Under Congestion Conditions, Using Genetic Algorithms, Traffic Microsimulation, and Cluster Computing ». In : *IEEE Transactions on Intelligent Transportation Systems* 11.1 (2010), p. 132-141.
- [138] Lisa A SCHAEFER, Gerald T MACKULAK, Jeffery K COCHRAN et Jennifer L CHERILLA. « Application of a general particle system model to movement of pedestrians and vehicles ». In : *1998 Winter Simulation Conference. Proceedings (Cat. No. 98CH36274)*. T. 2. IEEE. 1998, p. 1155-1160.
- [139] Bruno Castro da SILVA, Ana LC BAZZAN, Gustavo K ANDRIOTTI, Filipe LOPES et Denise de OLIVEIRA. « ITSUMO : an intelligent transportation system for urban mobility ». In : *International Workshop on Innovative Internet Community Systems*. Springer. 2004, p. 224-235.
- [140] John SOUSANIS. « World vehicle population tops 1 billion units ». In : *Wards Auto* 15 (2011).
- [141] Daniel SPERLING et Deborah GORDON. *Two billion cars : driving toward sustainability*. Oxford University Press, 2010.

- [142] P. F. STADLER. « Fitness Landscapes ». In : *Biological Evolution and Statistical Physics*. T. 585. LNP. Springer, 2002, p. 187-207.
- [143] Daniel H STOLFI et Enrique ALBA. « Smart mobility policies with evolutionary algorithms : the adapting info panel case ». In : *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*. 2015, p. 1287-1294.
- [144] Rainer STORN et Kenneth PRICE. « Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces ». In : *Journal of global optimization* 11.4 (1997), p. 341-359.
- [145] El-Ghazali TALBI. *Metaheuristics : from design to implementation*. T. 74. John Wiley & Sons, 2009.
- [146] Sara TARI, Matthieu BASSEUR et Adrien GOËFFON. « Sampled Walk and Binary Fitness Landscapes Exploration ». In : *Artificial Evolution*. Cham : Springer International Publishing, 2018, p. 47-57.
- [147] F. TEKLU, A. SUMALEE et D. WATLING. « A Genetic Algorithm Approach for Optimizing Traffic Control Signals Considering Routing ». In : 22 (2007), p. 31-43.
- [148] *The world's cities in 2016*. United Nations, Department of Economic et Social Affairs, 2016.
- [149] Robert TIBSHIRANI, Martin WAINWRIGHT et Trevor HASTIE. *Statistical learning with sparsity : the lasso and generalizations*. Chapman et Hall/CRC, 2015.
- [150] Vesselin K VASSILEV, Terence C FOGARTY et Julian F MILLER. « Smoothness, ruggedness and neutrality of fitness landscapes : from theory to application ». In : *Advances in evolutionary computing*. Springer, 2003, p. 3-44.
- [151] RJ VAUGHAN et EA COUSINS. « Optimum location of stops on a bus route ». In : *International Symposium on Transportation and Traffic Theory, 7th, 1977, Kyoto, Japan*. 1977.
- [152] Gerhard VENTER. « Review of optimization techniques ». In : *Encyclopedia of aerospace engineering* (2010).
- [153] Sébastien VEREL. « Contributions to fitness landscapes analysis for single- and multi-objective optimization ». Habilitation à diriger des recherches. Université du Littoral Côte d'Opale, déc. 2016.

- [154] Sébastien VEREL, Bilel DERBEL, Arnaud LIEFOOGHE, Hernan AGUIRRE et Kiyoshi TANAKA. « A surrogate model based on Walsh decomposition for pseudo-boolean functions ». In : *International Conference on Parallel Problem Solving from Nature*. Springer. 2018, p. 181-193.
- [155] Sinnathamby VIJAYAKUMAR. « Optimal vehicle size for road-based urban public transport in developing countries ». In : *Transport Reviews* 6.2 (1986), p. 193-212.
- [156] Carole VILET et Laurent PEZIN. *Schéma de développement urbain*. Rapp. tech. Ville de Calais, juil. 2010.
- [157] *Vissim*. <https://www.ptvgroup.com/fr/solutions/produits/ptv-vissim/>. Accessed : 2020-04-30.
- [158] Vukan R VUCHIC. « Urban public transportation systems ». In : *University of Pennsylvania, Philadelphia, PA, USA* (2002).
- [159] J. L. WALSH. « A Closed Set of Normal Orthogonal Functions ». In : *American Journal of Mathematics* 45.1 (1923), p. 5-24.
- [160] Hua WEI, Guanjie ZHENG, Huaxiu YAO et Zhenhui LI. « Intellilight : A reinforcement learning approach for intelligent traffic light control ». In : *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM. 2018, p. 2496-2505.
- [161] E. D. WEINBERGER. « Correlated and uncorrelated fitness landscapes and how to tell the difference ». In : *Biological Cybernetics*. 1990, 63 :325-336.
- [162] Darrell WHITLEY, Renato TINÓS et Francisco CHICANO. « Optimal Neuron Selection and Generalization : NK Ensemble Neural Networks ». In : *Parallel Problem Solving from Nature*. Sous la dir. d'Anne AUGER, Carlos M. FONSECA, Nuno LOURENÇO, Penousal MACHADO, Luís PAQUETE et Darrell WHITLEY. Cham : Springer International Publishing, 2018, p. 449-460.
- [163] Stefan M WILD, Rommel G REGIS et Christine A SHOEMAKER. « ORBIT : Optimization by radial basis function interpolation in trust-regions ». In : *SIAM Journal on Scientific Computing* 30.6 (2008), p. 3197-3219.
- [164] S. C. WIRASINGHE et Nadia S. GHONEIM. « Spacing of Bus-Stops for Many to Many Travel Demand ». In : *Transportation Science* 15.3 (1981), p. 210-221.
- [165] Jochen WITTMANN, Johannes GÖBEL, Dietmar MÖLLER et Bernard SCHROER. « Refinement of the virtual intermodal transportation system (vits) and adoption for metropolitan area traffic simulation ». In : *Proceedings of the 2007 summer computer simulation conference*. 2007, p. 1-5.

- [166] Fei YAN. « Contribution à la modélisation et à la régulation du trafic aux intersections : Intégration des communications Véhicule-Infrastructure ». Thèse de doct. 2012.
- [167] Nicholas YOUNG. *An introduction to Hilbert space*. Cambridge university press, 1988.
- [168] Martin ZAEFFERER. « Surrogate Models for Discrete Optimization Problems ». Thèse de doct. Technischen Universitat Dortmund, déc. 2018.
- [169] Martin ZAEFFERER, Jörg STORK et Thomas BARTZ-BEIELSTEIN. « Distance measures for permutations in combinatorial efficient global optimization ». In : *International Conference on Parallel Problem Solving from Nature*. Springer, p. 373-383.
- [170] Martin ZAEFFERER, Jörg STORK, Martina FRIESE, Andreas FISCHBACH, Boris NAUJOKS et Thomas BARTZ-BEIELSTEIN. « Efficient Global Optimization for Combinatorial Problems ». In : *GECCO*. 2014.
- [171] Changjiang ZHENG, Shukang ZHENG et Genghua MA. « The Bus Station Spacing Optimization Based on Game Theory ». In : *Advances in Mechanical Engineering* 7.2 (2015), p. 453979.
- [172] Zhenjun ZHU, X GUO, H CHEN, J ZENG et Jiangling WU. « Optimization of Urban Mini-bus Stop Spacing : A Case Study of Shanghai (China) ». In : *Tehnicki Vjesnik* 24 (juin 2017), p. 949-955.