

PIERRE-LUC GRÉGOIRE-GIRARD

Communication inter-véhicules et route-à-véhicule
Apprentissage de la communication inter-véhicules

Mémoire présenté
à la Faculté des études supérieures de l'Université Laval
dans le cadre du programme de maîtrise en informatique
pour l'obtention du grade de Maître ès sciences, (M.Sc.)

FACULTÉ DES SCIENCES ET DE GÉNIE
UNIVERSITÉ LAVAL
QUÉBEC

Août 2008

Résumé

L'industrie des transports est un acteur important de l'économie mondiale. Des millions d'emplois sont touchés de près ou de loin par cette industrie. De plus, le nombre de véhicules sur terre ne cesse d'augmenter et il est rendu courant dans plusieurs pays industrialisés d'avoir plus d'une voiture par famille. Tout cela entraîne également son lot de problèmes, notamment au niveau de la sécurité routière et de la pollution. Depuis des décennies, des compagnies privées et des organisations publiques se penchent sur ces problèmes. Ils innovent constamment en améliorant les véhicules et les infrastructures du réseau routier. Les voitures ne sont pas seulement de plus en plus sécuritaires mais aussi de plus en plus confortables et conviviales. Cependant, la plupart des systèmes intelligents présents dans les voitures acquièrent de l'information grâce aux capteurs. Ceux-ci sont limités et certaines données ne peuvent être accessibles aux capteurs. La solution repose donc sur l'utilisation de la communication sans fil pour échanger des informations entre les véhicules et entre les infrastructures et les véhicules.

Le sujet de cette maîtrise porte sur la communication inter-véhicules et route-à-véhicule. Elle se divise en deux parties. Tout d'abord, dans le cadre du projet AUTO21 au laboratoire DAMAS, un protocole de communication inter-véhicules doit être développé pour implanter dans un système de régulateur de vitesse collaboratif et adaptatif (« Collaborative Adaptive Cruise Control », CACC). La théorie multi-agents et l'apprentissage par renforcement sont utilisés pour apprendre une politique de communication optimale. La deuxième partie porte sur la communication route-à-véhicule appliquée au problème d'optimisation des feux de signalisation. Deux approches multi-agents sont utilisées pour optimiser la gestion des feux. Les agents placés aux intersections reçoivent de l'information relative au trafic grâce à la communication route-à-véhicule et tentent d'adopter une politique de contrôle optimale.

Avant-propos

Tout d'abord, j'aimerais profiter de ces quelques lignes pour remercier les personnes qui m'ont aidé et supporté pour la réalisation de ce mémoire. En premier lieu, je voudrais remercier mon directeur de recherche, Dr Brahim Chaib-draa, qui m'a guidé et aidé tout au long de cette maîtrise. Sa grande disponibilité et la liberté qu'il nous offre pour explorer différents champs de recherche sont grandement appréciées.

Je voudrais aussi remercier Abder Alikacem et Martin-Pierre Roy qui m'ont encouragé à entreprendre des études supérieures. De plus, travaillant à temps partiel tout au long de ces deux années et demie d'étude, je tiens à remercier mes supérieurs chez Cardinal Health et Myca qui ont fait preuve de compréhension et de souplesse face à ma disponibilité réduite. Plus particulièrement, je remercie Daniel Huot et Martin-Pierre Roy pour leur support.

Je veux aussi remercier Charles Desjardins et Julien Laumônier avec qui j'ai travaillé sur le projet AUTO21. Il est toujours agréable de faire équipe avec des gens sympathiques et travaillants comme Charles et Julien. Merci aussi aux autres membres du laboratoire DAMAS pour leurs nombreux conseils et pour le bon temps passé en leur compagnie. J'adresse également des remerciements à AUTO21 pour le soutien financier et pour les conférences annuelles qui nous ont permis de faire des rencontres chaleureuses. Merci à Lynda Goulet, agente de gestion des études à l'Université Laval, pour son dévouement envers les étudiants.

Je remercie aussi mes amis qui me permettent de garder un équilibre de vie en alliant études, travail et loisir. Un merci particulier à ma copine, Stéphanie, qui m'a supporté et encouragé tout au long de la rédaction de ce mémoire.

Finalement, je remercie mes parents, Jean Grégoire et Céline Girard, à qui je dédie ce mémoire. Ils m'ont non seulement encouragé et supporté tout au long de mes études mais aussi dans toutes les sphères de ma vie. Pour tout, merci infiniment.

*Je dédie ce mémoire à mes parents,
Jean Grégoire et Céline Girard.*

Table des matières

| | |
|--|-----------|
| Résumé | ii |
| Avant-propos | iii |
| Table des matières | v |
| Liste des tableaux | viii |
| Table des figures | x |
| 1 Introduction | 1 |
| 1.1 Problématique | 2 |
| 1.2 Motivations | 3 |
| 1.3 Objectifs | 4 |
| 1.4 AUTO21 | 4 |
| 1.5 Organisation du mémoire | 5 |
| 2 L'intelligence artificielle et les systèmes de transport intelligents | 7 |
| 2.1 La robotique | 7 |
| 2.2 L'intelligence artificielle et les agents | 9 |
| 2.3 Concepts d'IA | 11 |
| 2.3.1 Résolution de problèmes, arbres de recherche | 11 |
| 2.3.2 Problèmes de satisfaction de contraintes | 12 |
| 2.3.3 Apprentissage machine | 13 |
| 2.3.4 Théorie des jeux | 13 |
| 2.3.5 Autres | 14 |
| 2.4 Les systèmes de transport intelligents | 14 |
| 2.5 Conclusion | 16 |
| 3 Notions de communications et de communication sans fil | 17 |
| 3.1 Communication entre ordinateurs | 17 |
| 3.1.1 Architectures réseau | 18 |
| 3.1.2 Les couches sous-réseau | 19 |

| | | |
|----------|--|-----------|
| 3.1.3 | Les couches hôtes | 21 |
| 3.2 | Communication sans fil | 23 |
| 3.3 | Réseaux mobiles et communication inter-véhicules | 25 |
| 3.3.1 | Applications de la communication | 26 |
| 3.3.2 | Besoins technologiques des applications | 27 |
| 3.3.3 | État de l'art | 30 |
| 3.3.4 | DSRC et WAVE | 31 |
| 3.4 | Conclusion | 32 |
| 4 | Apprentissage de la communication inter-véhicules | 33 |
| 4.1 | Architecture du CACC | 34 |
| 4.2 | Processus décisionnels de Markov (MDP) et apprentissage par renforcement | 37 |
| 4.3 | Implémentation de la communication véhiculaire | 42 |
| 4.3.1 | Modélisation des couches de bas niveau | 42 |
| 4.3.2 | Architecture technique | 43 |
| 4.3.3 | Communication dans le module de coordination | 45 |
| 4.3.4 | Communication dans le module action | 45 |
| 4.4 | Environnement de simulation, simulateur | 49 |
| 4.5 | Résultats | 51 |
| 4.6 | Conclusion | 53 |
| 5 | Communication route-à-véhicule : solution au problème de congestion | 54 |
| 5.1 | Le trafic et la congestion | 54 |
| 5.1.1 | Étendu du problème de congestion | 55 |
| 5.1.2 | Causes, définitions et notions relatives au trafic | 56 |
| 5.1.3 | Solutions au problème de congestion | 61 |
| 5.2 | Gestion des intersections et des feux de signalisation | 64 |
| 5.2.1 | Conception d'une intersection | 64 |
| 5.2.2 | Contrôle des feux de signalisation | 65 |
| 5.2.3 | Problématique de l'optimisation des feux de signalisation | 66 |
| 5.2.4 | Simulateurs de trafic | 66 |
| 5.3 | Approche par la communication et les agents | 67 |
| 5.4 | État de l'art sur l'optimisation des feux de signalisation | 68 |
| 5.5 | Conclusion | 69 |
| 6 | Optimisation des feux de signalisation | 70 |
| 6.1 | Problème et modélisation | 70 |
| 6.1.1 | Complexité du problème | 71 |
| 6.1.2 | Modélisation d'une intersection | 73 |
| 6.1.3 | Débit d'une intersection en fonction de la demande | 76 |
| 6.1.4 | Choix d'actions pour un agent | 77 |

| | | |
|----------|---|------------|
| 6.1.5 | Fonction de récompense | 77 |
| 6.2 | Apprentissage par renforcement pour la gestion de carrefours | 79 |
| 6.2.1 | Simulation pour l'apprentissage | 79 |
| 6.2.2 | Modélisation sous forme de MDP | 80 |
| 6.2.3 | Résultats d'apprentissage | 82 |
| 6.2.4 | Conclusion partielle | 84 |
| 6.3 | Approche dynamique | 84 |
| 6.3.1 | Inconvénients de l'apprentissage | 85 |
| 6.3.2 | Intuition et description de l'approche | 85 |
| 6.3.3 | Décortiquation du problème et fonctionnement de l'agent | 86 |
| 6.3.4 | Synchronisation des lumières et coordination entre les agents | 93 |
| 6.3.5 | Simulateur | 94 |
| 6.3.6 | Résultats | 94 |
| 6.3.7 | Conclusion | 99 |
| 7 | Conclusion | 102 |
| 7.0.8 | Contributions | 103 |
| 7.0.9 | Travaux futurs | 103 |
| A | Urban Traffic Control Based on Learning Agents | 105 |
| | Bibliographie | 106 |

Liste des tableaux

| | | |
|------|---|----|
| 2.1 | Exemple d’environnements de tâche pour différents agents. | 11 |
| 3.1 | Modèles de diffusion [Wikipédia, 2008a]. | 22 |
| 3.2 | Résumé des protocoles de la famille 802.11. | 26 |
| 3.3 | Exemples d’applications basées sur la communication inter-véhicules. | 28 |
| 3.4 | Exemples d’applications basées sur la communication route-à-véhicule. | 29 |
| 4.1 | Actions des couches. | 36 |
| 5.1 | Temps moyen des déplacements pour faire l’aller-retour entre la maison et le lieu de travail, en minutes [Turcotte, 2005]. | 55 |
| 5.2 | Total annuel des coûts liés à la congestion par ville (en millions de dollars selon la valeur monétaire en 2002) [Transports Canada, 2006]. | 56 |
| 5.3 | Temps d’attente des usagers à l’entrée du pont Pierre-Laporte [Transport Québec, 2008b]. | 62 |
| 6.1 | Problème du contrôle des feux sous forme de MDP. | 71 |
| 6.2 | Exemple de phases possibles. | 76 |
| 6.3 | Moyenne et écart type du délai des véhicules pour un scénario à une intersection (Calculés sur 50 simulations). | 83 |
| 6.4 | Moyenne et écart type du délai des véhicules pour un scénario à quatre intersections (Calculés sur 50 simulations). | 84 |
| 6.5 | Répartition de la demande pour chaque connexion. La demande est exprimée en véhicules par heure. | 88 |
| 6.6 | Ensemble de phases possibles. | 89 |
| 6.7 | Pourcentage minimal du temps pour lequel chaque connexion doit avoir le droit de passage de façon à satisfaire la demande. | 90 |
| 6.8 | Cycle optimal calculé par l’agent. | 91 |
| 6.9 | Cycle du contrôleur fixe. | 97 |
| 6.10 | Temps de simulation et temps de parcourt moyen pour les deux contrôleurs. | 97 |
| 6.11 | Répartition de la demande pour chaque connexion de chaque modèle. La demande est exprimée en véhicules par heure. | 99 |

6.12 Cycles choisis par l'agent, chaque phase est représentée par la liste de connexions qu'elle contient. 101

Table des figures

| | | |
|------|---|----|
| 1.1 | Les six thèmes d'AUTO21. | 4 |
| 2.1 | Les trois révolutions industrielles. | 8 |
| 2.2 | Interraction d'un agent avec son environnement. | 10 |
| 2.3 | Exemple du jeu de Sudoku. Il faut placer des nombres dans chaque case en respectant les contraintes. | 13 |
| 3.1 | Les couches, les protocoles et les interfaces. | 19 |
| 3.2 | Le modèle de référence OSI [Tanenbaum, 2003]. | 20 |
| 3.3 | Réseau avec point d'accès et Réseau ad hoc. | 24 |
| 3.4 | Problème de la station cachée. A et C communiquent en même temps à B. | 25 |
| 3.5 | Réseau mobile dynamique avec point d'accès où les noeuds (véhicules) formant le réseau changent constamment. | 26 |
| 4.1 | Architecture du système CACC. | 35 |
| 4.2 | Choix de voie collaboratif. | 35 |
| 4.3 | Interractions entre les deux couches du système CACC. | 37 |
| 4.4 | Architecture d'un agent intelligent Russell et Norvig [2003]. | 38 |
| 4.5 | Processus décisionnel de Markov. | 39 |
| 4.6 | Architecture de la communication dans le simulateur. | 44 |
| 4.7 | Interraction en le module action et le module de perceptions. | 47 |
| 4.8 | Perception que le dernier véhicule se fait du premier. | 47 |
| 4.9 | Visualisation d'une simulation avec intersection. | 50 |
| 4.10 | Nombre de messages envoyés pour 1000 épisodes de 1000 étapes. | 52 |
| 4.11 | Erreur sur l'estimation de la position du véhicule. | 52 |
| 5.1 | Débit maximal en fonction de la vitesse. | 58 |
| 5.2 | Debit en fonction de la demande. | 60 |
| 5.3 | Vitesse en fonction de la demande. | 60 |
| 6.1 | « Simplification » de l'environnement réel par un environnement simulé : l'agent apprend à performer dans l'environnement simplifié-simulé. | 73 |

| | | |
|------|--|-----|
| 6.2 | « Simplification » de l'environnement par l'agent : l'agent apprend dans un environnement simulé « proche » de la réalité. | 74 |
| 6.3 | Intersection avec 12 voies sources et 12 voies de destination. | 75 |
| 6.4 | Apprentissage utilisant l'algorithme de gradient de politique [Grégoire <i>et al.</i> , 2007]. | 83 |
| 6.5 | Reseau routier de six intersections. La direction du trafic est vers le centre-ville. | 87 |
| 6.6 | Demande en véhicules par heure. | 87 |
| 6.7 | Les 12 connexions de l'intersection. | 88 |
| 6.8 | Réseau routier de la ville de Québec vu dans le simulateur SUMO. | 95 |
| 6.9 | Débit maximum en fonction du temps pour les connexions 1, 2 et 3. | 96 |
| 6.10 | Nombre de voitures dans le réseau en fonction du temps. | 98 |
| 6.11 | Choix du cycle par l'agent. | 100 |
| 6.12 | Nombre de véhicules en fonction du temps. | 101 |

Chapitre 1

Introduction

En l'an 2000, il y avait près de 600 millions de véhicules sur la terre, incluant camions et autobus, et il est prévu que ce nombre doublerait d'ici trente ans. L'industrie automobile est donc l'une des plus développées au Monde et on ne cesse de perfectionner la technologie à bord des véhicules. Outre l'amélioration du confort et de la qualité de conduite, la recherche dans ce domaine vise présentement à régler les trois plus gros problèmes causés par ce très grand nombre de véhicules sur la terre : (i) diminuer le nombre d'accidents, (ii) diminuer le trafic dans les grandes villes et (iii) réduire la pollution.

La recherche dans le cadre des systèmes de transport intelligents (« Intelligent Transport Systems », ITS) s'attaque aux deux premiers problèmes à savoir (i) et (ii). Une multitude de fonctionnalités ont vu le jour rapidement et nous pourrions les séparer de façon générale en trois catégories. Premièrement, plusieurs options visent à améliorer ou augmenter l'information visuelle du conducteur. Par exemple, certains manufacturiers offrent des phares adaptatifs qui modifient leurs angles d'éclairage dans un tournant ou encore installent des témoins lumineux avertissant la présence d'obstacles en marche arrière. Il y a également les systèmes de contrôle de bas niveau qui visent à augmenter la stabilité de la voiture dans différentes situations. Entrent dans cette catégorie les systèmes de répartition du freinage sur chacune des roues pour maintenir la trajectoire du véhicule dans un tournant ou encore les freins antiblocage (« Anti-lock Braking System », ABS) qui empêchent le dérapage lors d'une immobilisation brusque. Finalement, nous pourrions inclure dans une troisième catégorie les systèmes de contrôle de plus haut niveau qui remplacent une partie de la tâche de conduite. Les meilleurs exemples de cette catégorie sont évidemment les régulateurs de vitesse et les régulateurs de vitesse adaptatifs (« Adaptive Cruise Control », ACC) qui ajustent la vitesse du véhicule à celle du véhicule précédent. Tous ces systèmes visent à augmenter la sécurité routière

ou à régulariser le trafic pour diminuer la congestion.

1.1 Problématique

Pour l'instant, les capteurs sont aux coeurs de la plupart des dispositifs d'avant-garde dans le domaine. Cependant, ils sont relativement limités quant à la distance d'opération et à la fiabilité de détection. Par exemple, les capteurs radars ou lasers utilisés pour les ACC ne fonctionnent qu'au-dessus d'une certaine vitesse pour leur permettre de distinguer les voitures en mouvement des voitures ou des objets arrêtés sur le bord de la route.

Une solution à ces problèmes serait donc d'utiliser la communication inter-véhicules et la communication route-à-véhicule. En plus d'augmenter la fiabilité de l'information reçue, elle permet d'élargir considérablement le champ d'information pouvant être recueillie par la voiture dans le but d'aider le conducteur à prendre de meilleures décisions ou encore d'améliorer les systèmes de conduite automatique.

Plusieurs défis doivent être cependant surmontés pour permettre l'implantation de la communication dans les systèmes de transport intelligents. Par exemple, une problématique de la communication est de conjuguer avec un taux de pénétration variable de la technologie dans le marché. Comparativement aux capteurs qui rendent le véhicule complètement autonome des autres pour l'acquisition de l'information, la communication est basée sur un principe d'échange d'informations entre les voitures. Dès lors, plus le taux de pénétration est faible, moins les systèmes basés sur la communication sont efficaces.

Un autre problème réside dans le choix des informations qu'il convient de transmettre et à quel moment il faut les transmettre. D'un point de vue contrôle intelligent, la voiture évolue dans un environnement et doit prendre des décisions de conduite en suivant, autant que possible, une politique d'actions optimale. La communication vient améliorer la perception du véhicule face à son environnement. Il faut donc choisir la meilleure façon d'utiliser la communication pour récolter le plus d'informations possible. La grande difficulté vient du nombre pharamineux de facteurs et de possibilités à choisir et ajuster pour optimiser la quantité d'information reçue et sa fiabilité. Par exemple, est-ce que l'on transmet la position d'un véhicule dix fois par seconde ou l'on transmet sa position, sa vitesse et sa direction une fois par seconde? On peut aussi spécifier avec certitude à l'agent qu'il y a ou non un véhicule à tel endroit ou encore lui spécifier un pourcentage de fiabilité aux informations. Il serait aussi possible de combi-

ner les mécanismes de communication et les capteurs pour se faire une idée plus précise des objets qui entourent la voiture et choisir ensuite la bonne stratégie à adopter. Par exemple, un capteur détecte une voiture à l'avant : si cette voiture est communicante, on pourrait modifier la stratégie du ACC sachant que l'on sera plus rapidement averti en cas de freinage.

L'implantation de la communication dans les voitures présente donc plusieurs défis très intéressants et une façon de traiter le problème serait d'utiliser l'apprentissage par renforcement pour établir une politique de communication optimale. Cette technique consiste à faire évoluer l'agent dans son environnement et à lui donner des récompenses selon les actions qu'il pose et ce qu'il observe. Par exemple, s'il transmet trop de messages à la seconde et que le réseau devient surchargé, il obtient une récompense négative. Il en est de même s'il ne transmet pas sa position et qu'un autre véhicule entre en collision avec lui. La principale difficulté qu'apporte cette technique vient du choix d'action pratiquement continu et de l'évaluation des récompenses à donner vu la complexité de l'environnement.

Un autre problème traité dans ce mémoire est la diminution de la congestion urbaine par l'optimisation des feux de signalisation. Le réseau routier est constitué de nombreuses intersections qui représentent des goulots d'étranglement pour le flux de trafic. Ces intersections sont généralement équipées de feux de signalisation pour gérer la circulation de façon sécuritaire. Cependant, ces feux ne sont pas toujours efficaces face à la demande croissante des automobilistes qui empruntent le réseau routier et ils tiennent rarement compte des changements dynamiques dans le flux du trafic. Une solution est d'utiliser la communication route-à-véhicule pour acquérir plus d'information sur l'état du trafic et modifier le fonctionnement des feux en conséquence.

1.2 Motivations

La principale motivation sous-tendant la communication inter-véhicules est l'apport que celle-ci peut avoir dans les systèmes de transport intelligents. De nombreuses études sont présentement faites dans ce domaine ce qui démontre bien son importance. Les applications qui profiteraient d'une telle technologie sont nombreuses et touchent de multiples aspects, de la sécurité au divertissement. De plus, il semble que les outils informatiques issus de l'intelligence artificielle et plus particulièrement de l'apprentissage machine sont adéquats au domaine des systèmes de transport intelligents.

1.3 Objectifs

Les objectifs de cette maîtrise sont les suivants :

- Explorer les techniques d'apprentissages machine et évaluer leurs applications possibles à un domaine concret qu'est celui des systèmes de transport intelligents ;
- Proposer une solution de communication pour les systèmes d'aide à la conduite en conformité avec l'état de l'art des recherches en communication inter-véhicules ;
- Proposer des solutions au problème de la congestion par l'optimisation des feux de signalisation dans le contexte de la communication entre la route et les véhicules.

1.4 AUTO21

Cette maîtrise a été faite dans le cadre d'AUTO21, une initiative de recherche nationale appuyée par le Gouvernement du Canada par l'intermédiaire de la Direction des Réseaux de centres d'excellence et par plus de 110 partenaires industriels, gouvernementaux et institutionnels [AUTO21, 2008].

Le réseau AUTO21 a été mis sur pied dans le but de concentrer les connaissances et l'expérience des chercheurs canadiens pour améliorer et accentuer la compétitivité globale de l'industrie canadienne de l'automobile. Le réseau assiste présentement plus de 265 chercheurs dans 42 établissements universitaires, installations de recherche gouvernementales et laboratoires du secteur privé, d'un bout à l'autre du Canada.

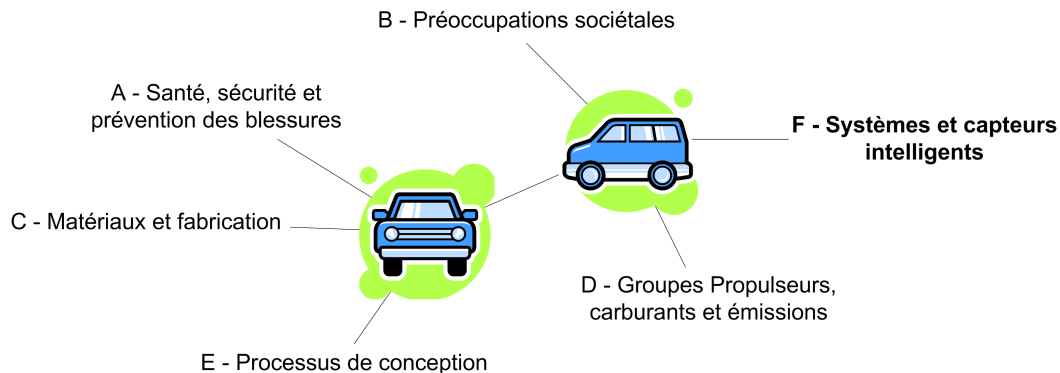


FIG. 1.1 – Les six thèmes d'AUTO21.

Les projets supportés par AUTO21 sont divisés en six thèmes (figure 1.1) : la sécurité, les préoccupations sociétales, les matériaux, les carburants, les processus de

conception et les systèmes intelligents. Notre projet, « Communications des véhicules et applications », fait parti du dernier thème, soit « Systèmes et capteurs intelligents ». Il est dirigé par Dr Brahim Chaib-draa de l'Université Laval et par Dr Soumaya Cherkaoui de l'Université de Sherbrooke. Le but de ce projet est de mettre au point une infrastructure de communication pour partager les données fournies par les capteurs des véhicules, soit avec d'autres véhicules ou avec des dispositifs situés le long de la route. Dans ce projet, il est spécifié que « le résultat serait une efficacité accrue qui réduirait le stress des conducteurs, minimiserait la pollution et apporterait des avantages économiques »[AUTO21, 2008]. Le projet est divisé en trois étapes :

1. Le développement d'une infrastructure de communication à faible coût et sa mise à l'essai au moyen d'une infrastructure de télécommunication.
2. La conception de stratégies de contrôle coopératif par simulation de l'infrastructure de communication dans un simulateur qui utilise des modèles précis de véhicules.
3. L'intégration des infrastructures de communication et des stratégies de collaboration précédemment testées dans le simulateur pour valider certaines applications dans des scénarios de trafic réalistes, comme pour contrôler des véhicules automatisés et donner des conseils de conduite au conducteur.

Les travaux faits à l'Université Laval au sein de notre groupe de recherche portent principalement sur la deuxième étape.

1.5 Organisation du mémoire

Le mémoire est composé de 5 chapitres excluant l'introduction et la conclusion. Le premier chapitre fait un survol des deux principaux domaines liés à ce mémoire, à savoir l'intelligence artificielle et les systèmes de transport intelligents. Le deuxième chapitre fait le tour des communications sans fil. On y explique d'abord les notions de base de la communication entre ordinateurs puis les principes de la communication sans fil jusqu'aux réseaux mobiles. Le troisième chapitre explique les recherches effectuées au cours de cette maîtrise sur la communication inter-véhicules. On met plus spécifiquement l'accent sur l'approche utilisée ainsi que les résultats obtenus.

Les deux derniers chapitres sont sur le deuxième aspect de cette maîtrise, c'est-à-dire sur l'optimisation des feux de signalisation grâce à la communication route-à-véhicule. Le quatrième chapitre expose le problème de la congestion et explique les

principales causes et solutions existantes à ce problème. Le cinquième chapitre porte sur l'optimisation des feux de signalisation et sur les approches testées au cours de cette maîtrise.

Chapitre 2

L'intelligence artificielle et les systèmes de transport intelligents

L'intelligence artificielle et les systèmes de transport intelligents sont deux domaines qui animent la curiosité de plusieurs d'entre nous. Plusieurs films dont l'action se passe dans le futur ont alimenté notre imagination sur l'évolution technologique que nous pourrions observer au cours des prochaines décennies. Ce chapitre fait un tour d'horizon de ces deux champs de recherche et du lien qui existe entre les deux.

2.1 La robotique

Notre monde a connu plusieurs révolutions depuis le début de l'histoire. Les plus récentes sont les révolutions industrielles qui marquent une grande évolution dans le contrôle de l'énergie, dans le transport et dans la transformation des matières premières [Wikipédia, 2008b]. La figure 2.1 montre les principaux apports de chacune de ces révolutions industrielles au cours des derniers siècles. La troisième, la révolution informatique, commença vers les années 1970 avec l'avènement des microprocesseurs et des ordinateurs de bureau. Depuis ce temps, l'informatique n'a pas cessé d'évoluer de manière très rapide. Deux domaines ont évolué en parallèle : les ordinateurs (rapidité, composantes, etc) et la réseautique. De plus, l'expansion rapide et massive d'Internet au cours des années 1990 et 2000 a complètement changé notre monde d'un bout à l'autre de la planète. En effet, il n'a jamais été aussi facile de communiquer et de partager de l'information entre les différents pays de la terre. Cette étape de notre évolution restera certainement parmi les plus marquantes de l'histoire.

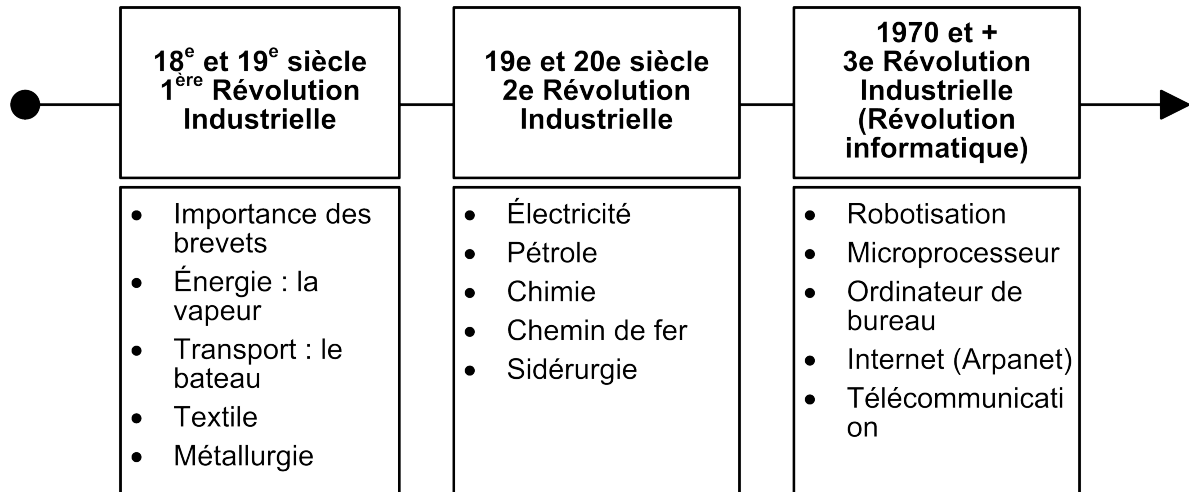


FIG. 2.1 – Les trois révolutions industrielles.

De plus, à voir la vitesse à laquelle évoluent les technologies, il est intéressant de se pencher sur les innovations à venir dans ce domaine. Depuis toujours, l'être humain a cherché à optimiser son travail, à faire plus avec moins. Que ce soit avec des techniques simples transmises de génération en génération ou avec l'invention d'outils spécialisés à certaines tâches, nous avons toujours amélioré notre façon de faire les choses pour gagner du temps et nous consacrer ainsi à d'autres domaines. Avec l'arrivée des moteurs et notre capacité à créer et gérer de l'énergie mécanique, l'automatisation a rapidement envahi le domaine industriel. De pair avec l'informatique, la mécanique s'est améliorée et s'est transformée ce qui a fait naître la robotique.

Pour l'instant, les robots sont principalement utilisés pour automatiser des tâches au niveau commercial. On en retrouve parfois dans nos maisons, mais généralement sous forme de jouets. Cependant, nous commençons à voir apparaître sur le marché divers produits robotisés permettant de nous aider dans notre train-train quotidien. Au cours des prochaines années, nous verrons probablement la démocratisation massive de ces petits robots ayant pour but de nous faciliter les tâches ménagères et d'entretien. On peut déjà trouver sur le marché, à des prix relativement modiques, des robots aspirateurs autonomes, des robots capables de laver les planchers et même des robots pour nettoyer les gouttières.

Trois facteurs vont certainement permettre l'expansion de ce domaine : la miniaturisation et le perfectionnement des pièces informatiques (processeurs, capteurs, etc), la diminution des prix et l'amélioration des techniques d'intelligence artificielles pour contrôler les robots. Ainsi, les appareils électriques et mécaniques qui nous entourent ne seront plus seulement des outils pour accélérer le travail, mais des entités autonomes

ou semi-autonomes capables de faire le travail à notre place. Par exemple, l'aspirateur conventionnel est très pratique pour faire le ménage, mais c'est encore plus pratique de ne pas avoir à le faire si un robot aspirateur s'en charge à notre place. Cependant, l'intelligence artificielle n'est pas seulement couplée à la robotique, il existe aussi des logiciels « intelligents » ou simplement du code « intelligent ».

2.2 L'intelligence artificielle et les agents

Le domaine de recherche de l'Intelligence Artificielle (IA) a comme but de créer des entités intelligentes, que ces entités soient physiques ou non (matérielles ou logicielles). Les définitions mêmes de l'intelligence artificielle varient d'un auteur à l'autre et peuvent être séparées en deux catégories : celles qui mettent l'accent sur le processus de raisonnement et celles qui mettent l'accent sur le comportement [Russell et Norvig, 2003]. En gros, certains diront d'un système qu'il est intelligent s'il agit de façon rationnelle alors que d'autres diront qu'il est intelligent s'il pense de façon rationnelle.

Une notion importante en intelligence artificielle est la notion d'agent et son extension aux systèmes multi-agents. À la base, un agent se définit simplement comme quelque chose qui agit. Cependant, en informatique, un agent se distingue d'un programme par ses attributs supplémentaires comme le fait « d'agir de façon autonome, de percevoir son environnement, d'être persistant pendant de longues périodes et de s'adapter aux changements » [Russell et Norvig, 2003]. Pour sa part, un agent rationnel agit toujours de façon à obtenir le « meilleur résultat ». Ces définitions peuvent sembler abstraites, mais l'important est de savoir que le but des agents est d'agir le mieux possible selon la situation et le but souhaité.

Les agents ou systèmes intelligents prendront de plus en plus de place dans nos vies à mesure que les domaines de l'informatique, de l'électronique et de la mécanique se perfectionneront. Par exemple, la miniaturisation et l'amélioration des caméras couplées à la recherche sur la reconnaissance visuelle ouvrent la porte à une foule de systèmes évolués : on peut penser à un système de détection d'intrusion non seulement capable de détecter les mouvements, mais aussi de reconnaître qui est dans la pièce ou encore de prendre des photos très nettes du visage pour les envoyer à une centrale de sécurité. Un autre exemple de système intelligent serait un téléphone mobile GPS se mettant en mode vibration si vous êtes au bureau et qu'une réunion est prévue à votre agenda. Ces deux exemples démontrent bien que plusieurs systèmes intelligents ne reposent pas seulement sur la recherche en informatique, mais aussi sur les domaines plus physiques comme l'électronique. En outre, il ne suffit pas d'être capable de choisir des actions,

il faut aussi être capable de bien cerner son environnement, par exemple à l'aide de capteurs tels qu'une caméra ou un GPS.

L'architecture de base d'un agent peut se résumer à la figure 2.2. L'agent perçoit son environnement à l'aide de capteurs, choisit la ou les actions à faire selon sa fonction interne et exécute ces actions dans l'environnement à l'aide de ses effecteurs. La fonction interne qui détermine les actions varie selon le type d'agent et peut tenir compte du percept courant ou encore d'un historique de percepts. Par exemple, le type d'agent le plus simple est l'agent réflexe simple. Il sélectionne des actions en fonction du percept courant sans tenir compte du reste de l'historique des percepts. Son fonctionnement est basé sur des règles simples sous la forme « **si ... alors ...** » (ex : *si piéton traverse la rue alors commence à freiner*). Cependant, certains types d'agent sont beaucoup plus complexes et certains sont capables d'apprentissage. Nous verrons plus en détail au chapitre 4 les agents capables d'apprentissage.

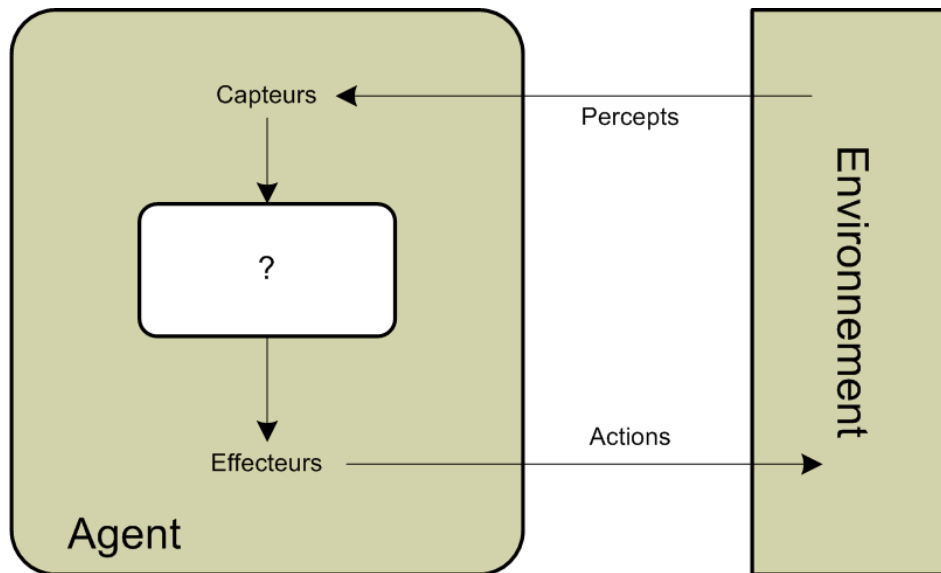


FIG. 2.2 – Interaction d'un agent avec son environnement.

Ainsi, un agent perçoit son environnement à l'aide de capteurs, pose des actions à l'aide de ses effecteurs et l'on évalue son efficacité à l'aide d'une mesure de performance. L'ensemble de ces éléments forme l'environnement de la tâche. Lors de la conception d'un agent, il faut toujours commencer par spécifier l'environnement de la tâche. Le tableau 2.1 donne quelques exemples.

| Type d'agent | Mesure de performance | Environnement | Effecteurs | Capteurs |
|-----------------------|-----------------------------|---|-------------------------------------|--|
| Conducteur automobile | Sécurité, rapidité, confort | Réseaux routiers, autres véhicules, piétons | Volant, accélérateur, frein | Caméras, détecteurs lasers, détecteurs infrarouges |
| Robot aspirateur | Propreté, rapidité | Planchers, meubles, obstacles | Roues motorisées, aspirateur | Détecteurs de poussières, détecteurs d'obstacles |
| Joueur échec | Victoires, nombre de pièces | Échiquier | Écran pour afficher le coup à jouer | Caméra pour observer le jeu |

TAB. 2.1 – Exemple d'environnements de tâche pour différents agents.

2.3 Concepts d'IA

Pour avoir une bonne idée d'où se situent les recherches de ce mémoire en matière d'intelligence artificielle, il est bon d'avoir une description sommaire des différents champs de recherches de ce domaine. Le livre de Russell et Norvig [Russell et Norvig, 2003] présente un excellent portrait très structuré de l'IA. Les principaux concepts sont expliqués dans cette section.

2.3.1 Résolution de problèmes, arbres de recherche

Un des exemples les plus communs lorsque l'on pense à l'intelligence artificielle est un ordinateur capable de jouer aux échecs [Wall, 2008]. Ce type d'agent est un agent de résolution de problèmes : il prend un problème et un but donné, calcule la suite d'action à faire pour atteindre le but puis exécute cette suite d'action. Un bon exemple de problème pour ce type d'agent est la recherche du plus court chemin dans un graphe ou encore la navigation d'un robot dans une pièce. Ce type de problème est communément résolu en utilisant un arbre de recherche : le premier noeud représente l'état initial, les noeuds suivants représentent l'état après avoir effectué chacune des actions possibles, et ainsi de suite. On utilise ensuite cet arbre pour choisir l'action qui a le plus de chance de nous mener à l'état but, c'est à dire la victoire dans le cas d'un jeu, ou encore la suite d'actions qui nous mènera le plus rapidement possible au but dans le cas de la recherche du plus court chemin. Pour le cas des échecs, cela

revient à calculer tous les états possibles du jeu après un coup, puis après un deuxième et troisième coup, etc. Après seulement trois coups, le nombre de possibilités est de 8902. Cependant, les problèmes ne sont pas tous aussi complexes que les échecs et il existe plusieurs algorithmes pour accélérer la recherche d'une solution dans l'arbre ou le graphe.

Pour évaluer la valeur d'un état quelconque, à savoir s'il est prêt du but ou non, il faut utiliser une fonction heuristique. Cette fonction est dépendante du problème et n'est pas la même d'un problème à l'autre. Par exemple, dans le cas du plus court chemin, la fonction d'heuristique pourrait calculer la distance à vol d'oiseau entre l'état et le but. De cette façon, l'agent pourrait commencer par explorer le noeud qui est le plus près à vol d'oiseau de la destination en supposant qu'il est plus probable que la solution passe par ce noeud. La qualité de la fonction heuristique est donc très importante pour permettre d'accélérer la recherche dans le graphe.

2.3.2 Problèmes de satisfaction de contraintes

Certains problèmes peuvent cependant être représentés de façon simple et conforme à un certain standard permettant d'utiliser des heuristiques générales plutôt que spécifiques au problème. Ce sont les problèmes de satisfaction de contraintes et sont définis par un ensemble de variables pouvant prendre certaines valeurs et un ensemble de contraintes sur ces variables. Un bon exemple est le jeu *Sudoku* : il faut assigner un chiffre à chaque case vide et ce chiffre ne doit pas se répéter dans la ligne, la colonne et le bloc. L'arbre de recherche peut être construit en assignant une valeur à chaque case pour chaque noeud de l'arbre. Le premier niveau représente toutes les valeurs possibles de la première case, le deuxième niveau représente les valeurs possibles pour la deuxième case, et ainsi de suite. On peut ensuite utiliser l'algorithme de *backtracking* pour résoudre un tel problème : on parcourt l'arbre en assignant une valeur à la première variable, puis à la deuxième variable, à la troisième, etc., jusqu'à ce qu'il n'y ait plus de valeurs légales pour une variable. À ce point, on sait qu'une des assignations précédentes était mauvaise. On remonte donc d'un niveau et on essaye une autre valeur pour ce niveau précédent. Le problème est résolu lorsque l'on réussit à atteindre la dernière variable et à lui assigner une valeur. Bien sûr, cette méthode ne semble pas très optimisée pour résoudre rapidement le problème. Elle assure cependant de trouver la solution et il existe d'autres méthodes plus performantes pour accélérer le traitement.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | 1 | | | | | | |
| | | 2 | | 3 | | | | 4 |
| | | | 5 | | | 6 | | 7 |
| 5 | | | 1 | 4 | | | | |
| | 7 | | | | | | 2 | |
| | | | | 7 | 8 | | | 9 |
| 8 | | 7 | | | 9 | | | |
| 4 | | | | 6 | | 3 | | |
| | | | | | | 5 | | |

FIG. 2.3 – Exemple du jeu de Sudoku. Il faut placer des nombres dans chaque case en respectant les contraintes.

2.3.3 Apprentissage machine

Dans le cas de l'apprentissage machine, un agent apprend à résoudre un problème en observant l'effet de ses actions dans un environnement. Autrement dit, plutôt que de calculer la suite d'actions à faire pour arriver à un but donné, l'agent exécute des actions dans l'environnement, observe les résultats de ses actions et mémorise l'effet qu'elles ont sur l'environnement. Par la suite, lorsque l'agent veut atteindre un but spécifique, il utilise les connaissances qu'il a apprises pour choisir les actions qui le mèneront à ce but donné.

De façon générale, les problèmes d'apprentissage se divisent en trois catégories : l'apprentissage supervisé, l'apprentissage non supervisé et l'apprentissage par renforcement. Le problème d'apprentissage supervisé se résume à apprendre une fonction en observant des exemples d'entrées et de sorties de cette fonction. Le problème d'apprentissage non supervisé consiste à discerner des motifs dans l'entrée quand aucune valeur de sortie spécifique n'est donnée. Finalement, dans le cas de l'apprentissage par renforcement, l'agent apprend à l'aide de récompenses plutôt que de se faire dire quoi faire par un superviseur. L'apprentissage machine, et plus particulièrement l'apprentissage par renforcement, sera vu plus en détail au chapitre 4.

2.3.4 Théorie des jeux

Les arbres de recherche et la représentation du problème permettent à un agent d'appliquer un algorithme pour atteindre un but spécifique, qui est la solution à son problème. Cependant, il arrive parfois que plusieurs agents travaillent sur le même problème ou plutôt dans le même environnement. Il s'agit alors d'environnements multi-

agents. Les agents peuvent avoir le même but et être en coopération ou encore avoir des buts complètement différents et être en compétition. On parle alors de jeux. En économie, la théorie des jeux étudie les environnements où les actions de plusieurs agents ont un impact sur les autres agents. En IA, le terme jeux est appliqué de façon plus restrictive sur les environnements déterministes, à deux joueurs, tour-à-tour, complètement observables et à somme nulle. Un jeux à somme nulle signifie que lorsqu'un agent gagne, l'autre perd. Il existe plusieurs algorithmes qu'un agent peut utiliser pour résoudre ce problème, certains étant plus performants et plus rapides que d'autres. Cependant, l'efficacité d'un algorithme dépendra surtout de la stratégie adoptée par l'autre joueur.

2.3.5 Autres

Il existe plusieurs autres champs de recherche en intelligence artificielle, mais ceux décrits précédemment sont les plus importants au niveau de ce mémoire. Parmi les autres notions importantes de l'IA, on retrouve :

- La représentation des connaissances ;
- Le raisonnement et la logique ;
- La planification ;
- Le raisonnement sous incertain ;
- Language naturel et perceptions.

Le domaine qui nous intéresse le plus dans le cas de ce mémoire est l'apprentissage par renforcement, lequel sera expliqué plus en détail au chapitre 4.

2.4 Les systèmes de transport intelligents

Un des domaines très en vogue pour l'application de l'intelligence artificielle est le domaine des transports. Depuis l'invention des premières automobiles, l'industrie n'a pas cessé de les améliorer et de les transformer pour répondre aux besoins des consommateurs. Que ce soit les essuie-glace ou encore le tout dernier système d'aide au stationnement en passant par les chargeurs à douze disques compacts, les constructeurs innent sans arrêt pour rendre nos véhicules plus confortables et plus sécuritaires. Encore là, l'informatique a révolutionné le domaine de l'automobile et y joue maintenant un rôle primordial. Une trousse à outils ne suffit plus pour réparer une voiture, il faut maintenant de l'équipement électronique pour se brancher et diagnostiquer l'ordinateur

de bord.

Les Systèmes de Transport Intelligents (STI, ou « Intelligent Transport System », ITS) perçoivent l'environnement à l'aide des capteurs et fournissent de l'information ou agissent sur le contrôle du véhicule pour aider le conducteur dans la tâche de conduite [Bishop, 2005]. Les STI sont la prochaine génération de système d'aide à la conduite par rapport à ceux qui existent présentement et qui ne fournissent qu'une assistance de base au contrôle sans percevoir l'environnement et sans évaluer le risque. Cependant, il est difficile de tracer une ligne parfaite entre les deux générations. Retenons simplement que l'avancé majeure de ces systèmes réside dans l'acquisition et le traitement d'informations supplémentaires sur l'environnement.

Voici quelques exemples de systèmes d'aide à la conduite :

- Direction assistée
- Changement de rapport automatique
- ABS antiblocage des roues
- ESP contrôle de trajectoire
- Régulateur de vitesse
- Aide à la navigation par GPS
- Avertisseur de déviation de trajectoire
- Alerte de franchissement involontaire de ligne (AFIL)
- Allumage automatique des feux de croisement
- Détecteur de pluie (activation automatique des essuies-glaces)
- Radar de recul (détection de distance arrière) parfois associé à une détection des obstacles à l'avant.

En regardant la définition des STI, on remarque qu'elle se colle particulièrement bien à celle des agents : il y a perception de l'environnement, prise de décision, etc. Les domaines des STI et de l'IA sont donc étroitement liés et il semble naturel d'appliquer la théorie multi-agents pour la conception de systèmes de transport intelligents.

De plus, il est évident que l'évolution de ces systèmes passe par l'amélioration des techniques d'acquisition de l'information sur l'environnement. Les communications inter-véhicules et route-à-véhicule sont des solutions idéales puisqu'elles permettent d'échanger une foule d'informations précises sur le comportement des véhicules. Le prochain chapitre en discute plus en détail.

2.5 Conclusion

Dans ce chapitre, nous avons vu des exemples concrets de l'application de l'intelligence artificielle à des problèmes de la vie courante. Nous avons vu qu'il existe différentes modélisations et différents champs de recherche en IA dépendamment du type de problème. Nous avons aussi vu le fonctionnement sommaire des agents et comment ceux-ci peuvent être intégrés au domaine des transports par l'entremise des systèmes de transport intelligents.

Le prochain chapitre traite de la communication entre ordinateurs et plus particulièrement de la communication sans fil. Nous verrons les principales notions de cette technologie ainsi que l'état de l'art sur la communication dans le domaine des transports.

Chapitre 3

Notions de communications et de communication sans fil

Cette maîtrise vise, en premier lieu, à voir de plus près si la communication dans les transports peut aider à résoudre certains problèmes dans ce domaine. La communication est vue ici comme tout échange d'informations entre les entités formant le réseau routier et les entités empruntant ce réseau. Il peut s'agir de feux de signalisation, de véhicules ou simplement de bornes placées sur le bord de la route. Pour communiquer, les ordinateurs de ces entités doivent utiliser un ou plusieurs protocoles ainsi qu'un ou plusieurs médias de transmission. Plusieurs recherches ont été faites dans le domaine des réseaux sans fil pour trouver les meilleurs protocoles et technologies à utiliser pour le domaine des systèmes de transport intelligents.

Ce chapitre se divise en 3 sections. La première fait le tour de la communication entre ordinateurs et des principes de base de la communication en informatique. La deuxième section porte sur les communications sans fil. Finalement, la troisième section traite des réseaux mobiles et de la communication inter-véhicules.

3.1 Communication entre ordinateurs

La communication inter-véhicules est un domaine complexe qui se subdivise en plusieurs champs de recherche : les ondes radio, les protocoles de routage, les types de message, etc. Pour bien comprendre les enjeux et les notions telles que le standard DSRC [[Armstrong, 2005](#)] ou WAVE [[IEEE, 2006](#)] décrits à la section [3.3](#), il est impor-

tant de revenir sur les notions de communication entre ordinateurs, c'est-à-dire sur les réseaux d'ordinateurs. Cette section explique le principe de couches, d'interfaces, de services et les architectures connues telles que le modèle OSI.

3.1.1 Architectures réseau

Commençons par les principes de bases des architectures réseau. Afin de réduire la complexité de la conception, la plupart des réseaux sont organisés en strates, appelées couches ou niveaux, chacune étant placée au-dessus de la précédente [Tanenbaum, 2003]. Le rôle de chaque couche est de fournir des services à la couche immédiatement supérieure tout en lui dissimulant les détails d'implémentation. La couche n d'une machine dialogue avec la couche n d'une autre machine. Les règles et les conventions qui gouvernent cette communication sont groupées collectivement sous le nom de protocole de couche n . Essentiellement, un protocole est une convention acceptée par les parties communicantes sur la façon dont leur dialogue doit prendre place. L'ensemble des protocoles utilisés par un certain système, avec un protocole par couche, est appelé pile de protocoles. Cependant, on utilise souvent le terme protocole pour désigner une pile de protocoles.

La figure 3.1 montre le lien entre les couches, les protocoles et les interfaces pour la communication entre deux hôtes dans un réseau à 5 couches. Chaque couche communique avec son homologue de la machine distante en utilisant les services de la couche inférieure. En fait, il s'agit de communication virtuelle puisqu'aucune donnée n'est directement transmise de la couche n d'une machine à la couche n de l'autre machine. L'échange de données se fait uniquement sur le support physique utilisé par la couche inférieure de chaque machine. Ce support physique peut être par exemple des câbles ou encore l'espace pour la transmission sans fil. Un exemple de communication entre deux couches n de machines distantes pourrait être de l'échange d'information entre un serveur web et un navigateur web utilisant tous les deux le protocole HTTP.

Les deux architectures de réseaux les plus connues sont sans doute les modèles de référence OSI (pour « Open Systems Interconnection ») et TCP/IP (pour les deux principaux protocoles de ce modèle). Le premier est une spécification plutôt théorique basée sur une structure à 7 couches alors que le second est un modèle à 5 couches dont les protocoles sont largement utilisés en pratique. Les 7 couches du modèle OSI sont présentées dans la Figure 3.2. Les trois premières couches n'interviennent qu'entre machines voisines et non entre les machines d'extrémités qui peuvent être séparées par plusieurs routeurs. Ces couches agissent donc dans la frontière du réseau. Les quatre autres couches n'interviennent au contraire qu'entre les hôtes distants et sont peu touchées par

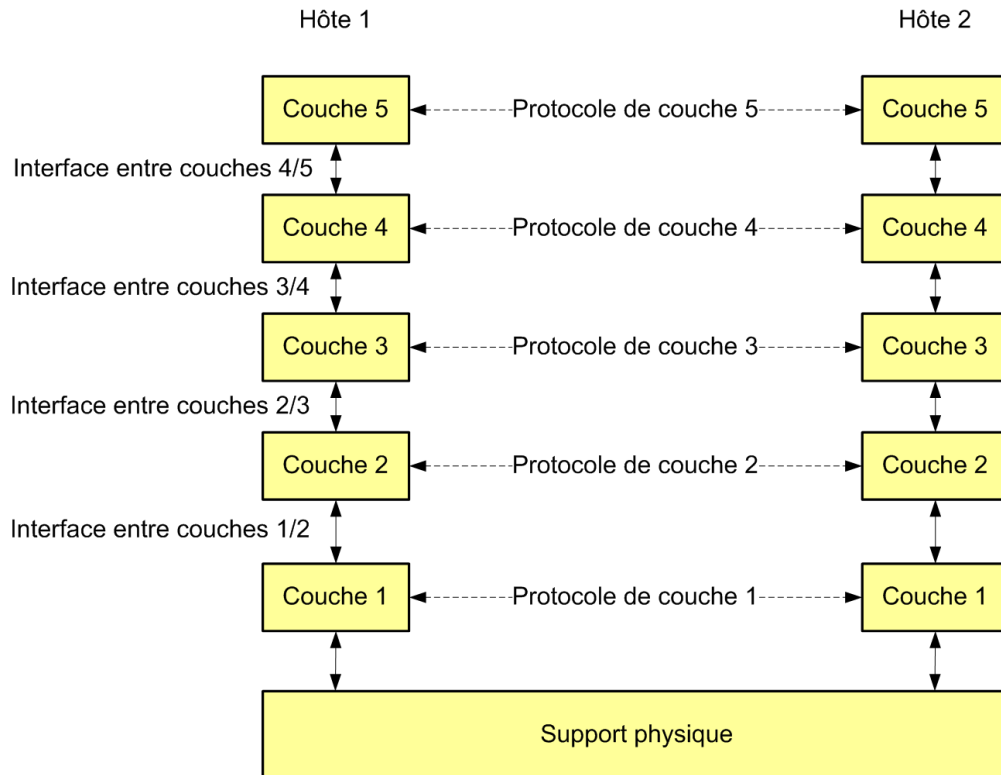


FIG. 3.1 – Les couches, les protocoles et les interfaces.

les caractéristiques physiques du réseau.

3.1.2 Les couches sous-réseau

La couche 1 est la couche physique. Elle est responsable de la conversion des bits en signal électrique. Son service est typiquement limité à l'émission et la réception d'un bit ou d'un train de bits continu.

Le rôle principal de la deuxième couche, la couche de liaison, est d'offrir un moyen de communication brut et fiable à la couche réseau. Elle divise les données en trames de données et envoie des trames d'acquittement pour s'assurer que la communication est exempte d'erreurs de transmission. Tout comme les couches supérieures, elle doit aussi gérer les écarts entre la vitesse d'émission et de réception en utilisant des fonctions de régulation pour éviter qu'un récepteur soit submergé. De plus, elle doit synchroniser l'accès au support physique lorsque celui-ci est partagé.

Bien que cela ne fasse pas partie du modèle OSI, la couche liaison est généralement

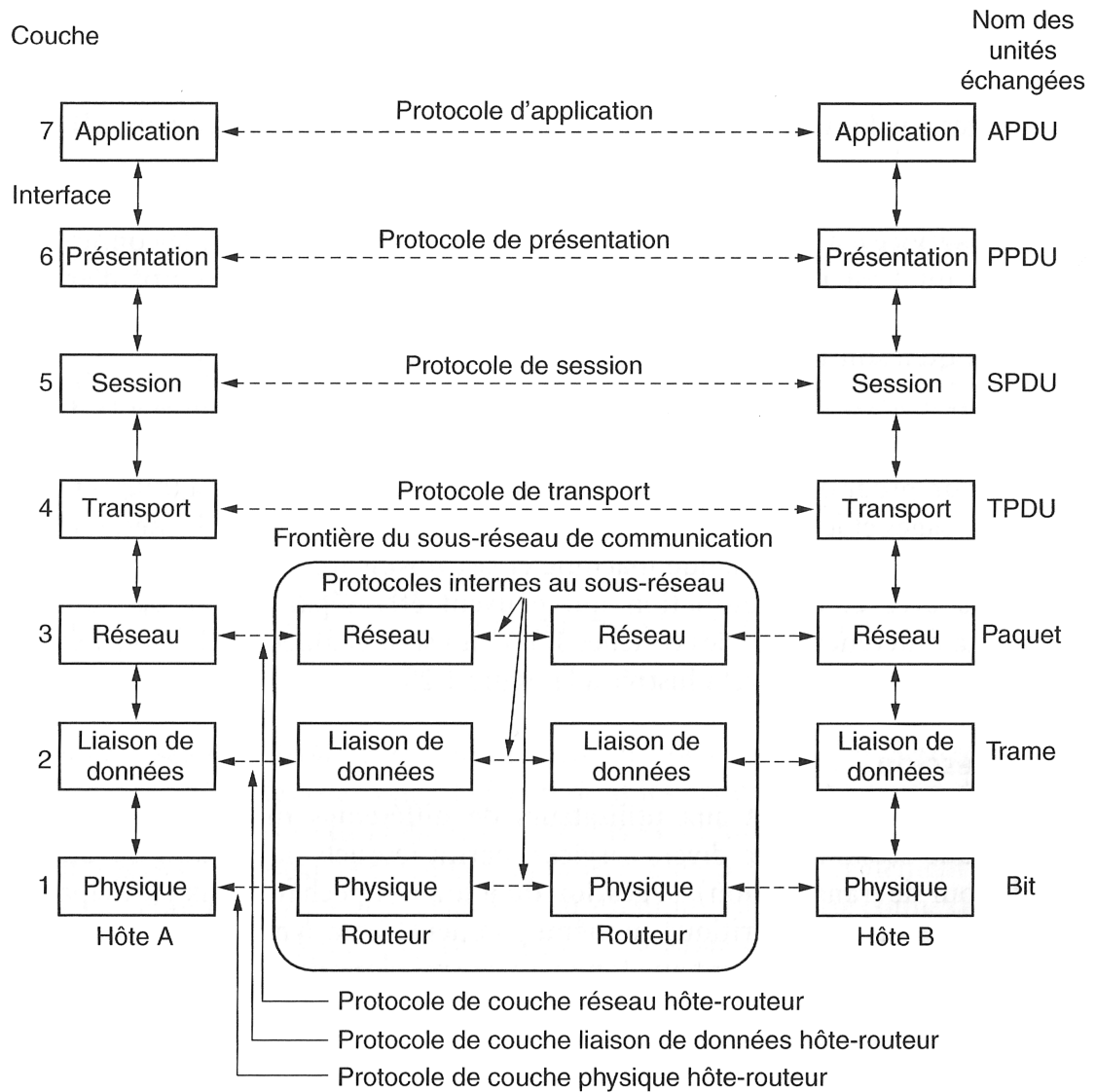


FIG. 3.2 – Le modèle de référence OSI [Tanenbaum, 2003].

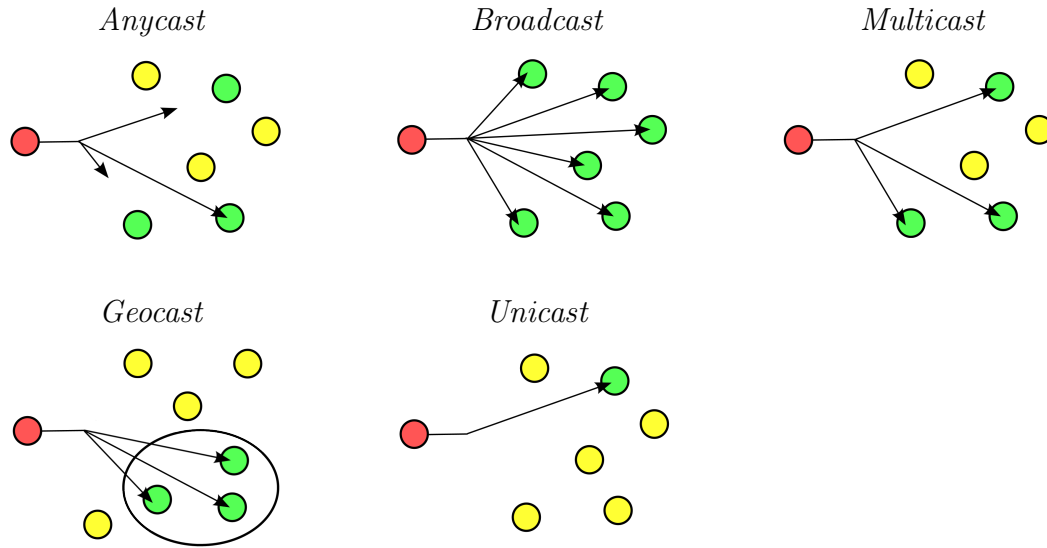
divisée en deux sous-couches. La couche LLC (*Logical Link Control*) est la sous-couche supérieure et regroupe toutes les fonctions de contrôle et de gestion d'erreur. La deuxième sous-couche est la couche MAC (*Media Access Control*) et elle gère l'accès au support physique. Dans le cas d'un réseau sans support partagé, elle ne fait généralement que de la détection d'erreur. En revanche, dans le cas où le support est partagé comme pour les communications sans fil, elle devient extrêmement complexe et cruciale pour les performances du réseau. C'est pourquoi il y a tant de recherches qui portent sur les protocoles de cette couche dans le domaine de la communication inter-véhicules.

La troisième couche est la couche réseau. Elle est responsable du fonctionnement du sous-réseau et principalement de la façon dont les paquets sont routés. Les routes peuvent être dynamiques selon la charge du réseau ou encore statiques. Si trop de données transigent par le même endroit, le réseau devient congestionné. Cette couche est donc responsable de gérer ces types de problèmes et d'assurer une certaine qualité de service (délais, temps de transit, etc.). De plus, elle est responsable de faire fonctionner ensemble des sous-réseaux hétérogènes.

Le protocole de routage de la couche réseau dépend énormément du modèle de diffusion utilisé. Le tableau 3.1 représente différents modèles de diffusion possibles. Le premier est le modèle *anycast* et permet de rediriger les données vers le récepteur le « plus proche » ou le « plus efficace » selon le protocole de routage. Il est entre autres utilisé pour fournir des services DNS (*Domain Name System*). Le deuxième modèle, le *broadcast*, envoie l'information à tous les noeuds du réseau ou du sous-réseau. Pour sa part, le modèle *multicast* ressemble beaucoup au modèle *broadcast* à l'exception que les messages sont adressés à un groupe d'ordinateurs spécifiques. L'avantage d'un tel protocole est que les données ne sont transmises qu'une seule fois et seront routées vers toutes les machines du groupe de diffusion. Cela est particulièrement utile dans le cas de la diffusion vidéo. Le quatrième modèle, le *geocast*, peut être vu comme une sous-catégorie du *multicast* où le groupe de diffusion est choisi selon la position géographique des noeuds. C'est une méthode très étudiée pour la communication inter-véhicules puisque l'environnement est typiquement propice à ce mode de diffusion. Le dernier modèle est la méthode de diffusion classique *unicast* : les données sont transmises d'un ordinateur à un autre uniquement. Dans ce dernier cas, on parle de diffusion point-à-point plutôt que de diffusion multipoints comme le *broadcast*, le *multicast* ou le *geocast*.

3.1.3 Les couches hôtes

Les couches présentées dans la sous-section précédente (la couche physique, la couche de liaison et la couche réseau) sont donc les trois premières couches du modèle OSI.



TAB. 3.1 – Modèles de diffusion [Wikipédia, 2008a].

L'implémentation de ces couches se fait généralement dans les périphériques et les appareils réseau tels que les routeurs et les cartes réseau. Les quatre couches restantes sont la couche transport, la couche session, la couche présentation et la couche application.

La couche transport détermine le type de service offert à la couche session. Par exemple, elle peut offrir une connexion point-à-point exempte d'erreur où l'ordre d'arrivée des messages est le même que l'ordre d'envoi. Il peut aussi s'agir d'une connexion où la remise des messages est isolée et où l'ordre n'a pas d'importance, mais où la vitesse est priorisée. Par exemple, lorsque l'on transmet un courriel, on ne veut pas qu'il manque des mots ou des lettres, et on veut que le message atteigne sa destination en parfait état, même si cela peut prendre plus de temps à transmettre. Cependant, lors d'une conférence vidéo, ce n'est pas très important s'il manque une image de temps en temps. Ce que l'on veut, c'est avoir une image fluide en temps réel. Ainsi, même si une image se perd durant la transmission, la machine émettrice ne la retransmettra pas, elle continuera d'envoyer l'image la plus récente.

Il s'agit aussi de la première couche à offrir un service complet de la source à la destination. En d'autres termes, un programme sur la machine source parle directement à un programme similaire sur la machine de destination. Dans les couches précédentes, les messages sont échangés entre les équipements voisins, et non directement entre la source et la destination, qui peuvent être séparées par une multitude de routeurs.

La couche session permet aux différentes machines d'établir des sessions. Elle gère la synchronisation, les interruptions de dialogue et la reprise, etc.

La couche présentation est responsable de la syntaxe des données. Les couches inférieures travaillant plutôt avec des bits, la couche présentation a comme rôle de convertir les données applicatives manipulées par les programmes en chaînes d'octets effectivement transportées par le réseau.

Finalement, la couche application contient une variété de protocoles utiles aux utilisateurs. Par exemple, le protocole HTTP est utilisé pour la navigation de page web alors que le protocole SMTP est utilisé pour la transmission de courriels.

3.2 Communication sans fil

L'arrivée d'ordinateurs et de dispositifs portables a rapidement fait naître le besoin d'être connecté à Internet sans avoir à se brancher par un câble réseau. Divers groupes de travail se sont donc penchés sur cette question et ont choisi comme solution l'approche la plus simple : équiper les ordinateurs portables et les édifices à bureaux d'émetteurs radio à faible portée pour leur permettre de communiquer [Tanenbaum, 2003]. Le comité IEEE qui s'occupait déjà de la normalisation des LAN filaires s'est vu confier la tâche de définir une norme pour les LAN sans fil. C'est ainsi qu'est née la norme 802.11 [IEEE, 2008a], communément appelée WiFi (Wireless Fidelity).

Si l'on se réfère au modèle OSI présenté plus tôt, cette norme s'applique aux deux premières couches, c'est-à-dire à la couche physique et la couche liaison. Il en va de même pour la norme Ethernet (IEEE 802.3) qui régit les LAN filaires. Ainsi, un protocole de couche 3, par exemple le protocole IP, peut utiliser les services de ces deux normes, soit Ethernet ou WiFi, sans que cela change quoi que ce soit pour lui et pour l'utilisateur. C'est un bon exemple du principe de couches et de services des architectures réseau qui assurent une certaine interopérabilité entre différentes technologies.

Comme le montre la figure 3.3, la norme WiFi prévoit deux modes de fonctionnement : avec ou sans station de base. On parle généralement de réseaux avec points d'accès ou bien de réseaux ad hoc pour définir respectivement ces deux modes. Dans le premier cas, les ordinateurs communiquent toujours avec une station de base et ne communiquent jamais entre eux. La station de base peut être, par exemple, un routeur sans fil comme ceux que l'on trouve dans plusieurs maisons ou encore des points d'accès commerciaux présents dans les édifices publics. Dans le cas des réseaux ad hoc, les ordinateurs communiquent directement entre eux sans qu'une station unique régisse les dialogues. Ces deux modes ont chacun une place importante dans la recherche sur la communication inter-véhicules et route-à-véhicule et plusieurs solutions utilisant l'un

ou l'autre de ces modes ont été proposées [Moske *et al.*, 2004; Ernst et Inria, 2002]. Cependant, comme nous verrons plus loin, les réseaux ad hoc sont plus propices aux applications de sécurité alors que les réseaux avec points d'accès sont plus utiles pour des applications d'information générale ou de divertissement.



FIG. 3.3 – Réseau avec point d'accès et Réseau ad hoc.

L'élaboration de la norme 802.11 avait plusieurs défis à surmonter. Il fallait tout d'abord choisir une bande de fréquence qui soit disponible dans une majorité de pays, gérer le fait que la portée de transmission soit limitée, se soucier de la confidentialité des données, tenir compte des implications de la mobilité informatique, etc. De plus, la norme Ethernet était largement répandue et il était primordial que le WiFi soit compatible pour les couches supérieures, autrement dit qu'un paquet IP puisse être transmis sur un LAN sans fil de la même façon que sur Ethernet.

Un autre problème important de la technologie sans fil est le problème de la station cachée (Hidden Node Problem) tel qu'illustré à la figure 3.4 : deux stations peuvent entendre l'activité d'une troisième station mais ne peuvent s'entendre mutuellement. Ce problème est particulièrement courant dans le cas de la communication inter-véhicules puisque les voitures sont constamment en mouvement. Il peut donc en découler une forte dégradation du service dépendamment du protocole utilisé pour la couche liaison [Günter et Großmann, 2005].

Un deuxième problème de taille de la technologie sans fil vient du fait qu'un signal radio ne se propage pas dans un environnement fermé comme dans un câble. Le signal voyage à travers l'espace entourant l'émetteur et peut donc être réfléchi par des objets solides, risquant ainsi d'être reçu plusieurs fois par le récepteur. Ce problème d'interférence, venant du fait que le signal se propage par plusieurs chemins, porte le nom d'affaiblissement par trajets multiples (*multipath fading*).

Pour palier à tous ces problèmes, plusieurs techniques ont été développées, autant au

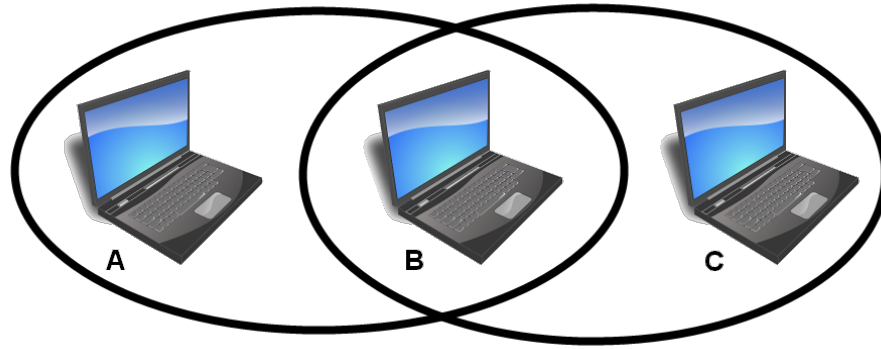


FIG. 3.4 – Problème de la station cachée. A et C communiquent en même temps à B.

niveau de la couche physique que de la couche liaison. Au niveau de la couche physique, il est possible de jouer sur la fréquence utilisée ou encore sur la technique de modulation. Cela détermine la bande passante possible et influence les risques d'interférence et la qualité du signal. Le tableau 3.2 résume les caractéristiques physiques des normes de la famille 802.11 ainsi que les vitesses et portées qui en découlent. Au niveau de la couche liaison, il existe plusieurs techniques d'accès au canal partagé permettant de contrôler les collisions. Par exemple, la norme 802.11 définit la méthode d'accès CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) où les stations se basent sur un système d'accusés de réception pour éviter les collisions.

Bien que les protocoles de la famille IEEE 802.11 soient les plus connus dans le monde informatique, les solutions et architectures proposées pour la communication inter-véhicules reposent généralement sur d'autres protocoles ou sur d'autres bandes de fréquence mieux adaptées aux besoins spécifiques de ces types de réseaux.

3.3 Réseaux mobiles et communication inter-véhicules

Les réseaux mobiles dynamiques se démarquent des réseaux sans fil standards par le fait que les noeuds sont constamment en mouvement et peuvent circuler à très grande vitesse, comme dans le cas des communications inter-véhicules et route-à-véhicule (figure 3.5). Dans de telles circonstances, les protocoles standards comme IEEE 802.11b ou 802.11g ne sont pas toujours très appropriés [Borgonovo *et al.*, 2003]. Plusieurs travaux ont été menés sur le sujet pour proposer des protocoles de communication adaptés aux besoins de la communication inter-véhicules et route-à-véhicule. Cette section présente

| Protocole | Date de sortie | Fréquence d'opération | Débit (max) |
|-----------------|-------------------------|---------------------------------|---------------------------------|
| 802.11 (legacy) | 1997 | 2.4 GHz | 2 Mbits/s |
| 802.11a | 1999 | 5 GHz | 54 Mbits/s |
| 802.11b | 1999 | 2.4 GHz | 11 Mbits/s |
| 802.11g | 2003 | 2.4 GHz | 54 Mbits/s |
| 802.11n | En dév. | 2.4 GHz et 5 GHz | 248 Mbits/s |
| 802.11y | En dév. | 3.7 GHz | 54 Mbits/s |
| | Technique de modulation | Rayon de transmission intérieur | Rayon de transmission extérieur |
| 802.11 (legacy) | | ~20 mètres | ~100 mètres |
| 802.11a | OFDM | ~35 mètres | ~120 mètres |
| 802.11b | DSSS | ~38 mètres | ~140 mètres |
| 802.11g | QFDM | ~38 mètres | ~140 mètres |
| 802.11n | | ~0 mètres | ~250 mètres |
| 802.11y | | ~50 mètres | ~5000 mètres |

TAB. 3.2 – Résumé des protocoles de la famille 802.11.

les applications possibles de la communication dans les transports, les besoins de ces applications en terme de communication et l'état de l'art dans le domaine.

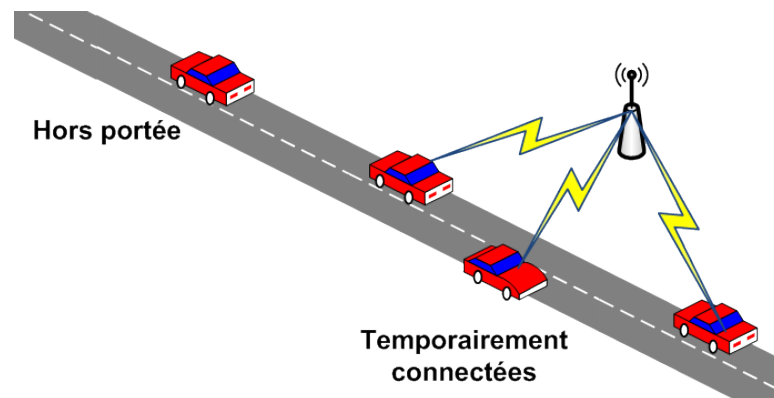


FIG. 3.5 – Réseau mobile dynamique avec point d'accès où les noeuds (véhicules) formant le réseau changent constamment.

3.3.1 Applications de la communication

Plusieurs applications installées dans les véhicules pourront bénéficier de cette technologie. Tout d'abord, on retrouve les applications utilisant la communication inter-

véhicules. Ces applications se basent sur le partage entre véhicules des informations provenant des capteurs. Les véhicules peuvent communiquer leurs intentions, leur statut ou encore de l'information sur l'état des routes (conditions routières, trafic, etc.).

L'autre type de communication dans les transports est la communication route-à-véhicule. Ce type de communication sert principalement à fournir de l'information à la voiture à propos des routes et de la signalisation qui l'entourent. Les applications qui utilisent ces informations peuvent alors avertir le conducteur des risques avoisinants.

Les tableaux 3.3 et 3.4 donnent quelques exemples d'applications de la communication inter-véhicules et route-à-véhicule [Bishop, 2005].

3.3.2 Besoins technologiques des applications

Pour répondre aux besoins des applications de sécurité comme les systèmes de détection de collision et les régulateurs de vitesse adaptatifs, il convient bien entendu d'avoir un temps de latence très bas. Deuxièmement, la bande passante doit être raisonnable étant donné le nombre de voitures pouvant utiliser le canal en même temps dans un espace relativement restreint. De plus, les techniques d'adressage et de routage doivent tenir compte de l'aspect hautement dynamique du réseau. Un véhicule peut entrer et sortir très rapidement de la portée de communications d'un autre véhicule. Pour cette raison, l'établissement de la connexion doit se faire très rapidement, et si possible utiliser un protocole ne demandant pas de connexion préalable pour l'envoi de messages.

Comme le temps de latence doit être très petit, le choix d'une architecture réseau cellulaire, c'est-à-dire avec des stations de base, n'est pas approprié. En effet, le temps que le signal atteigne l'antenne de la station de base et revienne jusqu'aux destinataires est trop élevé pour des applications sensibles telles qu'un système d'évitement de collision. Pour cette raison, les travaux sur la communication inter-véhicules portent essentiellement sur les réseaux mobiles ad hoc (« Mobile ad hoc NETWORKS », MANET) [Conti *et al.*, 2004]. Cependant, pour des applications de divertissement ou d'information sur le réseau routier global ou des applications véhiculant d'autres données non critiques, une architecture avec station de bases serait appropriée.

| Application | Fonction | Données transmises |
|--|---|--|
| Avertisseur coopératif de collision frontale | Aide le conducteur à éviter une collision frontale. Les données reçues de la voiture venant en sens inverse sont couplées avec celles de la voiture locale pour estimer les risques de collision. | Position, vitesse, direction, accélération, taux de changement de direction. |
| Feu de freinage électronique d'urgence | Quand un véhicule freine brusquement à l'avant, un message est envoyé aux autres véhicules qui le suivent pour avertir de la manœuvre même si les capteurs radars ou la visibilité du conducteur sont limités par la température ou par d'autres véhicules. | Position, direction, vitesse, décélération. |
| Avertisseur de conditions routières | Les conditions de la route sont détectées par des capteurs du véhicule et sont transmises aux autres véhicules. Un véhicule recevant le message peut alors faire des recommandations de vitesse au conducteur. | Position, direction, conditions de la route, paramètres. |
| Avertisseur de changement de voie | Avertit le conducteur si un risque de collision est présent lorsqu'il signale son intention de changer de voie en traitant les messages reçus des véhicules avoisinants. | Position, direction, vitesse, accélération, état des clignotants. |

TAB. 3.3 – Exemples d'applications basées sur la communication inter-véhicules.

| Application | Fonction | Données transmises |
|---|--|--|
| Avertisseur de violation de feux de signalisation | Avertit le conducteur qu'il doit s'immobiliser s'il approche une lumière, qu'il n'a pas le droit de passage et que le système prédit qu'il sera en infraction selon la vitesse du véhicule et sa décélération. | Statut du feu de signalisation, position de la ligne d'arrêt de l'intersection, condition de la surface de la route. |
| Avertisseur de violation de panneau d'arrêt | Avertit le conducteur si sa vitesse est anormalement élevée pour la distance qui le sépare du panneau d'arrêt. | Position du panneau d'arrêt, direction du panneau. |
| Assistance d'intersection avec arrêt | Avertit le conducteur qui s'engage dans l'intersection s'il y a risque de collision avec un autre véhicule approchant l'intersection. | Position de l'autre véhicule, sa vitesse et sa direction. |
| Assistance de vitesse pour les courbes | Aide le conducteur à s'engager dans les courbes avec une vitesse sécuritaire selon l'information envoyée par des stations fixes à l'approche des courbes. | Position de la courbe, limite de vitesse de la courbe, rayon de courbure, condition de la surface de la route. |

TAB. 3.4 – Exemples d'applications basées sur la communication route-à-véhicule.

3.3.3 État de l'art

Plusieurs groupes de chercheurs se sont penchés sur le sujet de la communication sans-fil dans les transports, même avant que les LAN sans fil se démocratisent. Par exemple, des tests ont été faits dans les années 1990 pour comparer les performances de diverses méthodes d'accès multiple. [Tank et Linnartz \[1997\]](#) de l'Université de Californie et supporté entre autres par le projet « *Partners in Advanced Transit and Highways* » (PATH) a testé les méthodes suivantes : accès multiple à répartition dans le temps (« Time Division Multiple Access », TDMA), accès multiple par répartition en code avec séquence directe (« Direct-Sequence Code Division Multiple Access », DS-SS) et TDMA avec saut de fréquence (« Frequency-Hopping »). Ses résultats démontraient que la méthode CDMA obtenait légèrement de meilleurs résultats par rapport au taux d'erreurs de transmission.

Plusieurs consortiums se sont aussi formés entre différents groupes de chercheurs, des acteurs de l'industrie et des gouvernements pour faire avancer la recherche dans le domaine de la communication inter-véhicules. « Fleetnet - Internet on the Road » est un projet basé en Europe, plus particulièrement en Allemagne, et regroupe six compagnies et trois universités dont le but est de développer un réseau ad hoc pour la communication inter-véhicules [[FleetNet, 2002](#)]. Le réseau proposé doit couvrir les champs d'application tels que l'assistance à la conduite, le partage d'informations routières de façon décentralisée ainsi qu'un média de communication pour les usagers [[Hartenstein et al., 2001](#)]. Les partenaires de ce projet ont voulu que la communication inter-véhicules ne serve pas uniquement à la sécurité des passagers, mais aussi au confort et aux loisirs. Un exemple d'application serait la possibilité pour les passagers arrière de jouer à des jeux en réseaux avec les usagers d'autres véhicules voisins.

Le choix pour le système sans-fil du projet Fleetnet est le « UTRA Terrestrial Radio Access with Time Division Duplexing » (UTRA-TDD). L'avantage d'une telle technologie est qu'elle supporte le transfert de données dans les deux sens (ascendant et descendant), qu'elle permet la communication sur de grandes distances et qu'elle supporte une vitesse élevée des nœuds du réseau. De plus, une bande publique réservée dans la plage des 2010-2020 MHz existe en Europe pour cette technologie. Cela exigerait cependant du développement pour transformer l'architecture cellulaire de cette technologie en une architecture ad hoc.

Le protocole de routage de Fleetnet est basé sur la position des véhicules plutôt que sur la topologie du réseau [[Festag et al., 2004](#)]. Leurs résultats démontrent que sur l'autoroute, le taux de messages reçus reste à 100 % alors que la courbe décroît

pour un protocole de routage standard. En ville, les résultats sont moins concluants. Il faut combiner le protocole de routage à une carte des rues pour obtenir une meilleure propagation des messages en milieu urbain.

Omis Fleetnet, il y a plusieurs autres projets européens dont CarTALK2000. Celui-ci possède un nouveau protocole MAC dont les performances ont été analysées dans [Borgonovo et al. \[2003\]](#). Ce protocole ad hoc MAC utilise un mécanisme de TDMA dynamique qui peut être facilement adapté pour le UTRA-TDD, couche physique choisie pour le projet CarTALK2000. Les résultats de ce protocole montrent qu'il offre de très bonnes performances par rapport au temps d'accès pour le broadcast « simple-saut ». De plus, il offre une façon simple d'implémenter au niveau de la couche MAC un service de broadcast « multi-sauts ».

3.3.4 DSRC et WAVE

DSRC, pour *Dedicated Short Range Communications*, est « un service de communication de courte à moyenne portée qui supporte des applications de sécurité et d'autres types d'applications dans un environnement de communication route-à-véhicule et inter-véhicules » [[Armstrong, 2005](#)]. DSRC se veut un complément de la communication cellulaire en offrant des vitesses de transfert élevées avec un temps de latence minimale dans des circonstances où le délai de transmission est critique. Ce standard est développé par un groupe de travail formé de plusieurs organismes internationaux ainsi que plusieurs compagnies dont GM, Daimler-Chrysler, Motorola, Nissan et Toshiba. Les principaux responsables de son développement sont les comités ASTM [[ASTM, 2008](#)] et IEEE [[IEEE, 2008b](#)].

En octobre 1999, la Commission Fédérale des Communications (*Federal Communications Commission* (FCC)) américaine a alloué une bande de fréquence dans les 5.9 GHz pour les applications reliées au domaine des systèmes de transport intelligents. Le standard DSRC utilise cette bande fréquence et est basé sur la couche physique de la norme IEEE 802.11a ainsi que sur la couche MAC de la norme IEEE 802.11 [[Yin et al., 2004](#)]. La portée théorique de communication est de 1000 mètres mais en pratique, elle se situe plutôt aux alentours de 200 mètres [[Bishop, 2005](#)]. Le principal avantage du DSRC est le temps de latence très bas, sous les 100 millisecondes, ce qui est idéal pour les applications de sécurité qui demandent un délai dans cette plage.

Le standard WAVE [[IEEE, 2006](#)], pour *Wireless Access Vehicular Environment*, englobe en partie le standard DSRC dans le sens qu'il supporte les caractéristiques traditionnelles du DSRC mais supporte aussi une portée d'opération plus grande (plus

d'un kilomètre), des vitesses de transfert plus élevées ainsi que de la communication unicast [Bishop, 2005]. Adapté de IEEE 802.11a, WAVE a reçu la désignation de IEEE 802.11p dans la famille des protocoles sans fil. Aux États-Unis en particulier, l'industrie est axée fortement sur l'utilisation future du protocole WAVE avec les fréquences dédiées au DSRC. WAVE peut être vu comme l'intégration de DSRC dans le monde des standards sans fil IEEE.

3.4 Conclusion

Dans ce chapitre, nous avons vu les architectures de bases des réseaux informatiques basées sur le principe de couches et d'interfaces. Chacune des couches du modèle OSI répond à des besoins particuliers et offre des services à la couche supérieure tout en utilisant les services de la couche inférieure. Cette architecture permet aux applications utilisant la communication de faire abstraction du support de transmission des données et de fonctionner aussi bien sur des réseaux filaires que sur des réseaux sans fil.

Nous avons aussi vu les principes de base des communications sans fil et plus particulièrement des réseaux mobiles. Les réseaux sans fil sont maintenant indispensables à la pleine exploitation de nos appareils mobiles et ils gagnent constamment en vitesse et en fiabilité. Il existe deux types de réseaux sans fil : les réseaux ad hoc et les réseaux avec point d'accès. Tous deux ont leurs avantages, mais les réseaux ad hoc sont préférables pour les applications de sécurité dans les véhicules dû à leur très faible temps de latence.

Il reste beaucoup de travail à faire afin d'améliorer les techniques de communications sans fil et les réseaux mobiles. Cependant, vu l'immense bénéfice qu'ils peuvent avoir dans le domaine des systèmes de transport intelligents, il n'y a aucun doute qu'ils feront leur apparition prochainement sur nos routes.

Chapitre 4

Apprentissage de la communication inter-véhicules

Dans les chapitres précédents, nous avons vu en quoi consiste l'intelligence artificielle, son utilité pour les systèmes de transport intelligents et en quoi elle pourrait aider à la communication inter-véhicules. Nous avons aussi vu les principes de bases des réseaux et plus particulièrement des réseaux sans fil. Bien qu'il reste plusieurs défis à surmonter avant l'apparition des réseaux mobiles sur nos routes, nous pouvons tout de suite nous attaquer au problème de l'utilisation de cette nouvelle technologie dans nos voitures. Quelles applications devraient utiliser la communication inter-véhicules et de quelle façon ?

Ce chapitre présente la recherche effectuée dans le cadre de cette maîtrise pour l'application de la communication inter-véhicules à un système de régulateur de vitesse adaptatif et collaboratif (« Collaborative Adaptive Cruise Control », CACC). L'architecture du CACC et l'utilisation de la communication dans le système seront présentées dans la première section. La deuxième section présente, quant à elle, les algorithmes d'apprentissage par renforcement et les processus de Markov les sous-tendant. Suivent ensuite la description de l'approche utilisée pour implanter la communication dans le CACC, la présentation du simulateur et les résultats.

4.1 Architecture du CACC

Pour bien comprendre ce qu'est un système de régulateur de vitesse adaptatif et collaboratif, il faut regarder l'évolution du régulateur de vitesse (« Cruise Control », CC). Tout d'abord, les régulateurs de vitesse ont comme but d'aider le conducteur en maintenant une vitesse constante sans qu'il ait à garder le pied sur la pédale de gaz. Le but est de réduire la fatigue du conducteur et d'améliorer la fluidité du trafic. Vient ensuite les régulateurs de vitesse adaptatifs (« Adaptive Cruise Control », ACC) équipés de laser à l'avant du véhicule. En plus de maintenir une vitesse constante, ces systèmes réduisent automatiquement la vitesse du véhicule pour garder une distance sécuritaire avec la voiture qui précède. Lorsque la voie est libre, le système reprend sa vitesse normale.

Les CACC sont donc une évolution des CC et ACC. En échangeant des informations sur le freinage et l'accélération à l'aide de la communication, ces systèmes permettent de garder une plus petite distance entre chaque véhicule tout en restant sécuritaire. Cela augmente aussi le débit de trafic et diminue le risque d'accident dû aux erreurs humaines. Cependant, le système présenté dans ce chapitre est plus évolué qu'un CACC de base et s'approche plus d'un système de contrôle autonome du véhicule. En plus de contrôler la vitesse, il peut contrôler la direction du véhicule et décider à quel moment changer de voie.

Ce système de CACC a été réalisé dans le cadre du projet d'AUTO21 au laboratoire DAMAS. Julien Laumônier, étudiant au doctorat, était responsable de l'aspect coopératif des voitures et du contrôle de haut niveau [Laumonier, 2008]. Charles Desjardins, étudiant à la maîtrise, était responsable du contrôle de bas niveau comme l'accélération, le freinage et la direction du véhicule [Desjardins, 2008]. Pour ma part, mon rôle était de concevoir une architecture de communication entre les voitures.

Le système est conçu comme un agent intelligent et autonome capable de contrôler la voiture. Comme tout agent intelligent décrit dans le chapitre 2, il perçoit son environnement à l'aide de capteurs, choisit une action et l'exécute. Dans le cas présent, la prise de décision d'actions se fait par une architecture à deux couches. De plus, un module de communication permet d'échanger de l'information entre les agents, c'est-à-dire entre les véhicules voisins. La figure 4.1 décrit l'architecture du système.

La première couche est la couche de coordination. Elle est responsable du choix d'actions de haut niveau. Les voitures avoisinantes communiquent entre elles et doivent délibérer pour choisir les meilleures actions à prendre selon la situation. Par exemple,

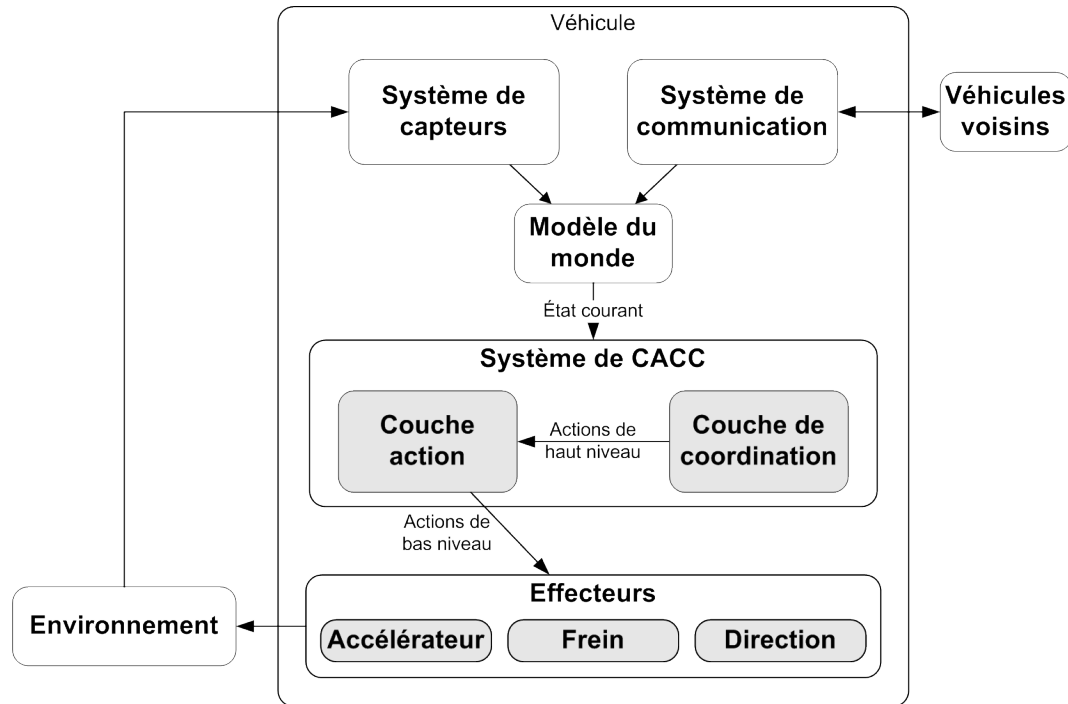


FIG. 4.1 – Architecture du système CACC.

la figure 4.2 démontre un choix de voie collaboratif où le dernier véhicule se déplace dans la voie de gauche pour laisser le deuxième véhicule dépasser par la voie du centre. Ce type de situation conflictuelle est très présent sur nos routes et force souvent le deuxième véhicule à freiner pour laisser passer le dernier véhicule avant de pouvoir faire son déplacement. La couche de coordination de la figure 4.1 choisit donc la meilleure action possible en fonction des actions des autres agents pour maximiser la sécurité et l'efficacité du trafic local et global.

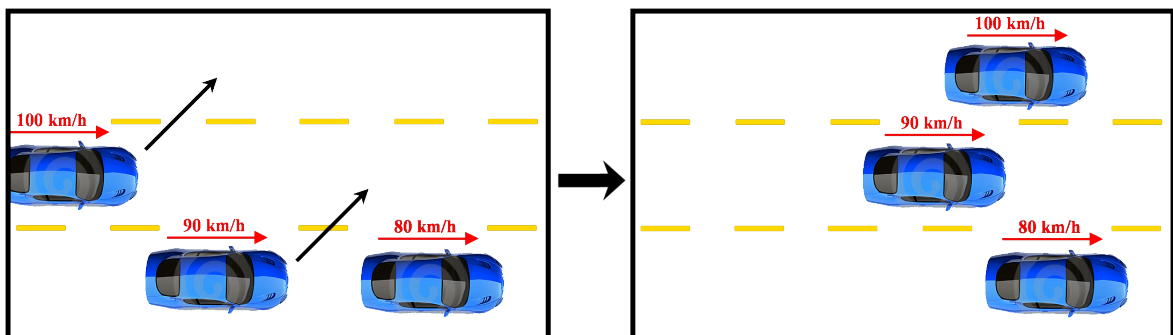


FIG. 4.2 – Choix de voie collaboratif.

La deuxième couche est la « couche action » et représente le contrôleur de bas niveau. Une fois qu'une politique de haut niveau a été choisie par la couche de coordination, elle

| Politique | |
|--|---|
| | |
| Actions de haut niveau | Politiques de la « couche action » |
| <ol style="list-style-type: none"> 1. Freiner et suivre le véhicule précédent jusqu'à ce que la voie de dépassement soit libre. 2. Changement de voie. 3. Accélération jusqu'à vitesse désirée. | <ol style="list-style-type: none"> 1. Politique de suivi. 2. Politique de changement de voie. 3. Politique de régulation de vitesse. |

TAB. 4.1 – Actions des couches.

transmet les actions de cette politique une à une à la « couche action ». La « couche action » est responsable de mener à bien l'exécution de chaque action. Dès qu'une action est complétée avec succès, elle en informe la couche coordination qui lui transmet l'action suivante. Le tableau 4.1 montre un exemple d'une situation de dépassement. Le véhicule roulant à 90 km/h doit dépasser un véhicule plus lent à l'avant. La couche de coordination choisit la politique qui consiste à attendre que la voie de gauche se libère puis effectue le dépassement. Cela se traduit en trois actions de haut niveau qui seront communiquées à la « couche action ». Pour chacune de ces actions, la « couche action » exécute la politique optimale correspondante. Le contrôleur de haut niveau peut interrompre le contrôleur de bas niveau au milieu de l'exécution d'une action sauf pour les changements de voies. Nous faisons l'hypothèse qu'un changement de voie doit toujours être complété correctement avant d'amorcer une autre action. Le diagramme d'état à la figure 4.3 montre l'interaction entre les deux couches et les transitions possibles qui peuvent être faites par la couche de coordination.

Pour qu'un tel système soit fonctionnel, un module de communication est primordial pour échanger des informations avec les autres voitures. Autant la couche coordination que la « couche action » s'appuient sur ce module pour avoir de bonnes performances. Pour la couche de coordination, la communication lui permet de dialoguer avec les autres voitures pour délibérer sur les actions à prendre dans la situation courante. Pour la « couche action », la communication est une source d'information cruciale sur l'état dynamique des autres voitures, comme leurs vitesses, accélération, etc. La section 4.3

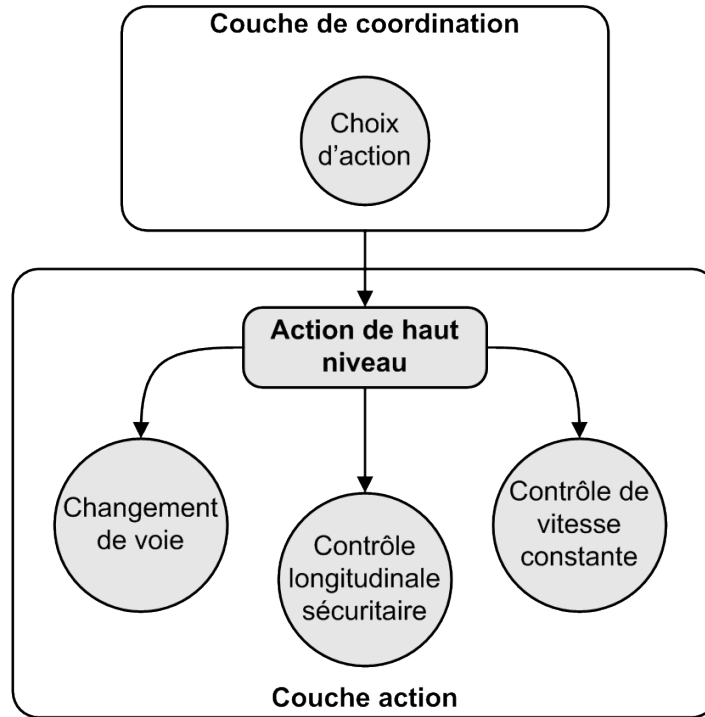


FIG. 4.3 – Interactions entre les deux couches du système CACC.

décrit en détail l’approche utilisée pour implanter la communication dans le système.

Les trois modules utilisent des algorithmes d’apprentissage par renforcement. De cette façon, le système peut s’adapter à son environnement et adopter des politiques qui réagissent correctement aux évènements perçus. Pour le module de contrôle de la « couche action », cela implique d’apprendre les actions de bas niveau à appliquer sur les effecteurs pour guider correctement le véhicule. Pour la couche de haut niveau, le module doit apprendre à réagir correctement aux actions des autres agents pour prendre des décisions efficaces et sécuritaires. Pour le module de communication, l’apprentissage par renforcement permet d’obtenir une politique de communication qui optimise la bande passante tout en offrant des informations précises et récentes aux modules de contrôle.

4.2 Processus décisionnels de Markov (MDP) et apprentissage par renforcement

Comme mentionné précédemment, le système est vu comme un agent intelligent et autonome. De plus, c’est un agent apprenant, c’est à dire capable d’apprentissage. Un agent apprenant est un agent qui apprend et s’adapte en fonction de son expérience.

La figure 4.4 montre l'architecture interne d'un tel agent.

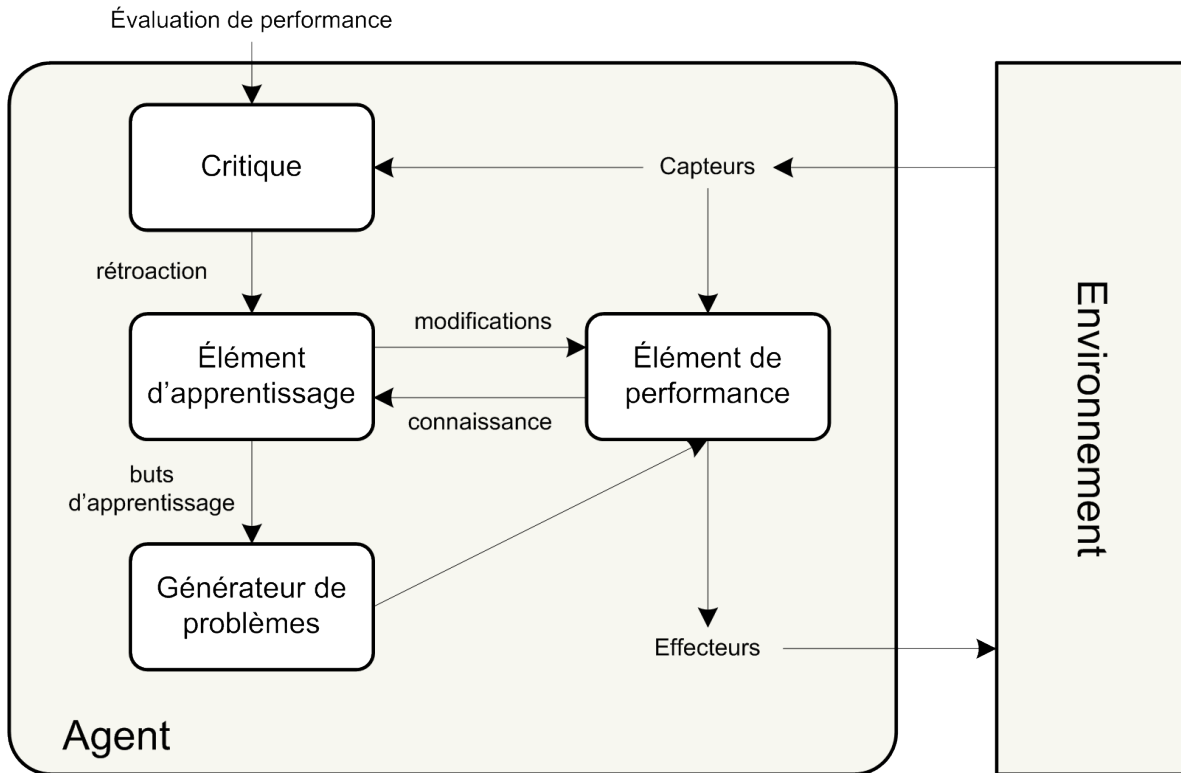


FIG. 4.4 – Architecture d'un agent intelligent [Russell et Norvig \[2003\]](#).

Le principal élément est le module de performance. Il est responsable de choisir l'action externe en fonction des perceptions de l'environnement. Il s'agit en fait de l'élément de base que l'on retrouve au niveau de tous les agents. La principale différence d'un agent apprenant réside plutôt au niveau de l'élément d'apprentissage. Ce module a pour but de modifier le module de performance en fonction des résultats des actions de l'agent dans l'environnement. Cependant, une perception de l'environnement ne donne aucune information sur les performances de l'agent. Autrement dit, l'agent perçoit l'état courant de l'environnement mais ne sait pas si celui-ci est bon ou non. Par exemple, un agent jouant aux échec peut constater qu'il a gagné la partie, mais il a besoin d'une fonction de récompense pour savoir que c'était le but souhaité. Ainsi, le module de critique est responsable d'informer l'élément d'apprentissage de la façon dont l'agent se comporte par rapport à un critère de performance donnée.

Le dernier module est le générateur de problèmes. Cet élément a pour but de proposer des actions à l'élément de performance pour permettre à l'agent de découvrir des expériences nouvelles et enrichissantes. Sans ce module, l'agent réutiliserait sans cesse les mêmes actions qu'il pense les meilleures sans essayer de nouvelles possibilités.

Avec le générateur de problème, l'agent risquera de nouvelles actions qui peuvent être sous-optimales à court terme mais qui lui feront peut-être découvrir des stratégies plus optimales à long terme.

Pour permettre à l'agent d'apprendre, un cadre formel de modélisation de l'environnement est nécessaire. Les Processus Décisionnels de Markov (« Markov Decision Process », MDP) constituent un tel cadre depuis plusieurs années dans le domaine de l'apprentissage par renforcement. Ils sont particulièrement efficaces pour représenter des problèmes de décision séquentiels où le but est de trouver les meilleures actions à effectuer pour maximiser l'utilité que l'agent peut recevoir. L'utilité pour l'agent représente sa satisfaction. Lorsqu'un état est préférable à un autre, on dit qu'il a une plus grande utilité pour l'agent.

Formellement, un MDP peut être décrit par :

- A un ensemble fini d'actions,
- S un ensemble fini d'états,
- $\mathcal{R} : S \times A \rightarrow R$ une fonction donnant la récompense immédiate d'effectuer une action donnée dans un état donné,
- $\mathcal{P} : S \times A \times S \rightarrow [0, 1]$ une fonction de transition donnant la probabilité de transiger d'un état à un autre en effectuant une action.

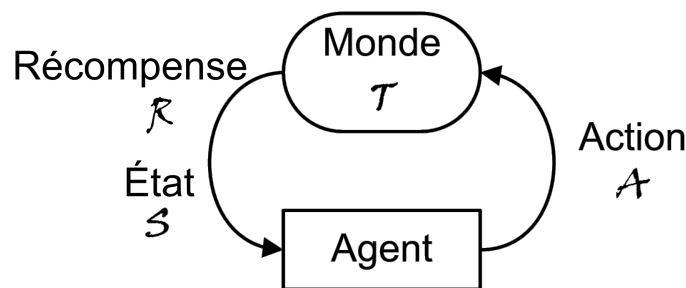


FIG. 4.5 – Processus décisionnel de Markov.

La plupart des problèmes peuvent être modélisés d'une manière ou d'une autre par un MDP. Prenons par exemple la construction de l'espace d'états, c'est-à-dire la construction de l'ensemble S . L'environnement d'un agent peut être représenté par un ensemble de variables. Dans le cas d'un robot aspirateur, son état peut-être représenté par trois variables : deux pour sa position (x, y) et une pour l'état du plancher sous le robot (propre ou sale). On obtient alors un espace d'états à trois dimensions, une pour chaque variable. À tout moment, l'état du robot correspond à un point dans cet espace d'états. Il en va de même pour l'espace d'actions qui est formé par toutes les variables pour lesquelles l'agent doit choisir une valeur.

Lorsque les variables formant les espaces d'états ou d'actions sont continues, on se retrouve avec une infinité de possibilités d'états ou d'actions. Généralement, pour simplifier les problèmes, on discrétise alors l'espace d'états et d'actions. Cela correspond à transformer une variable continue en une variable discrète en la découpant en plage de valeurs. Par exemple, l'espace d'états du robot aspirateur formé par trois variables continues (position en x , position en y , quantité de saletés) pourrait être réduit en discrétisant sa position en cases de dimension fixe et l'état du plancher par une variable binaire : propre ou sale. En divisant une pièce en 10 rangées de 10 cases, on obtient un ensemble de 200 états (100 cases, propres ou sales).

La solution d'un MDP ne peut pas être représentée par une séquence fixe d'actions, car l'agent peut aboutir dans des états différents à chaque exécution de cette séquence. En conséquence, la solution d'un MDP doit indiquer ce que l'agent doit faire pour tout état qu'il pourrait atteindre. Une solution de ce type se nomme une politique. Une politique est généralement représentée par le symbole π , et $\pi(s)$ est l'action recommandée par la politique π pour l'état s .

Pour trouver la solution optimale d'un MDP, la propriété de Markov doit être satisfaite. Cette propriété indique que la probabilité d'atteindre le prochain état s' dépend uniquement de l'état courant s . Avec cette propriété, l'état courant du système encapsule toute l'information nécessaire pour prendre une décision.

En connaissant la fonction de transition ainsi que la fonction de récompense, on peut calculer facilement la politique optimale d'un MDP à l'aide de l'équation de Bellman. Cette équation calcule une table de valeurs « état » ($V^*(s)$) qui donne pour chaque état la récompense espérée si la politique optimale (π^*) est suivie. Cette valeur se nomme l'utilité de l'état. L'équation de Bellman (4.1) calcule l'utilité d'un état :

$$V^*(s) = \max_a \sum_{s'} \mathcal{P}_{ss'}^a [\mathcal{R}_s^a + \gamma V^*(s')] \quad (4.1)$$

où

- $V^*(s)$ est l'utilité de l'état s ,
- $\mathcal{P}_{ss'}^a$ est la probabilité d'atteindre l'état s' en exécutant l'action a dans l'état s ,
- \mathcal{R}_s^a est la récompense obtenue en exécutant l'action a dans l'état s ,
- et γ est une constante dans l'intervalle $[0, 1]$ permettant de pondérer l'importance des récompenses à long terme par rapport à la récompense immédiate.

Cependant, dans plusieurs problèmes réels, les fonctions de transition et de récompense sont inconnues puisque nous ne connaissons pas le modèle de l'environnement. Dans ces cas, nous ne pouvons pas trouver la politique optimale à l'aide de l'équation de Bellman et nous ne pouvons pas faire d'apprentissage supervisé puisque nous ne savons pas qu'elles actions mènent au but souhaité. L'agent doit alors agir et observer le résultat de ses actions directement dans l'environnement pour apprendre une politique optimale. Ce processus s'appelle : l'apprentissage par renforcement, RL (« Reinforcement Learning »).

Le Q-Learning est un des algorithmes les plus connus pour résoudre des problèmes d'apprentissage par renforcement. Avec cet algorithme, l'agent apprend une table de valeurs « état-action » plutôt qu'une table de valeurs « état ». Chaque valeur état-action, aussi appelée une Q-Valeur, donne la récompense espérée de choisir l'action a dans l'état s . Il devient alors possible d'apprendre l'effet de chaque action possible dans un état donné et de comparer l'utilité de ces actions sans connaître le modèle de l'environnement. L'algorithme 1 présente la fonction interne d'un agent utilisant le Q-Learning.

Algorithme 1 : fonction Agent-Q-Learning(percept) **retourne** une action (adapté de [Russell et Norvig \[2003\]](#)).

Précondition : **percept** donne l'état courant s' et la récompense r' .

Précondition : Q est la table des Q-Valeur indexée par état et par action.

Précondition : N_{sa} est une table de fréquences pour des paires état-action.

Précondition : s, a, r sont l'état, l'action et la récompense précédente.

```

1: si  $s$  n'est pas null alors
2:   incrémenter  $N_{sa}[s, a]$ 
3:    $Q[a, s] \leftarrow Q[a, s] + \alpha(N_{sa}[s, a])(r + \gamma \max_{a'} Q[a', s'] - Q[a, s])$ 
4: fin si
5: si  $s'$  est un état terminal alors
6:    $s, a, r \leftarrow null$ 
7: sinon
8:    $s \leftarrow s'$ 
9:    $a \leftarrow \operatorname{argmax}_{a'} f(Q[a', s'], N_s a[s', a'])$ 
10:   $r \leftarrow r'$ 
11: fin si
12: retour  $a$ 

```

Le coeur de l'algorithme est la mise à jour d'une valeur état-action à la ligne 3. Chaque fois que l'agent exécute une action dans l'environnement, il reçoit une récompense. Avec cette nouvelle expérience, il peut ajuster la Q-Valeur de cette action pour cet état avec la récompense obtenue et avec l'équation de la ligne 3 de l'algorithme.

La fonction α ajuste la vitesse d'apprentissage. Elle prend en paramètre le nombre de fois que l'action a a été exécuté dans l'état s et retourne une valeur entre 0 et 1. Le paramètre γ représente pour sa part l'importance des récompenses futures plutôt que des récompenses à court terme, il s'agit donc d'un facteur. Finalement, l'agent sélectionne une action avec la fonction d'exploration versus exploitation de la ligne 9.

Il existe bien sûr plusieurs autres algorithmes pour faire de l'apprentissage par renforcement. Certains ne nécessitent pas une discrétisation de l'espace d'états et d'actions comme le Q-Learning et ont de meilleures capacités de convergence dépendamment du problème. Cependant, c'est l'algorithme du Q-Learning qui a été utilisé pour l'apprentissage de la communication inter-véhicules présentée dans ce chapitre.

4.3 Implémentation de la communication véhiculaire

Cette section explique l'implémentation de la communication véhiculaire autant du point de vue architecturale que technique. Nous verrons comment chacune des deux couches du système utilise la communication et à quel niveau se situe l'apprentissage de la communication véhiculaire.

4.3.1 Modélisation des couches de bas niveau

Le module de communication est au coeur de notre système de CACC. Cependant, les recherches dans le cadre de ce mémoire ne portent pas sur le choix des technologies sans fil à utiliser mais plutôt sur la définition d'une stratégie de communication de haut niveau. Si l'on se réfère au modèle OSI du chapitre 3, il s'agit des couches applications de haut niveau. Pour développer cette stratégie sans avoir à gérer les problèmes et les solutions des couches de bas niveau, nous faisons l'hypothèse que ces couches répondent aux critères suivants :

- *Protocole sans connexion* : un message peut être envoyé d'un point à un autre sans arrangement préalable, c'est-à-dire sans qu'il y ait eu une connexion d'établie entre ces deux noeuds.
- *Protocole de routage broadcast/multicast* : les messages ne sont pas envoyés à un seul véhicule mais plutôt à tous les véhicules qui sont dans le rayon de transmission de l'émetteur.

- *Détermination du rayon de transmission* : pour chaque message envoyé, nous devons être en mesure de spécifier le rayon de propagation du message. Dépendamment de la distance demandée et de la capacité des antennes de transmission, un protocole de routage à sauts multiples pourrait être utilisé.
- *Possibilité de géocasting* : les destinataires du message peuvent être spécifiés de façon géographique. Par exemple, un message pourrait être envoyé uniquement aux véhicules derrière et dans la même voie que le transmetteur.
- *Réseau ad hoc* : un tel réseau est préférable vu qu'un réseau avec infrastructure serait trop coûteux à maintenir.
- *Paquets de petite taille* : le contenu de chaque message sera minimal.
- *Gratuit* : l'utilisation de bande de fréquence gratuite serait préférable.

Nous avons choisi de nous baser sur les spécifications des protocoles de la famille 802.11 pour l'implémentation des protocoles de bas niveau dans nos simulations. Selon les résultats obtenus dans [Günter et Großmann, 2005] et [Jatinder Pal Singh et Clawin, 2004], nous avons une bonne idée des performances qui peuvent être obtenues avec ces protocoles dans des tests réels. Nous faisons aussi l'hypothèse, en se basant sur ces protocoles, que nous avons une bande passante de 4 Mbits/sec, un délai de 200 millisecondes et un rayon de transmission de 100 mètres. Ces données peuvent sembler pessimistes mais le but est de simuler un environnement urbain où certains effets physiques (décrits précédemment) diminuent la qualité de la transmission. À chaque pas de temps, un véhicule peut décider d'envoyer un ou plusieurs messages ou décider de ne pas en envoyer du tout. Cependant, la taille totale de tous les messages envoyés par tous les véhicules doit être conforme à la bande passante fixée. Les messages en surplus sont gardés et envoyés au pas de temps suivant. Pour le modèle de routage, nous avons décidé d'utiliser la diffusion, c'est à dire le « broadcasting ». Ainsi, chaque message est reçu, après le délai spécifié, par tous les véhicules présents dans le rayon de transmission de l'émetteur.

4.3.2 Architecture technique

Nous avons modélisé la communication dans le simulateur à l'aide de trois modules : un module de gestion global qui modélise le média de communication, un module de transmission et un module de réception, tous deux implantés dans chaque véhicule du simulateur. Le module de transmission est utilisé pour créer des messages et les envoyer sur un port de communication spécifique. Les messages sont ensuite transférés au module de gestion globale de la communication qui simule les couches de routage, de liaison et la couche physique du protocole de communication sans fil. Ce module est utilisé à des fins de simulation et gère la bande passante, les délais de transmission,

le rayon de transmission, les collisions et le modèle de routage. Il transmet ensuite les messages aux modules de réception selon les paramètres de communication du protocole implanté. Finalement, le module de réception sert de passerelle pour que les véhicules reçoivent les messages.

La figure 4.6 décrit l'« architecture technique » de la communication dans le simulateur. Les messages contenus dans la pile d'envois sont les messages en attente de transmission (qui attendent l'accès au canal partagé) et les messages en cours de transmission (simulation de la propagation). À chaque pas de temps, la fonction interne de simulation de la communication décide quels messages de la pile d'envois atteignent finalement leurs destinataires et quels messages sont perdus (flèche pointillée). Elle place les messages dans la pile de réception où ils seront distribués aux véhicules correspondants.

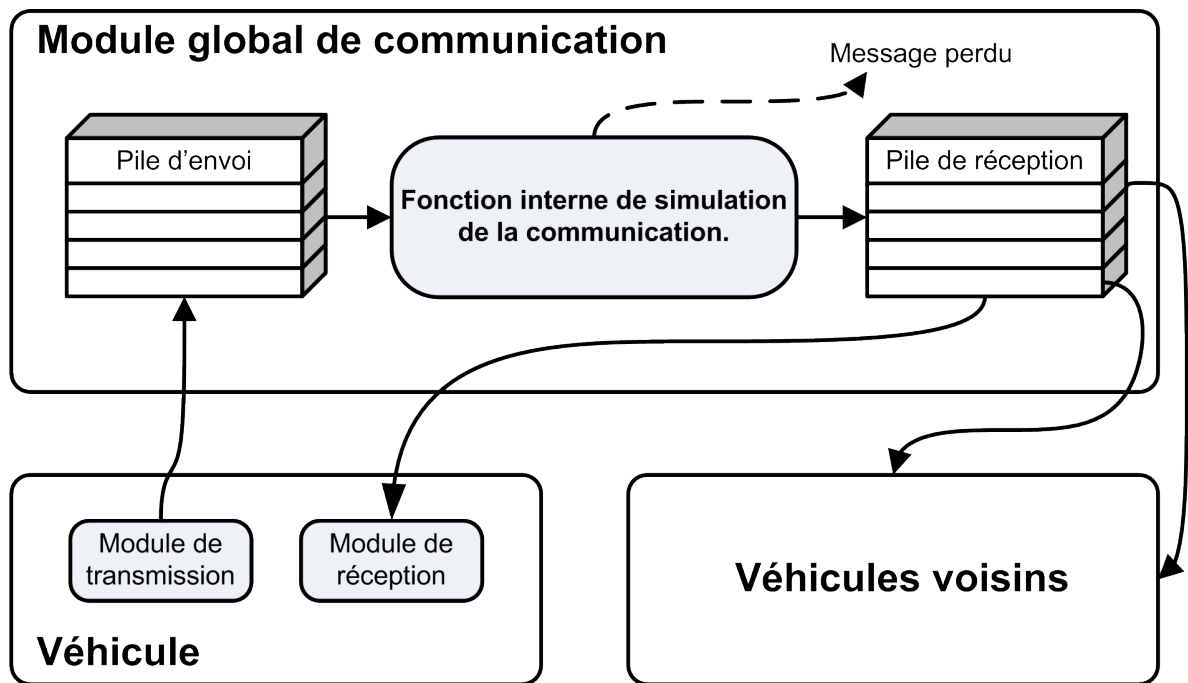


FIG. 4.6 – Architecture de la communication dans le simulateur.

Chaque message contient l'identification de l'émetteur, l'étampe du temps d'envoi, un numéro de port, l'identifiant du destinataire et une chaîne de caractère représentant le contenu du message. Le numéro du destinataire est configuré à -1 s'il s'agit d'un message diffusé (ce qui est toujours le cas pour l'instant dans nos simulations). La taille de la chaîne représentant le contenu est utilisée par le module de communication globale pour gérer la bande passante.

Cette structure offre une grande flexibilité pour toutes applications nécessitant de

la communication dans notre simulateur. Vu que chaque message inclut un numéro de port, une application n'a qu'à choisir un port disponible pour envoyer des messages à travers le transmetteur et le récepteur en spécifiant ce numéro de port. Avec cette approche, un message envoyé par le module de coordination du véhicule n'interférera pas avec un message envoyé par le module action puisque chaque message sera envoyé sur un port différent dépendamment de l'application.

4.3.3 Communication dans le module de coordination

Pour le module de coordination, la communication sert principalement au processus de délibération entre les agents. Des messages sont envoyés et reçus suivant ce processus et portent sur le choix des actions de haut niveau qui devraient être faites dans la situation actuelle. Comme le choix des messages à envoyer fait partie intégrante du protocole de délibération et de décision de la couche de coordination, elle est libre d'envoyer des messages à n'importe quel moment et à n'importe qui. L'apprentissage de la communication se fait donc plus au niveau de la « couche action ».

4.3.4 Communication dans le module action

Le but premier de la communication dans le module action de notre système de CACC est d'échanger des données provenant des capteurs. Un véhicule peut ainsi avoir une représentation plus précise de son environnement local, par exemple sur la vitesse et l'accélération du véhicule qui le précède. Cela est possible grâce aux véhicules voisins qui envoient leurs propres positions et des informations sur leurs mouvements.

La problématique est de savoir quelles informations envoyer, à quelle fréquence, etc. Beaucoup de recherches ont été faites sur la communication entre agents. Une façon très commune d'intégrer la communication dans un problème multi-agents représentable par un MDP est d'ajouter des actions de communication à l'ensemble des actions déjà présentes dans le modèle. Ainsi, à chaque prise de décision, l'agent peut décider d'envoyer un message plutôt que de poser une action dans l'environnement. L'envoi de message entraîne généralement une récompense négative représentant le coût de la communication. Cependant, la communication apporte aux agents plus d'informations ou une meilleure collaboration entre eux ce qui leur permet de mieux performer par la suite. Le but de l'agent est alors de performer le mieux possible en choisissant un juste équilibre entre l'utilisation de la communication et l'exécution d'actions plus rentables.

Le principal problème de cette modélisation est qu'elle suppose qu'on a qu'un seul contrôleur. Or, un véhicule peut facilement être muni de plusieurs contrôleurs et un seul peut s'occuper de la communication sans nuire aux autres contrôleurs. En d'autres termes, une voiture peut à la fois accélérer, tourner et envoyer des messages aux autres voitures, elle n'est pas contrainte de choisir une seule action. Cela ne veut pas dire qu'il n'y a aucun coût à la communication et que le véhicule peut envoyer ses informations 100 fois par seconde. Comme mentionné précédemment, il existe certaines contraintes sur la rapidité d'envoi des messages et sur la bande passante totale que les véhicules doivent se partager.

De plus, il peut être difficile d'intégrer dans un seul MDP le problème du contrôle de véhicule et le problème de la communication. Cependant, ils sont étroitement liés car les effets d'une bonne politique de communication se répercuteront directement sur un meilleur contrôle du véhicule (i.e. vitesse constante en fonction du véhicule précédent, diminution de l'effet de chaîne entre les voitures, etc.). Pour séparer les deux problèmes, il fallait trouver une façon d'évaluer la performance d'une politique de communication sans impliquer le contrôle du véhicule ; autant d'aspects qu'on a préféré laisser pour de futurs travaux.

Pour contourner ce problème, nous avons toutefois opté pour la solution (provisoire) suivante. Plutôt que de fournir des informations de communication directement au module action de notre système et que celui-ci les interprète à sa guise, on ajoute un module de perceptions responsable de traiter l'information provenant de la communication. Le module de perceptions construit un modèle de l'environnement à l'aide des messages reçus par la communication et rend ce modèle disponible au module action. À mesure que des messages sont reçus par le module de perceptions, il met à jour le modèle de l'environnement. Par exemple, si la « couche action » désire savoir à quelle vitesse roule le véhicule précédent et quelle est sa position, elle accède à ses deux données directement par le module de perceptions. Elle n'a pas à fouiller à travers les messages reçus pour trouver l'information la plus récente et à extrapoler les données voulues en fonction de l'étampe de temps du message envoyé. Toute cette logique est faite dans le module de perceptions. La figure 4.7 décrit cette interaction entre le module action et le module de perceptions.

On peut ensuite comparer le modèle de l'environnement construit par le module de perceptions à l'environnement réel et évaluer l'algorithme de communication selon la marge d'erreur entre les perceptions et le monde réel. Le module de communication devient donc indépendant du module action et on peut l'optimiser en se basant sur le module de perceptions. La figure 4.8 montre le fonctionnement de cette approche. Le dernier véhicule reçoit les messages du premier véhicule et construit la perception

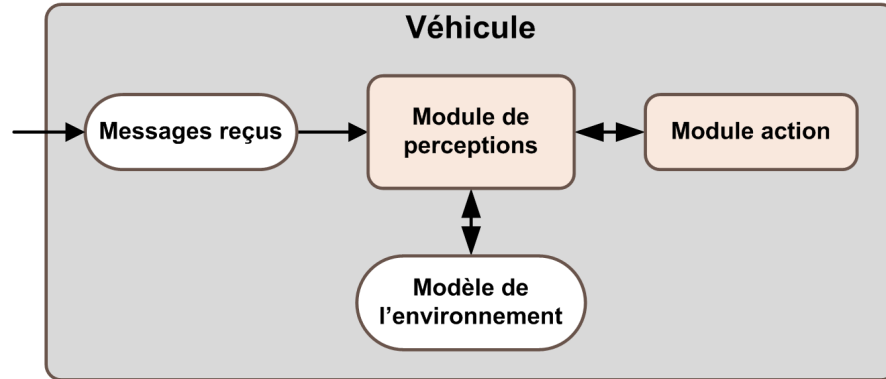


FIG. 4.7 – Interaction en le module action et le module de perceptions.

qu'il a de ce véhicule. La perception est représentée par l'image fantôme superposée au premier véhicule. Lorsque l'image fantôme couvre parfaitement le véhicule, l'erreur de la perception est nulle et le dernier véhicule ne se trompe pas sur la position du premier. Cependant, lorsque l'image fantôme est décalée, cela signifie qu'il y a une erreur au niveau de la perception.

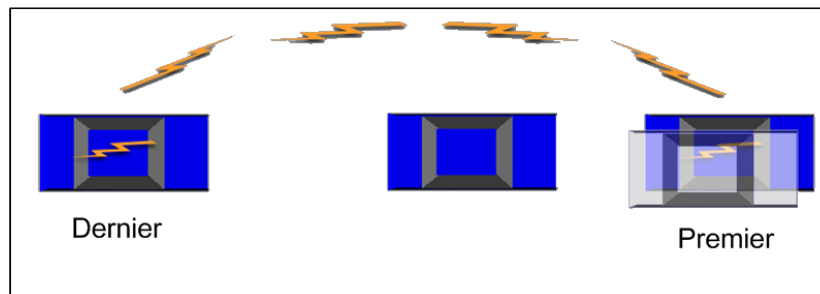


FIG. 4.8 – Perception que le dernier véhicule se fait du premier.

Les messages envoyés par les véhicules peuvent contenir plusieurs données : la vitesse, la position, la direction, l'accélération, le rayon de la courbe suivie, etc. En fait, l'état dynamique d'un véhicule à un instant t peut être représenté par sa position, un vecteur de direction et sa vitesse. Toutes les autres données sont en rapport avec l'évolution de ces variables dans le temps. Par exemple, l'accélération représente l'évolution de la vitesse dans le temps. Elle peut donc être représentée par une fonction ou par une valeur numérique simple. Dans notre cas, nous avons simplifié le problème et les messages envoyés incluent uniquement la position, la vitesse, la direction et une valeur numérique pour l'accélération.

Comme il est suggéré dans [Yang et al. \[2004\]](#), deux approches peuvent être utilisées pour gérer la communication dans des systèmes coopératifs d'évitement de collisions : une approche passive ou une approche active. L'approche passive consiste à envoyer les informations à un intervalle régulier alors que l'approche active envoie des messages suite

à des évènements, par exemple lorsque le véhicule accélère ou freine rapidement. Nous pensons toutefois que la meilleure solution peut se situer entre ces deux approches. Par exemple, un véhicule peut envoyer ses informations de façon constante et envoyer des messages lors de changements majeurs sur sa vitesse ou sa direction. Il pourrait même modifier la fréquence d'envoi des messages selon les conditions du trafic. Par exemple, lorsque les véhicules sont près les uns des autres ou qu'une situation inhabituelle se produit, les applications de sécurité ont besoin d'avoir des informations précises sur la position et le mouvement des autres véhicules. Dans ce cas, les véhicules pourraient échanger plus de messages que dans la situation où le trafic est stable et sécuritaire.

Notre approche consiste donc à faire apprendre au véhicule à quel moment il convient d'envoyer un message en fonction de son état actuel. Grâce au module de perceptions, nous avons tout ce qu'il nous faut pour modéliser le problème sous forme d'un MDP et exécuter des algorithmes d'apprentissage à l'aide de ce modèle. Le problème consiste à trouver la meilleure politique de communication en optimisant les deux aspects suivants :

1. envoyer le moins de messages possible car la bande passante est une ressource extrêmement limitée ;
2. Estimer la position d'un véhicule à travers les modules de perceptions de façon à ce qu'elle soit la plus précise possible.

La modélisation de ce problème sous forme de MDP a été faite en adoptant les variables d'état suivantes :

- le délai en secondes depuis le dernier message envoyé ;
- l'accélération ;
- la vitesse ;
- l'écart entre la vitesse actuelle et celle lors du dernier message envoyé ;
- l'écart entre l'accélération actuelle et celle lors du dernier message envoyé.

À chaque pas de temps, l'agent décide d'envoyer ou non un message contenant sa position, sa vitesse, sa direction et son accélération. La récompense est attribuée en fonction du nombre de messages envoyés et de l'erreur sur la position estimée. L'erreur, représentée en mètres, est calculée selon la distance entre la position exacte du véhicule et la position de son estimation dans le module de perceptions des autres véhicules. Comme notre implémentation des couches de bas niveau de la communication est très simplifiée, tous les véhicules voisins à l'agent reçoivent les messages de celui-ci en même temps. Ainsi, la perception qu'ils ont de sa position est la même d'un véhicule à un autre. C'est cette perception et l'erreur commune qui en découle qui sont utilisées pour

la récompense de l'agent.

De plus, la fonction utilisée par le module de perceptions pour mettre à jour les informations sur les autres voitures est la même pour tous les agents et elle est exécutée à chaque pas de temps :

1. Pour chaque message de la pile de réception de la communication :
 - (a) Vérifier l'identifiant de l'émetteur ;
 - (b) S'il existe une perception pour cet émetteur, la mettre à jour avec ces nouvelles informations ;
 - (c) Sinon, ajouter une nouvelle perception pour cet émetteur ;
2. Pour chaque perception :
 - (a) Vérifier l'étampe de temps de la perception et mettre à jour les données en extrapolant selon l'étampe de temps actuel ;
 - (b) Si le dernier message reçu pour cette perception excède un certain délai, supprimer la perception.

Le fait de supprimer une perception pour laquelle il n'existe pas de message récent permet de ne pas garder indéfiniment des estimations sur des véhicules qui ne sont plus à proximité de l'agent. L'extrapolation est très simple à faire à l'aide de calculs de base. En effet, il suffit d'appliquer l'accélération à la vitesse et la vitesse à la direction et à la position de base pour connaître la nouvelle position du véhicule.

Les résultats de l'apprentissage par renforcement sur ce modèle sont présentés dans la section 4.5.

4.4 Environnement de simulation, simulateur

Pour tester notre architecture, nous avons développé notre propre simulateur de trafic automobile où chaque véhicule est simulé individuellement et peut prendre des décisions sur sa conduite. Nous avons mis l'emphasis sur la simulation de petits groupes de véhicules interagissant entre eux dans des situations routières typiques.

Notre simulateur a été programmé en C++ et s'exécute sous la forme d'un programme console. Il prend en entrée des fichiers de configuration sous forme de texte et produit en sortie des fichiers de journalisation. Ces fichiers contiennent toutes les

informations sur la simulation, la position exacte de tous les véhicules à chaque pas de temps de la simulation et tout autres informations utiles. Un visualisateur utilisant les bibliothèques DirectX a aussi été fait pour visionner les simulations dans une interface graphique (figure 4.9). Il prend en entrée les fichiers de journalisation des simulations et affiche les routes, les véhicules et les autres objets comme les panneaux de signalisation.

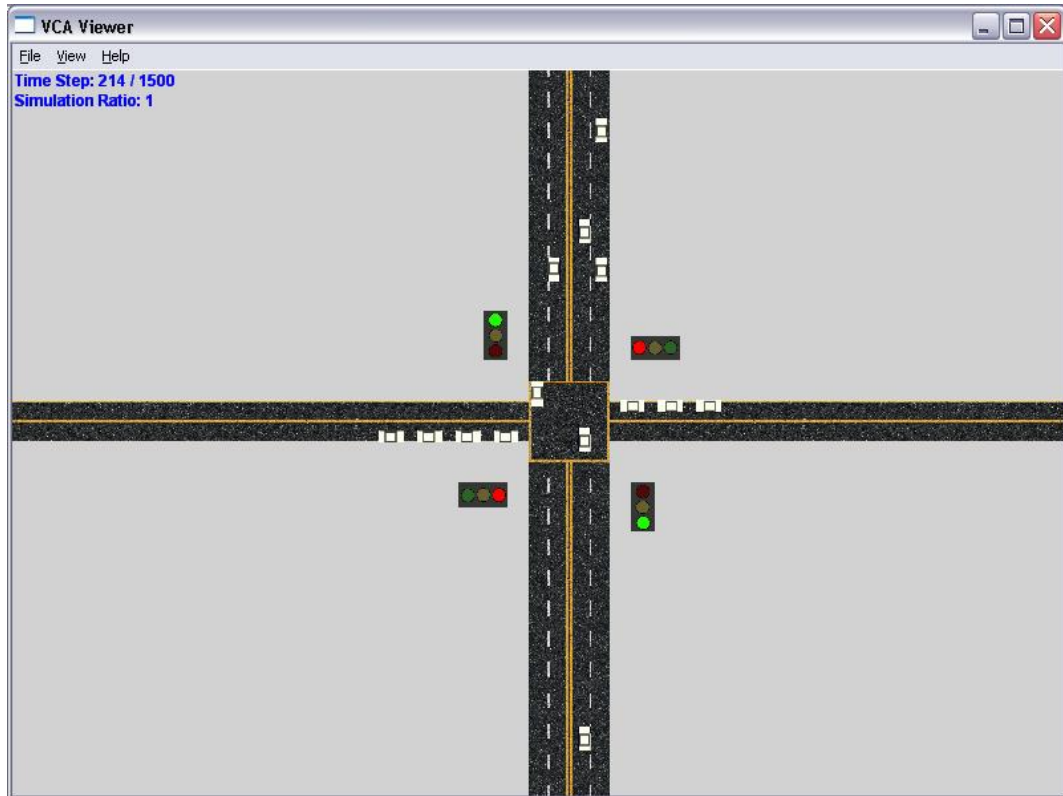


FIG. 4.9 – Visualisation d’une simulation avec intersection.

Le simulateur est discrétisé dans le temps et continu dans l’espace. Les positions des véhicules et des routes sont représentées sur un plan à deux dimensions et la discrétisation dans le temps est de 0.250 seconde. Cela signifie que la prise de décision pour chaque véhicule se fait à tous les 250 millisecondes.

L’architecture des classes est très simple. Une simulation est représentée par un environnement quelconque et un scénario. Pour faire de l’apprentissage, il faut exécuter le même scénario plusieurs fois. Chaque exécution est un épisode. L’environnement représente le modèle de l’environnement réel et contient la liste de tous les objets de la simulation. Certains objets sont statiques tels que les routes alors que d’autres sont dynamiques tels que les véhicules ou les feux de signalisation. Le scénario représente l’évolution de ces objets dans le temps. Un exemple de scénario serait une situation où trois véhicules roulent sur une même route à des vitesses différentes. Le but pourrait être de faire apprendre à un véhicule à dépasser correctement les autres véhicules.

Pour chaque épisode, le simulateur initialise l'environnement et les objets puis lance une boucle de contrôle. Chaque itération de la boucle représente un pas de temps d'une durée configurable (250 millisecondes par défaut). Chaque itération s'exécute en quatre phases : le traitement préalable au choix d'action, le choix d'action, la mise à jour de l'environnement et le traitement après-action. Pour chacune de ses phases, le simulateur boucle sur l'ensemble des objets de l'environnement et appelle la méthode correspondante à la phase en cours. Cela ressemble à un système par événements où chaque objet est averti à chacune des étapes de l'exécution et peut agir en conséquence.

Le simulateur est conçu pour exécuter en une seule fois plusieurs scénarios contenant chacun plusieurs épisodes. Bien qu'il simule chaque épisode de façon séquentielle et non en parallèle, cela est très utile pour démarrer une série de simulations coûteuses en temps sans avoir à surveiller constamment l'exécution.

Le tout est entièrement configurable. Par exemple, on peut implémenter différents modèles de dynamique pour les véhicules et changer de modèle selon le scénario voulu. Chaque véhicule utilise un modèle de dynamique, un modèle de conduite, des capteurs, un gestionnaire de communication, etc. Cette grande flexibilité en fait un outil puissant pour toutes les simulations nécessaires à nos recherches.

4.5 Résultats

Pour trouver une politique optimale de contrôle de la communication, il fallait utiliser un scénario d'apprentissage où le véhicule est placé dans différentes situations. Le scénario utilisé est un parcours d'une durée de 250 secondes où le véhicule doit accélérer, freiner, garder une vitesse constante et dépasser un autre véhicule. Le véhicule suit un trajet prédéfini qui inclut les vitesses auxquelles il doit rouler et les manœuvres auxquelles il doit faire face. Les seules actions qu'il doit choisir sont celles reliées à l'apprentissage de la communication. Ainsi, à chaque pas de temps, le véhicule peut envoyer ou non ses informations aux autres véhicules. Ceux-ci reçoivent les messages et estiment sa position dans leur module de perception.

L'algorithme Q-Learning a été exécuté sur 1000 épisodes. Comme nous pouvons le voir dans la figure 4.10, le nombre de messages décroît à mesure que l'agent apprend à communiquer. Comme la fonction de récompense pénalise beaucoup plus l'agent en cas d'erreur sur l'estimation que pour l'envoi de message, il commence sa phase d'apprentissage en envoyant beaucoup de messages. Cela lui permet de garder une erreur sur l'estimation relativement basse. Cependant, à mesure qu'il agit dans l'environne-

ment, il apprend peu à peu quels moments sont cruciaux pour l'envoi de messages et à quels moments il peut s'abstenir de communiquer sans trop altérer la perception que ses voisins ont de sa position.

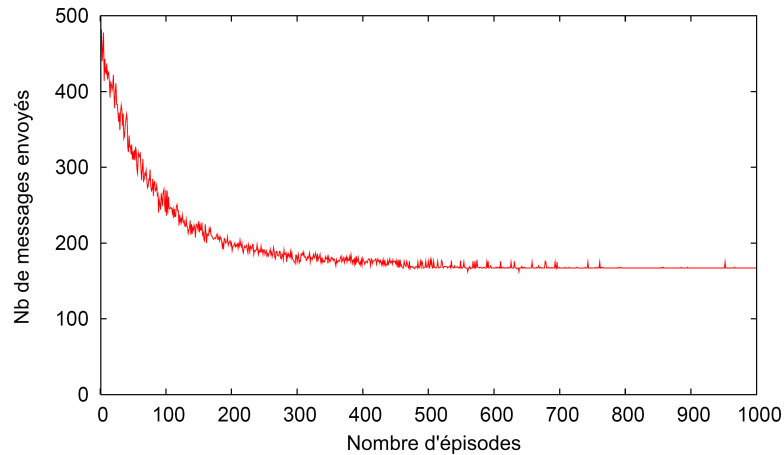


FIG. 4.10 – Nombre de messages envoyés pour 1000 épisodes de 1000 étapes.

Ainsi, on voit dans la figure 4.11 que la moyenne de l'erreur sur l'estimation reste toujours sous la barre des 80 centimètres pour chaque épisode. Au début, le véhicule envoie des messages deux fois par seconde, soit une fois par deux pas de temps. À la fin de l'apprentissage, le véhicule envoie un message par cinq pas de temps. On peut donc constater que le processus d'apprentissage diminue le nombre de messages de $2/5$ alors que l'erreur sur l'estimation augmente seulement de 20 cm, ce qui est très peu dans le cas de véhicules de 5 mètres de long se suivant à une distance d'au moins 10 mètres.

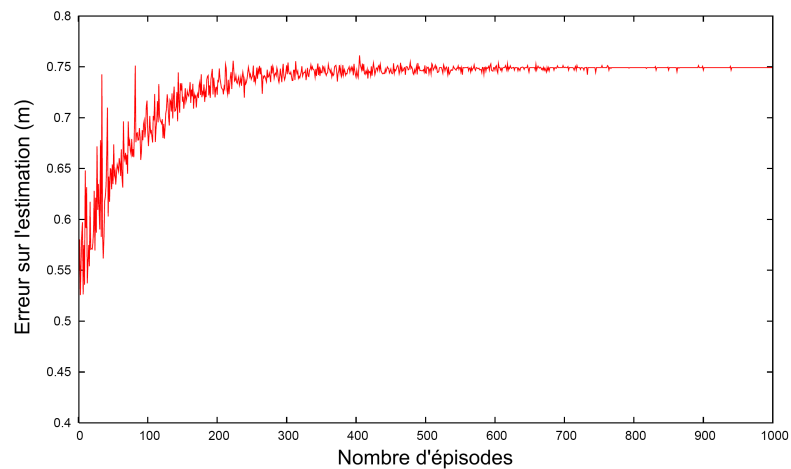


FIG. 4.11 – Erreur sur l'estimation de la position du véhicule.

4.6 Conclusion

Bien que les résultats démontrés dans ce chapitre soient préliminaires, ils démontrent bien que l'apprentissage par renforcement est une solution viable pour déterminer une politique de communication adéquate pour des applications de sécurité à bord des voitures. L'environnement complexe du trafic automobile se prête particulièrement bien aux techniques d'apprentissage machine puisqu'il est difficile de concevoir des règles de contrôle précises qui soient efficaces dans un tel milieu.

La prochaine étape serait de faire des tests avec un environnement de simulation plus réaliste sur la propagation des ondes et sur la perte de paquets de communication sans fil. Il serait aussi intéressant d'essayer d'autres algorithmes d'apprentissage avec différents paramètres pour comparer la vitesse de convergence vers une politique optimale.

Chapitre 5

Communication route-à-véhicule : solution au problème de congestion

Un autre aspect important de la communication dans le domaine des systèmes de transport intelligents est la possibilité qu'ont les véhicules de communiquer avec la route. Nous avons vu à la section 3.3 que plusieurs exemples d'applications utilisant la communication route-à-véhicule ont comme but d'améliorer la sécurité routière. Cependant, la communication entre les infrastructures du réseau routier et les véhicules permet aussi de s'attaquer à un autre problème majeur : la congestion urbaine. Dans ce chapitre, nous allons tout d'abord aborder en détail le problème du trafic et de la congestion urbaine. Nous verrons ensuite quelles sont les solutions à un tel problème et comment la communication route-à-véhicule peut être utilisée dans l'implantation de solutions concrètes.

5.1 Le trafic et la congestion

Pour plusieurs grandes villes dans le monde, le trafic et la congestion font parti du quotidien et tout laisse à croire que la situation n'ira pas en s'améliorant au cours des prochaines années. À Montréal, plusieurs émissions de radio du matin commencent plus tôt qu'il y a quelques années pour satisfaire les auditeurs qui se lèvent de plus en plus tôt pour éviter le trafic. Par exemple, la station de radio « énergie » débute son émission du matin à 5h30 à Montréal alors qu'elle ne commence qu'à 6h00 à Québec. De plus, le trafic est devenu un facteur important dans le choix d'un lieu de travail et de résidence, certaines personnes préférant une ville plus petite même si le salaire est

| Régions | 1992 | 1998 | 2005 | Régions métropolitaines | 1992 | 1998 | 2005 |
|----------------------|------|------|------|-------------------------|------|------|------|
| Canada, total | 54 | 59 | 63 | Toronto | 68 | 76 | 79 |
| Atlantique | 39 | 50 | 51 | Montréal | 62 | 65 | 76 |
| Québec | 52 | 57 | 63 | Vancouver | 70 | 68 | 67 |
| Ontario | 59 | 63 | 68 | Ottawa-Gatineau | 57 | 62 | 65 |
| Prairies | 45 | 53 | 57 | Calgary | 52 | 64 | 66 |
| Colombie-Britannique | 59 | 61 | 60 | Edmonton | 50 | 58 | 62 |
| | | | | Autre RMR/AR | 44 | 50 | 53 |
| | | | | Non RMR/Rural | 44 | 50 | 54 |

TAB. 5.1 – Temps moyen des déplacements pour faire l’aller-retour entre la maison et le lieu de travail, en minutes [Turcotte, 2005].

moins pour ne pas avoir à affronter le trafic matin et soir.

5.1.1 Étendu du problème de congestion

Au Canada, en 2005, le temps moyen de déplacement pour faire l’aller-retour entre la maison et le travail était de 63 minutes par travailleur, soit 9 de plus qu’en 1992 [Turcotte, 2005]. Cela représente près de 275 heures par année consacrées uniquement à faire la navette entre le lieu de travail et de résidence. Le tableau 5.1 donne un aperçu de cette augmentation du temps de déplacement pour différentes régions et différentes villes du Canada entre 1992 et 2005.

Aux États-Unis, les citoyens urbains passent 4,2 milliards d’heures de plus dans leurs véhicules à cause de la congestion et dépensent 2,9 milliards de gallons d’essence de plus [Schrank et Lomax, 2007]. Cela représente respectivement 220 millions et 140 millions de plus qu’en 2004, preuve que le problème ne cesse de s’aggraver. Pour chaque automobiliste américain, le coût de la congestion représente 38 heures par année et \$710 US par année. Au Canada, le coût de la congestion est évalué entre 2,3 et 3,7 milliards de dollars [Transports Canada, 2006]. Plus de 90% de cette somme est associée au temps que les travailleurs perdent dans le trafic, 7% pour l’essence consommée et 3% pour l’augmentation de l’émission de gaz à effet de serre. Si l’on regarde le coût que cela représente pour chacune des grandes villes canadiennes, on obtient les résultats du tableau 5.2. Le niveau de congestion a été calculé en utilisant le seuil de la vitesse qui reflète un pourcentage de la vitesse en condition d’écoulement libre sur une route.

D’autres villes touchées par la congestion urbaine et la pollution prennent des mesures exceptionnelles pour réduire celles-ci. C’est le cas par exemple de Mexico où le

| Zone urbaine | Au seuil de 50 p. 100 | Au seuil de 60 p. 100 | Au seuil de 70 p. 100 |
|--------------------------------------|--------------------------|--------------------------|--------------------------|
| Ville de Québec | 37,5 \$ | 52,3 \$ | 68,4 \$ |
| Montréal | 701,9 \$ | 854,0 \$ | 986,9 \$ |
| Ottawa-Gatineau (complet) | 39,6 \$ | 61,5 \$ | 88,6 \$ |
| Toronto | 889,6 \$ | 1 267,3 \$ | 1 631,7 \$ |
| Hamilton (complet) | 6,6 \$ | 11,3 \$ | 16,9 \$ |
| Winnipeg | 48,4 \$ | 77,2 \$ | 104,0 \$ |
| Calgary | 94,6 \$ | 112,4 \$ | 121,4 \$ |
| Edmonton | 49,4 \$ | 62,1 \$ | 74,1 \$ |
| Vancouver | 402,8 \$ | 516,8 \$ | 628,7 \$ |
| Total pour toutes les zones urbaines | 2 270,2 \$ | 3 015,0 \$ | 3 720,6 \$ |

TAB. 5.2 – Total annuel des coûts liés à la congestion par ville (en millions de dollars selon la valeur monétaire en 2002) [[Transports Canada, 2006](#)].

nombre de véhicules dans la région métropolitaine atteint les 4 millions. Les autorités ont ainsi mis en place un système pour réduire le nombre de véhicules circulant en ville. Ce système, nommé « Hoy No Circula » (« une journée sans circuler »), consiste à interdire la circulation à 20% des véhicules chaque jour en se basant sur le dernier chiffre de la plaque d'immatriculation. Les véhicules sont classés en différentes catégories et dépendamment du niveau d'alerte de la qualité d'air, certaines catégories ont des droits restreints de circulation [[Wikipédia, 2007](#)].

5.1.2 Causes, définitions et notions relatives au trafic

La congestion urbaine vient principalement de l'augmentation massive du nombre de véhicules personnels au cours des dernières décennies. Les infrastructures routières ont vite atteint leurs capacités maximales et la construction de nouvelles routes est limitée par des contraintes politiques, économiques, sociales et environnementales [[OCDE, 1994](#)]. Mais concrètement, quelle est la définition exacte de la congestion et comment peut-on l'expliquer ? Dans [Bull \[2004\]](#), la congestion est définie comme le résultat de la friction ou de l'interférence mutuelle entre les voitures dans le flux de trafic. À un niveau peu élevé de trafic, les véhicules peuvent circuler librement selon leurs vitesses désirées, les vitesses permises, la fréquence des intersections, etc. Cependant, à un niveau plus élevé de trafic, chaque véhicule ajouté au flux du trafic existant interfère avec ceux déjà présents. C'est ainsi que débute le phénomène de congestion. Plus formellement, « la congestion est la situation où l'introduction d'un véhicule additionnel dans le flux de

trafic augmente le temps de déplacement des autres ».

Bien qu'elle soit claire et précise, cette définition nous en apprend peu sur le trafic puisqu'il va de soi que la congestion est créée par un trop grand nombre de véhicules sur les routes. Ce qui nous intéresse, c'est de comprendre pourquoi certaines routes sont plus propices à la congestion et qu'est-ce qui cause des bouchons à des endroits inattendus. Par exemple, vous est-il déjà arrivé de devoir vous immobiliser complètement sur l'autoroute, comme si celle-ci était bloquée, puis, lorsque le trafic redevient fluide, de vous rendre compte qu'il n'y avait ni accident, ni sortie de route, bref rien pour ralentir la circulation ? Nous nous demandons toujours ce qui peut causer une telle congestion et que, cinq minutes plus tard, la circulation soit complètement fluide sans aucun changement apparent. Pour comprendre de tels phénomènes, il faut se pencher davantage sur les notions entourant la dynamique des flux de véhicules et sur les notions de congestion.

La notion de débit de trafic

Une fonction importante est le débit du trafic, qui se calcule selon l'équation 5.1. Par exemple, sur une voie où les véhicules circulent à 100 km/h (100 000 mètres/heure), avec une densité d'une voiture tous les 60 mètres (1/60 véhicule/mètre), le débit sera de 1667 véhicules/heure. Le calcul du débit est important puisqu'il mesure la capacité et l'efficacité d'une route ou d'un itinéraire donné.

$$\text{debit} = \text{vitesse} * \text{densite} \quad (5.1)$$

On peut aussi calculer automatiquement le débit maximal du trafic en fonction de la vitesse des véhicules et en tenant compte que les conducteurs gardent une certaine distance de sécurité avec le véhicule qui les précède. La figure 5.1 montre les différents débits obtenus dépendamment de si les conducteurs gardent 2 secondes, 1.5 seconde ou 1 seconde d'écart entre chaque voiture. Il faut noter que la distance recommandée dans plusieurs pays, surtout en Europe, est de 2 secondes [Bishop, 2005], mais plusieurs conducteurs gardent une distance moindre. Avec ce graphique, nous observons que le débit augmente peu une fois passé une certaine vitesse de base. Ainsi, sur un boulevard où la circulation est fluide et où les véhicules gardent un temps d'espacement de 2 secondes, le débit sera de 1565 véhicules à l'heure. Sur une autoroute fluide, où les véhicules roulent au double de la vitesse (120 km/h), le débit ne sera que de 1674 véhicules à l'heure, une mince augmentation de 109 véhicules/heure.

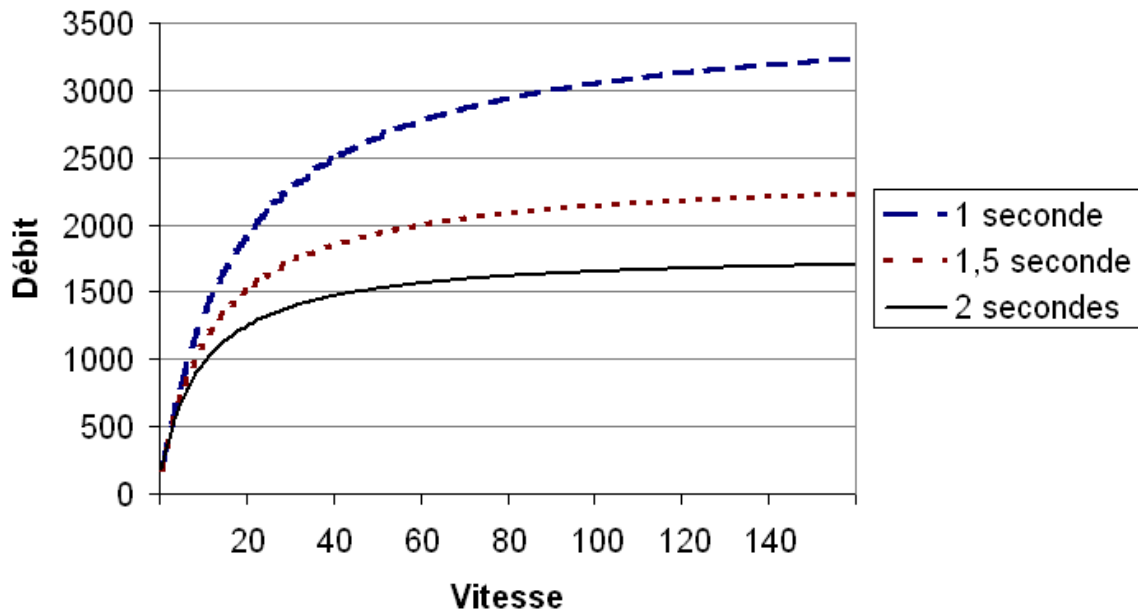


FIG. 5.1 – Débit maximal en fonction de la vitesse.

Temps de parcours

Étant donné que le débit augmente peu en fonction de la vitesse, nous pourrions en conclure que pour optimiser un itinéraire, l'augmentation de la vitesse moyenne des véhicules empruntant cet itinéraire a peu d'importance. Cependant, cela nous paraît intuitivement faux, car nous avons vu que les premiers signes de congestion sont principalement le ralentissement du trafic. En effet, il ne faut pas oublier une notion importante qui est le temps de parcours de chaque automobiliste. En reprenant l'exemple précédent, il faut noter que le véhicule roulant à 60 km/h prendra ni plus ni moins que le double du temps du véhicule roulant à 120 km/h pour atteindre sa destination. Il est donc important de garder en tête qu'il ne faut pas seulement optimiser le débit du trafic mais aussi le temps de parcours moyen des véhicules empruntant le réseau routier.

Effet de chaîne

Plusieurs facteurs et phénomènes viennent détériorer la fluidité du trafic. Un premier phénomène important est l'effet de chaîne causé par les erreurs humaines [Liang et Peng, 2000]. Lorsqu'un véhicule ralentit pour une raison quelconque, le conducteur qui le suit doit freiner et ajuster sa vitesse en conséquence. Cependant, un conducteur humain ne

peut savoir exactement à quelle vitesse va son prédécesseur et choisira pendant quelques instants une vitesse moindre que celui-ci. Le troisième véhicule réagit à son tour selon sa perception du deuxième véhicule, augmentant ainsi l'écart de vitesse. L'erreur se propage de cette façon jusqu'au dernier véhicule de la file.

Si le trafic est moindre, ces erreurs ne posent pas problème et sont vite « absorbées » par les espaces plus grands déjà existants entre les voitures. Cependant, si le trafic est plus élevé et que toutes les voitures se suivent à une distance sécuritaire minimale, une réaction en chaîne se produit et l'erreur est propagée comme une vague jusqu'à ce que le dernier véhicule soit obligé de s'immobiliser complètement. Ainsi, un seul véhicule entrant sur une autoroute dont le trafic est déjà au maximum de la capacité et forçant un autre véhicule à ralentir pour le laisser passer peut entraîner un bouchon à plusieurs centaines de mètres en arrière. Cela explique en partie pourquoi il nous arrive de devoir nous arrêter complètement sur l'autoroute même s'il n'y a ni accidents, ni travaux ou autres.

Goulot d'étranglement

Une autre notion importante sur le débit du trafic est le principe de goulot d'étranglement. Cela se produit lorsqu'il y a un rétrécissement, un passage à une seule voie, une accentuation de la pente, un accident, etc. Si le volume de trafic demeure en dessous de la capacité d'écoulement du goulot, l'allongement du temps de parcours sera faible. Cependant, si la demande de trafic excède durablement la capacité du goulot, les délais résultants seront importants. De cela découle un théorème important du trafic routier : le débit maximum d'un itinéraire s'égalise à celui de son point critique [Derycke, 1997]. Cela signifie que pour calculer le débit maximum d'un itinéraire, il faut d'abord calculer le débit de chaque tronçon de l'itinéraire puis de chaque jonction. On prend ensuite le débit le moins élevé et ce nombre représente le débit maximum de l'itinéraire. Prenons par exemple une autoroute à deux voies ayant un débit de 3000 véhicules à l'heure. Supposons qu'un tronçon de cette autoroute est en réparation et ne compte qu'une seule voie ayant un débit de 1500 véhicules à l'heure. Le débit maximum de l'autoroute (incluant ce tronçon) sera alors de 1500 véhicules à l'heure.

Relation demande-vitesse-débit

Nous avons vu plus tôt que le débit augmente en fonction de la vitesse et de la densité des véhicules. Cependant, nous venons aussi de voir que plus la densité est

élevée, plus il y a d'interactions entre les voitures, plus la congestion augmente et plus le débit diminue. Cela met en relation la capacité de la route, la demande des usagers, la vitesse de ceux-ci et le débit résultant.

Dans une situation de trafic fluide, lorsque les véhicules sont peu nombreux, ils peuvent se déplacer à la vitesse désirée sans problème. Cette vitesse varie en fonction des performances de chaque véhicule, des caractéristiques de la route, des conditions atmosphériques et est limitée par la vitesse légale. Bien entendu, la vitesse diminue à mesure que la circulation s'intensifie. Ainsi, une augmentation de la demande entraînera d'abord une légère diminution puis, progressivement, une diminution plus forte de la vitesse instantanée [Derycke, 1997]. Les figures 5.2 et 5.3 montrent les relations du débit et de la vitesse en fonction de la demande.

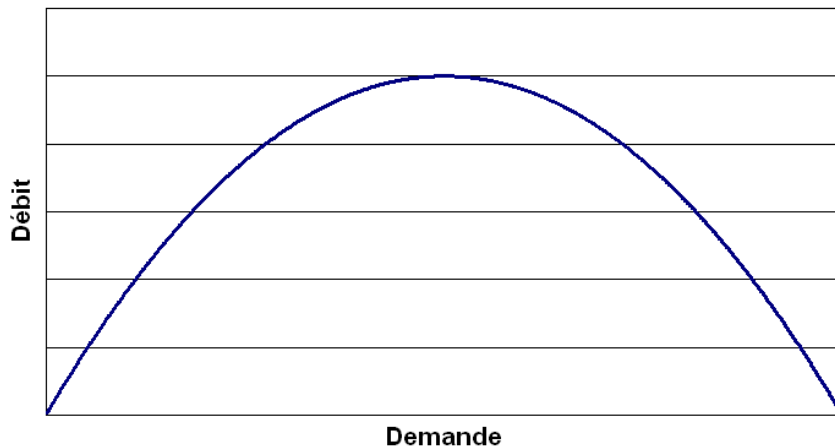


FIG. 5.2 – Débit en fonction de la demande.

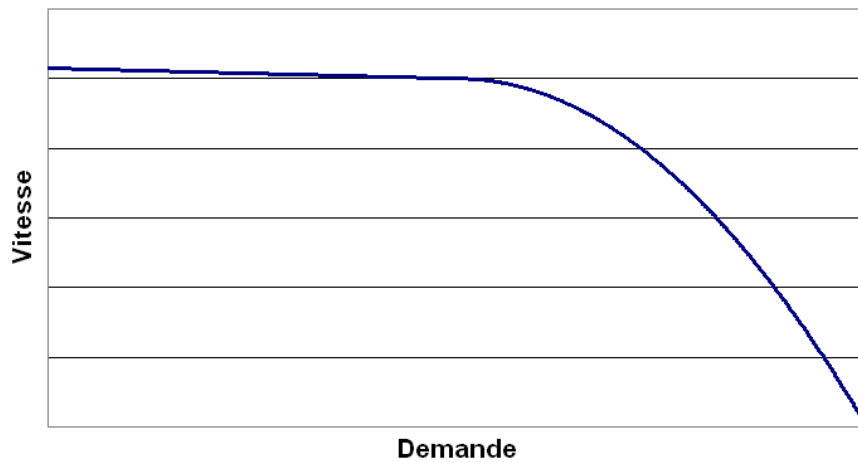


FIG. 5.3 – Vitesse en fonction de la demande.

5.1.3 Solutions au problème de congestion

Pour améliorer l'efficacité des routes, le débit du trafic et le temps de parcours, plusieurs solutions ont été analysées, testées et implantées. Plusieurs autres solutions sont encore à l'étape de la recherche et de la conception.

Systèmes d'aide à la conduite

Le premier type de solutions tourne autour des systèmes d'aide à la conduite. Par exemple, les régulateurs de vitesse, en plus d'augmenter le confort des conducteurs, ont aussi pour but de régulariser le flux du trafic en maintenant les véhicules à une vitesse constante. Dans la même optique, les régulateurs de vitesse adaptatifs (« Adaptive Cruise Control » (ACC)) permettent d'augmenter considérablement le débit du trafic en réduisant l'écart entre chaque voiture. En effet, la réaction d'un tel système est beaucoup plus rapide que celle d'un conducteur humain et est beaucoup plus précise. Cela permet donc de laisser moins d'espace entre les voitures tout en étant sécuritaire. Cela permet également de diminuer l'effet de chaîne expliqué à la section 5.1.2 en éliminant les répercussions d'erreurs.

Un autre type de systèmes que l'on retrouve dans les véhicules est constitué des systèmes d'information sur le réseau routier. Les systèmes de navigation GPS en sont un bon exemple et ont envahi le marché au cours des dernières années. Certains de ces systèmes permettent même de télécharger en temps réel, par l'entremise d'un appareil cellulaire, les informations les plus récentes sur l'état du trafic. Cela permet donc aux conducteurs de choisir le chemin le plus rapide en fonction de la congestion et de désengorger les sections les plus critiques.

Feux de gestion d'accès

Les feux de gestion d'accès, plus connus sous le nom de « ramp meter » [Zhang et Ritchie, 1997; Papageorgiou et Kotsialos, 2000] en anglais, ont pour but d'améliorer la sécurité et la fluidité des autoroutes en contrôlant le nombre de véhicules qui y entre. Ces feux sont installés sur les bretelles d'accès et limitent le nombre de véhicules pouvant accéder à l'autoroute dépendamment du trafic déjà présent. Cela permet de garder un débit et une vitesse moyenne plus élevée en diminuant la demande, comme il a été vu à la section 5.1.2.

| En provenance de | Avec feu de gestion d'accès | Sans feu de gestion d'accès |
|-------------------|-----------------------------|-----------------------------|
| Boul. Champlain | 12 minutes | 3 minutes |
| Aut. Henri-IV Sud | 4 minutes | 11 minutes |

TAB. 5.3 – Temps d'attente des usagers à l'entrée du pont Pierre-Laporte [[Transport Québec, 2008b](#)].

En 2002, ce type de feux a été installé à l'entrée de quatre bretelles des autoroutes Jean-Lesage et Henri-IV à Québec pour contrôler le trafic à l'entrée du pont Pierre-Laporte. Cela a eu pour conséquence de grandement améliorer la fluidité du trafic sur le pont [[Transport Québec, 2008b](#)]. De plus, cela a incité plusieurs automobilistes à prendre un autre trajet, autrement dit le pont de Québec. Ainsi, grâce à ces feux, le débit de circulation journalier du pont de Québec a augmenté d'environ 35%, soit plus de 40 000 véhicules par jour. Les usagers circulant sur l'autoroute Henri-IV Sud ont vu quotidiennement leur temps d'attente diminué d'environ 65% à l'entrée du pont Pierre-Laporte. Cette diminution est de 45% pour l'ensemble des usagers empruntant ce pont.

Dans le tableau 5.3, on remarque que le temps d'attente des usagers en provenance du boulevard Champlain et voulant emprunter le pont Pierre-Laporte a augmenté considérablement. Cependant, le temps d'attente de ceux provenant d'Henri-IV Sud a diminué dans les mêmes proportions. Ces résultats sont excellents puisque 1000 véhicules à l'heure arrivent du boulevard Champlain contre 5000 véhicules à l'heure pour d'Henri-IV Sud.

Carrefours giratoires

Un carrefour giratoire est un aménagement comprenant une, deux ou trois voies de circulation entourant un îlot central [[Transport Québec, 2008a](#)]. La circulation dans ces voies se fait dans le sens contraire des aiguilles d'une montre. De plus, les usagers qui veulent s'engager dans un carrefour doivent céder le passage aux piétons et aux véhicules qui y circulent déjà. Depuis plusieurs années, il connaît beaucoup de succès dans des pays tels l'Angleterre, la France, la Suisse, le Danemark, la Norvège et l'Australie, étant donné qu'il s'agit d'un concept d'aménagement routier dont les avantages sont la réduction de la vitesse, la diminution du nombre et de la gravité des accidents, ainsi que la gestion efficace de la circulation. En Amérique du Nord, les carrefours giratoires sont apparus plus tard, mais leur nombre a augmenté au cours des dernières années, notamment dans les États américains du Colorado et du Maryland.

La présence d'un carrefour giratoire accroît grandement la fluidité de la circulation puisque les véhicules n'ont pas à effectuer un arrêt complet pour traverser l'intersection. Cependant, certaines conditions doivent être respectées pour qu'une telle solution soit avantageuse [Bull, 2004] :

1. L'intersection doit avoir cinq branches ou plus avec plus ou moins le même volume de trafic sur chaque branche ;
2. Le flux de trafic voulant tourner doit être plus grand que le flux voulant aller tout droit ;
3. Il doit y avoir assez d'espace pour aménager un carrefour de cette taille ;
4. Il doit y avoir peu de piétons.

Offre et la demande

Plusieurs autres solutions ont été proposées ou implantées pour réduire la congestion. Un premier type de solutions est de faire diminuer la demande. Par exemple, l'installation de postes de péage peut inciter les usagers à diminuer leurs déplacements ou à utiliser des moyens alternatifs comme le covoiturage ou les transports en commun. On peut aussi améliorer les services de ces moyens alternatifs en ajoutant des parcours, en créant des voies réservées et en offrant des stationnements bon marché près des terminus.

D'autres solutions peuvent jouer sur l'offre comme les feux de gestion d'accès et les carrefours giratoires discutés plus tôt. Ces moyens améliorent ou augmentent l'offre en permettant une meilleure efficacité du réseau routier. Une solution utilisée dans plusieurs villes est l'implantation de voies inversibles. Ces voies peuvent servir aux autobus ou encore à tous les usagers et sont orientées dans la direction du trafic. Par exemple, sur une route à 5 voies dont une est inversible, nous aurions 3 voies le matin en direction du centre-ville et 3 voies le soir en direction de la banlieue, contre 2 dans le sens opposé.

Optimisation des feux de signalisation

Finalement, la solution sur laquelle nous nous penchons dans ce mémoire est l'optimisation des feux de signalisation. Le but est de choisir les phases et leurs durées de façon à ce qu'un maximum de voitures puisse circuler à une intersection. Il faut de

plus synchroniser les feux des différents carrefours pour améliorer la fluidité du trafic dans une direction spécifique. Ceci n'est pas une tâche triviale et l'explication de cette problématique fait l'objet de la prochaine section.

5.2 Gestion des intersections et des feux de signalisation

Les réseaux routiers sont composés de segments et de jonctions. Les segments sont les routes permettant de se déplacer dans diverses directions alors que les jonctions permettent aux véhicules de passer d'une route à l'autre, selon leur itinéraire. Il existe différents types de jonction dépendamment du type de routes qu'elles relient, du trafic qui y circule et de l'emplacement physique de celles-ci. Par exemple, une jonction peut prendre la forme d'une rampe d'accès pour entrer sur l'autoroute, d'un carrefour giratoire ou d'une intersection avec des arrêts pour toutes directions.

Le problème des intersections est qu'elles constituent des goulots d'étranglement pour les routes sur lesquelles elles sont placées. En effet, pour éviter les collisions et assurer la sécurité des véhicules qui empruntent l'intersection, des systèmes doivent être mis en place pour gérer la circulation aux jonctions du réseau. Que ce soit des panneaux d'arrêts, des panneaux pour céder le passage ou des feux de signalisation, ces systèmes dictent la priorité de passage et indiquent qui peut entrer sur l'intersection et sous quelles conditions.

L'intersection constitue donc un goulot d'étranglement pour les routes menant à l'intersection (à l'exception de celles qui ont la priorité absolue en tout temps). Comme il est mentionné à la section 5.1.2, une demande de trafic excédant durablement la capacité d'écoulement du goulot provoquera des délais importants pour les véhicules empruntant l'intersection. De plus, étant donné que le débit maximum d'un itinéraire s'égalise à celui de son point critique, il est primordial d'optimiser la circulation aux intersections pour améliorer la fluidité globale du réseau.

5.2.1 Conception d'une intersection

Plusieurs facteurs doivent être pris en compte pour la conception d'une intersection. On doit en particulier considérer :

- Le types de routes qu'elle relie : autoroute, route nationale, route régionale, boulevard, rue résidentielle, etc ;
- La quantité de trafic qui y circule ;
- Le nombre de voies sur chacune des routes ;
- La position physique des routes : emplacement géographique, directions des routes, superficie disponible pour la construction de l'intersection, etc ;
- Les facteurs administratifs (financement, etc).

À partir de ces facteurs, les experts choisissent le type de jonction qui correspond à la situation. Par exemple, sur une autoroute, on ne retrouve généralement aucun carrefour à niveau, sauf peut-être à ses extrémités. Un viaduc et des rampes d'accès sont donc utilisés dans la majorité des jonctions entre une route et une autoroute.

Si le type de jonction choisi est une intersection avec des feux de signalisation, il faut concevoir le système de contrôle des feux.

5.2.2 Contrôle des feux de signalisation

Le premier aspect à prendre en compte dans le contrôle d'un feu est les actions possibles que les véhicules peuvent faire à l'intersection. Un contrôleur possède un plan de signalisation qui décrit les différentes phases disponibles. Une phase consiste en la description des différentes transitions possibles que les véhicules peuvent effectuer pour chaque voie de l'intersection [Button et Hensher, 2001]. Les phases décrivent aussi les lumières qui doivent être affichées aux véhicules pour leur indiquer les droits de passage. Évidemment, les transitions permises dans chacune des phases doivent être non conflictuelles. Pour de grandes intersections, la conception est généralement faite par un expert du trafic à l'aide de logiciels conçus pour cette tâche. Nous reviendrons sur la modélisation des intersections au chapitre 6.

Il existe deux façons différentes de gérer les changements de phases d'un plan de signalisation. Les contrôleurs à temps fixe sont intuitivement simples et fonctionnent avec l'heure de la journée. C'est le choix de contrôle le moins coûteux et le plus logique pour un réseau routier où le trafic est stable et prévisible. L'inconvénient principal et que son efficacité peut devenir très faible lorsque le flux de trafic est instable. Cela peut se produire pour plusieurs raisons comme des accidents, de mauvaises conditions météorologiques, etc. Dans ce cas, l'utilisation de contrôleurs avec capteurs peut aider à résoudre le problème. Ces contrôleurs utilisent, par exemple, des détecteurs enfouis sous les voies ou des caméras pour connaître l'état du trafic.

5.2.3 Problématique de l'optimisation des feux de signalisation

La problématique de l'optimisation des feux de signalisation se divise en deux. Tout d'abord, il faut avoir le plus d'informations possible sur l'utilisation du réseau routier pour pouvoir décider de la programmation des feux. Cela n'est pas une mince tâche car cette utilisation est définie par une multitude d'usagers empruntant un parcours distinct à des heures variables. Deuxièmement, même avec ces informations, il faut faire des choix qui avantagent certains automobilistes au détriment des autres. Bien sûr, il faut essayer d'avantager la fluidité sur les parcours les plus en demande. Cependant, à quel point faut-il désavantager les autres usagers et que faire lorsque deux parcours très achalandés entre en conflit à une ou plusieurs intersections ?

Si l'on prend une seule route et que l'on synchronise les feux dans une direction, on se rend compte que cela est désavantageux pour les voitures circulant en sens inverse. De plus, la synchronisation d'une artère influence les feux de signalisation sur l'intersection de chacune des routes croisant cette artère. En d'autres mots, la synchronisation d'une artère influence la synchronisation de toutes les autres artères qui la croisent. Cet effet de domino dû à l'interrelation des intersections empêche la synchronisation parfaite des feux sur toutes les routes du réseau.

Un autre problème est l'aspect dynamique de l'environnement. Le flux de véhicules et la demande varient en fonction de l'heure de la journée, du jour de la semaine et de la période de l'année. Cependant, même en tenant compte de ces aspects, la programmation des feux peut devenir inefficace lors d'évènements spéciaux ou de changements temporaires comme la fermeture de routes pour cause de construction ou autre.

5.2.4 Simulateurs de trafic

Il existe plusieurs simulateurs pour modéliser ce problème d'optimisation des feux et tester des solutions, chacun pouvant être placé dans l'une des trois classes suivantes [Burghout, 2004] :

1. Microscopique
2. Macroscopique
3. Mesoscopique

Les simulateurs microscopiques simulent chaque véhicule du réseau : l'accélération, le freinage, les changements de voies, etc. Pour modéliser le trafic avec précision, plusieurs facteurs doivent être pris en compte ce qui nécessite un grand nombre de paramètres de simulation. Les simulateurs microscopiques sont donc plus précis et réalistes mais demandent généralement plus de temps de calcul.

Les simulateurs macroscopiques sont plus orientés sur le flux du trafic que sur les véhicules eux-mêmes. Ils ne simulent pas chaque véhicule mais plutôt l'état du trafic en calculant la densité des véhicules, leur vitesse moyenne, etc.

Les simulateurs mesoscopiques se veulent un mélange des deux autres types. Ils ont pour but de gérer un certain nombre de détails tout en étant capables de traiter avec des réseaux plus gros. Le réseau peut être découpé en cellules où les véhicules entrent et sortent à un temps donné. Ce temps est calculé en fonction de la structure des routes internes à la cellule, du nombre de véhicules dans la cellule et de la vitesse à laquelle entre le véhicule.

5.3 Approche par la communication et les agents

L'approche proposée dans ce mémoire est d'attaquer le problème de l'optimisation des feux de signalisation en utilisant la théorie multi-agents ainsi que la communication route-à-véhicule et la communication entre les intersections. Le but est de placer un agent à chaque intersection pour contrôler les feux de signalisation de ce carrefour. Pour cela, on suppose que les véhicules sont munis de systèmes GPS ainsi que de systèmes de communication sans fil. Ils sont donc en mesure de communiquer à l'infrastructure leurs positions et leurs destinations, ainsi que le parcours qu'ils comptent emprunter.

Avec ces informations, les agents placés aux intersections doivent calculer la stratégie optimale pour améliorer la fluidité du trafic à leur carrefour et doivent communiquer entre eux pour optimiser le flux global du trafic.

5.4 État de l'art sur l'optimisation des feux de signalisation

Plusieurs travaux ont été faits sur l'optimisation des feux de signalisation. Cette section en présente quelques-uns.

Les contrôleurs de feux de signalisation pour une seule jonction peuvent être séparés en deux catégories : les contrôleurs à temps fixe et les contrôleurs réactifs. Les contrôleurs à temps fixe utilisent généralement des programmations qui correspondent à différents modèles de trafic et bouclent à travers celles-ci au cours de la journée. Dans cette catégorie, [Improta et Cantarella \[1984\]](#) ont proposé d'optimiser le contrôleur d'une intersection à l'aide de programmation linéaire dans le but de minimiser le délai. Les contrôleurs réactifs utilisent quant à eux des données de trafic récoltées en temps réel par des capteurs. Dans cette catégorie, [Vincent et Young \[1986\]](#) ont proposé un contrôleur qui fait des ajustements dynamiques aux temps de ses cycles et à la séparation de ses phases.

À l'inverse des contrôleurs ne gérant qu'une seule jonction se trouvent les méthodes de contrôle centralisé. On trouve dans cette catégorie les méthodes TRANSYT, SCATS and SCOOT [[Button et Hensher, 2001](#)]. [Robertson et Bretherton \[1991\]](#) ont étudié et évalué la méthode SCOOT en calculant le débit des véhicules, la longueur moyenne de la file d'attente et le nombre de véhicules arrêtés. Les données sont emmagasinées en temps réel à partir des capteurs et, à partir du modèle de trafic mis à jour, SCOOT fait quelques modifications au plan de coordination précédent. Cette méthode a montré des délais réduits en Angleterre par rapport aux feux de signalisation à temps fixe.

Quelques chercheurs en informatique ont aussi étudié l'utilisation d'agents intelligents et d'apprentissage pour le contrôle des feux de signalisation. Par exemple, [Wiering \[2000\]](#) a proposé l'apprentissage par renforcement basé sur un principe de modèles pour que les feux de signalisations trouvent la stratégie de contrôle optimale. Son approche est basée sur une modélisation hautement simplifiée du flux de trafic. Il expérimente aussi le « co-learning » où les véhicules partagent les mêmes fonctions de valeurs que les feux de signalisation et tentent d'apprendre le trajet optimal pour atteindre leurs destinations.

[Dresner et Stone \[2006\]](#) ont traité le problème différemment et ont suggéré d'utiliser un système de réservations aux intersections pour éviter les collisions. Dans ce système, les véhicules font une demande à un agent central, l'intersection. Si la requête est acceptée, le véhicule doit suivre le trajet prescrit avec la garantie d'être en sécurité durant

la traversée de l'intersection. Les conflits sont résolus par l'intersection et les véhicules qui ne peuvent pas respecter le plan suggéré doivent s'arrêter à l'intersection. L'idée est très intéressante mais n'est pas très adaptable à la présence de conducteurs humains. De plus, ses recherches ne sont pas axées sur la coordination entre les intersections, un des principaux problèmes dans l'optimisation des feux de signalisation.

La coordination de plusieurs agents aux intersections a été explorée par [Bazzan \[2005\]](#) où la théorie des jeux évolutionnaires a été appliquée pour laisser les agents apprendre à coopérer. Cette approche exige la présence d'un système de communication entre les agents pour partager leurs choix d'action locale. De plus, avec l'aide de [da Silva et al. \[2006\]](#), l'auteur s'attaque à l'aspect non stationnaire du problème. Pour détecter les changements dans le flux du trafic, les auteurs ont développé un algorithme de détection de contexte qui apprend et utilise les différentes stratégies de contrôle dépendamment du flux de trafic détecté. Cette technique semble particulièrement utile pour gérer le fait que différents flux de trafic sont souvent observés dépendamment de la période la journée, par exemple à l'heure de pointe le matin et le soir. L'algorithme détecte ces changements et s'adapte aux conditions présentes.

Thorpe, à travers différentes recherches [[Thorpe et Anderson, 1996](#); [Thorpe, 1997](#)], a mis l'accent sur l'application de l'apprentissage de plusieurs feux de signalisation placés sur une grille. Il démontre à travers ses travaux quels sont les problèmes à résoudre pour faire un système de lumières adaptatives basées sur l'apprentissage par renforcement et sur les agents autonomes.

Finalement, d'autres techniques d'intelligence artificielle comme la programmation dynamique ont aussi été utilisées pour optimiser la coordination des feux de signalisation. En particulier, [Heung et al. \[2005\]](#) a montré qu'il était possible de réduire le délai d'attente des véhicules avec cette technique, particulièrement quand la demande atteint la capacité de la jonction.

5.5 Conclusion

Dans ce chapitre, nous avons vu en quoi consiste le problème de la congestion urbaine et les coûts qui y sont reliés pour plusieurs villes en Amérique du Nord. Nous avons aussi vu différentes notions relatives au trafic et aux flux de véhicules. Une des solutions pour diminuer la congestion urbaine est d'optimiser la circulation aux intersections. Ceci n'est pas une tâche simple et nous verrons au prochain chapitre comment ce problème peut être traité.

Chapitre 6

Optimisation des feux de signalisation

Le chapitre précédent a montré clairement que la congestion est un problème réel dans le quotidien d'une majorité de citoyens urbains. De plus, les intersections, et plus particulièrement celles munies de feux de signalisation, représentent des goulots d'étranglement pour la circulation. Il existe plusieurs techniques pour améliorer l'efficacité de ces intersections dont plusieurs entrent en jeu lors de la conception de celle-ci. Cependant, comme la demande des usagers et le modèle du trafic changent constamment, il est difficile d'opérer des feux de signalisation de façon optimale. La solution proposée vise à utiliser la communication inter-véhicules ainsi que la théorie agent pour récolter de l'information sur le trafic et contrôler les feux de signalisation.

Ce chapitre expose d'abord en détail la problématique et la complexité du problème de l'optimisation des feux de signalisation. Nous verrons ensuite comment ce problème peut être représenté par un modèle plus théorique pour lequel on peut utiliser plus aisément des algorithmes. Finalement, nous ferons état des différentes approches étudiées au cours de cette maîtrise et les résultats auxquels ils ont donné lieu.

6.1 Problème et modélisation

Comme il a été vu au chapitre 4, la modélisation d'un problème sous forme plus théorique permet d'utiliser des algorithmes et des techniques d'apprentissage machine pour résoudre le problème. Dans le cas des feux de signalisation, l'agent qui contrôle

| Modèle MDP | Contrôle des feux |
|--------------------------------|---|
| S un ensemble d'états | L'ensemble des états possibles de la circulation ; l'état du trafic : nombre de voitures sur chaque tronçon, vitesse de celles-ci, etc. |
| A un ensemble d'actions | L'ensemble des combinaisons de lumières que l'agent peut choisir. |
| T une fonction de transition | Les changements observés dans la circulation après avoir choisi un jeu de lumière spécifique ; le passage d'un état du trafic à un autre. |
| R une fonction de récompense | Une fonction basée sur les mesures de performances telles que le délai de parcourt des usagers et le débit du trafic à l'intersection. |

TAB. 6.1 – Problème du contrôle des feux sous forme de MDP.

une intersection doit observer l'environnement, c'est-à-dire l'état du trafic, et poser des actions en contrôlant l'état des feux de signalisation. Le tableau 6.1 montre un aperçu de la modélisation du problème sous forme de processus décisionnels de Markov (MDP). Chaque élément de l'ensemble d'états décrit le trafic à un instant précis. Il peut s'agir par exemple du nombre de voitures sur chaque tronçon de route. L'ensemble d'actions contient les actions offertes à l'agent pour contrôler les lumières. La fonction de transition reçoit en paramètre l'état du trafic ainsi que l'action de l'agent et retourne les probabilités de se retrouver dans chacun des états suivants. C'est cette fonction que l'agent tente d'apprendre pour choisir l'action menant au meilleur état possible. La fonction de récompense indique à l'agent si l'état obtenu est désirable ou non. Elle se base sur des mesures de performances telles que le délai de parcours des usagers et le débit de trafic à l'intersection.

6.1.1 Complexité du problème

Modéliser et résoudre le problème de l'optimisation des feux de signalisation est une tâche très complexe pour deux raisons majeures : l'espace d'états de chaque intersection est représenté par un ensemble énorme de variables continues et le choix d'actions est composé d'une multitude de paramètres.

Prenons par exemple toutes les variables qui représentent l'état du trafic dans une ville quelconque ou qui sont susceptibles de l'influencer. Pour commencer, on peut y inclure toutes les voitures et leurs états dynamiques respectifs. Cela inclut la position,

la vitesse, l'accélération, la direction et tous les autres paramètres rattachés au mouvement de chaque voiture. On pourrait aussi inclure l'état statique tel que le poids, la taille, la force du moteur, car ces paramètres influencent la vitesse d'accélération et de freinage des véhicules. Ensuite, il faudrait inclure tous les paramètres climatiques pouvant influencer les conditions routières, le trajet de chacun, etc.

Il est vrai qu'il est exagéré d'inclure tous ces paramètres dans l'espace d'états d'une seule intersection et même dans l'état complet du système. Cependant, cela démontre que si l'on veut absolument tenir compte de tous les paramètres influençant l'état du trafic, le problème devient pratiquement impossible à résoudre. La question est plutôt de savoir : qu'est-ce que nous gardons dans l'espace d'états pour avoir une assez bonne description de l'état du trafic sans que cela devienne impossible à calculer par des machines ?

Une bonne solution pour diminuer l'espace l'état du problème est d'inclure plusieurs facteurs dans une même variable. Par exemple, l'ensemble des caractéristiques statiques des véhicules influence le flux du trafic en jouant sur la vitesse à laquelle les voitures démarrent, atteignent leurs vitesses maximales et freinent. Il en est de même pour les conditions climatiques qui peuvent forcer les automobilistes à garder une certaine distance entre les voitures, à réduire leurs vitesses ou à limiter leurs capacités d'accélération [Agarwal *et al.*, 2005]. Toutes ces variables influencent le débit du trafic, sa densité et la vitesse moyenne des véhicules. Ainsi, une solution serait de représenter l'état du trafic du réseau routier par le débit et la vitesse moyenne des véhicules sur chaque segment. L'agent tient uniquement compte de ces données pour prendre des décisions (ce qui lui facilite la tâche) et le simulateur gère les autres facteurs (tels que la dynamique des véhicules) pour garder un bon niveau de réalisme.

Une autre solution serait de simplifier le problème directement plutôt que de simplifier uniquement l'espace d'états de l'agent. La figure 6.1 montre une modélisation où l'environnement réel a été simplifié au niveau de la simulation. L'agent apprend dans ce monde simplifié et le modèle qu'il se fait de son environnement ressemble beaucoup au modèle d'environnement du simulateur. Le désavantage d'une telle approche est que l'agent apprendra une politique optimale, assurant ainsi de bons résultats uniquement dans un monde simplifié. Dès lors, il est difficile de savoir si cette même politique serait efficace dans le monde réel. La figure 6.2 montre une modélisation plus réaliste de l'environnement au niveau du simulateur. L'agent évolue dans ce monde complexe (mais quand même plus simple que la réalité) et se fait son propre modèle simplifié de l'environnement pour la prise de décision. La politique qu'il apprend est alors plus proche du monde réel. C'est sur cette deuxième façon de modéliser que sont basées les approches présentées dans ce mémoire.

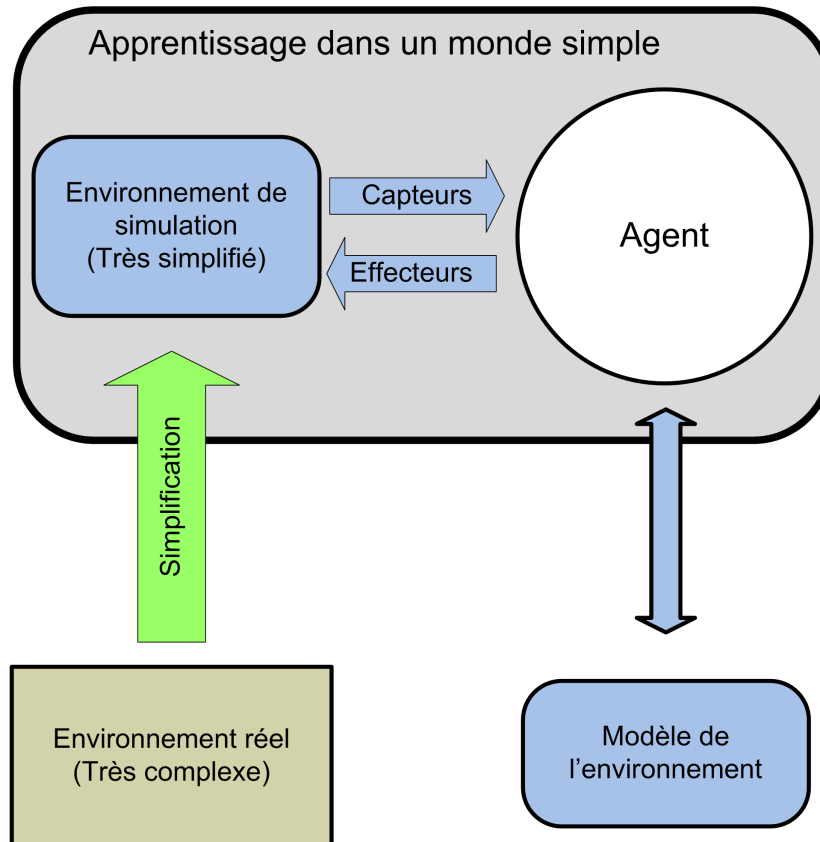


FIG. 6.1 – « Simplification » de l’environnement réel par un environnement simulé : l’agent apprend à performer dans l’environnement simplifié-simulé.

6.1.2 Modélisation d’une intersection

Cette section montre comment il est possible de modéliser l’environnement d’une intersection sous forme de structures mathématiques. Il y a plusieurs façons de représenter le problème, toutefois l’approche adoptée ici semble être assez « intuitive ».

Une intersection représente le croisement d’une ou plusieurs routes. Chaque route peut avoir une ou plusieurs voies et peut être à sens unique ou à double sens. Les voies qui permettent aux voitures d’entrer dans l’intersection sont appelées les voies sources. Les voies permettant aux voitures de quitter l’intersection sont appelées les voies de destination. Dans la figure 6.3, les voies sources sont représentées par des numéros et les voies de destination par des lettres majuscules. À partir d’une voie source donnée, un véhicule peut atteindre une ou plusieurs voies de destination. Ces liens possibles sont appelés les connexions, chaque connexion représentant un lien entre une seule voie source et une seule voie de destination. Chaque voie peut aussi appartenir à plusieurs connexions.

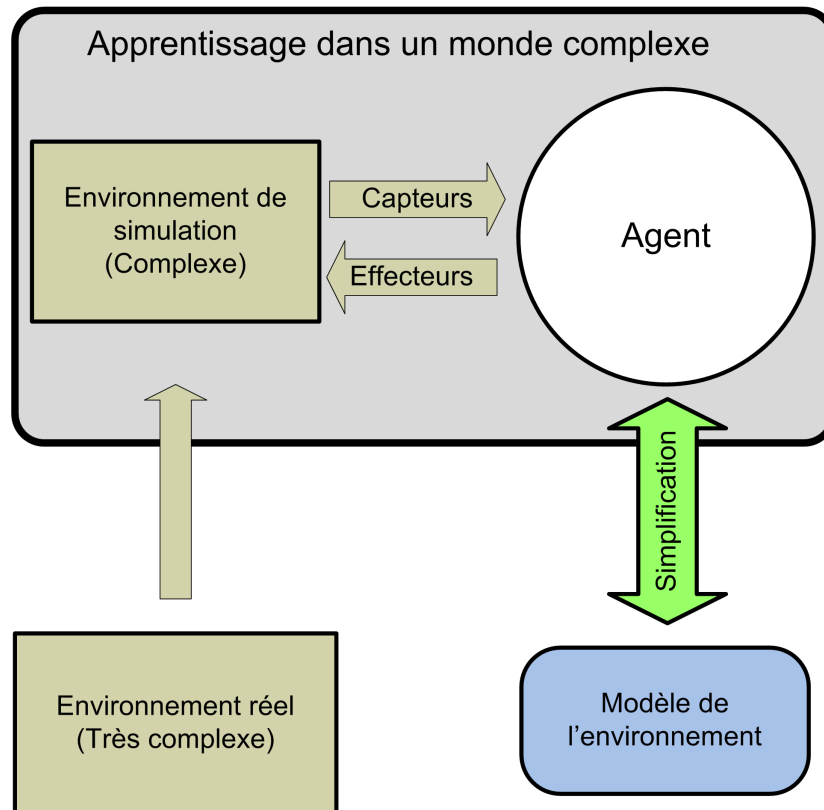


FIG. 6.2 – « Simplification » de l'environnement par l'agent : l'agent apprend dans un environnement simulé « proche » de la réalité.

Lorsqu'une connexion croise physiquement une autre connexion, on dit que ces connexions sont conflictuelles ou ennemies. C'est le cas des connexions « b » et « c » dans la figure 6.3. Si deux voitures empruntent simultanément ces deux connexions, il y a risque de collision. À l'inverse, lorsque deux connexions ne se croisent en aucun point, on dit qu'elles sont non conflictuelles ou amies (connexions « a » et « b » par exemple). Des véhicules peuvent y circuler simultanément sans risque de collision. Le but d'un système de signalisation à une intersection est de gérer ces conflits entre les connexions. Ces systèmes indiquent à quels moments et sous quelles conditions un véhicule peut traverser l'intersection. Par exemple, pour une intersection simple en forme de croix avec des panneaux d'arrêt aux quatre voies, un véhicule peut traverser l'intersection une fois qu'il s'est arrêté complètement et qu'aucun autre véhicule n'est engagé dans une connexion conflictuelle à la sienne. Si deux véhicules veulent emprunter des connexions conflictuelles en même temps, celui qui est arrivé le premier à l'intersection a priorité.

Pour une intersection avec feux de signalisation, la couleur des lumières indique quelles connexions ont le droit de passage et certaines règles du code de conduite spécifient lesquelles ont la priorité. Par exemple, un véhicule tournant à gauche doit

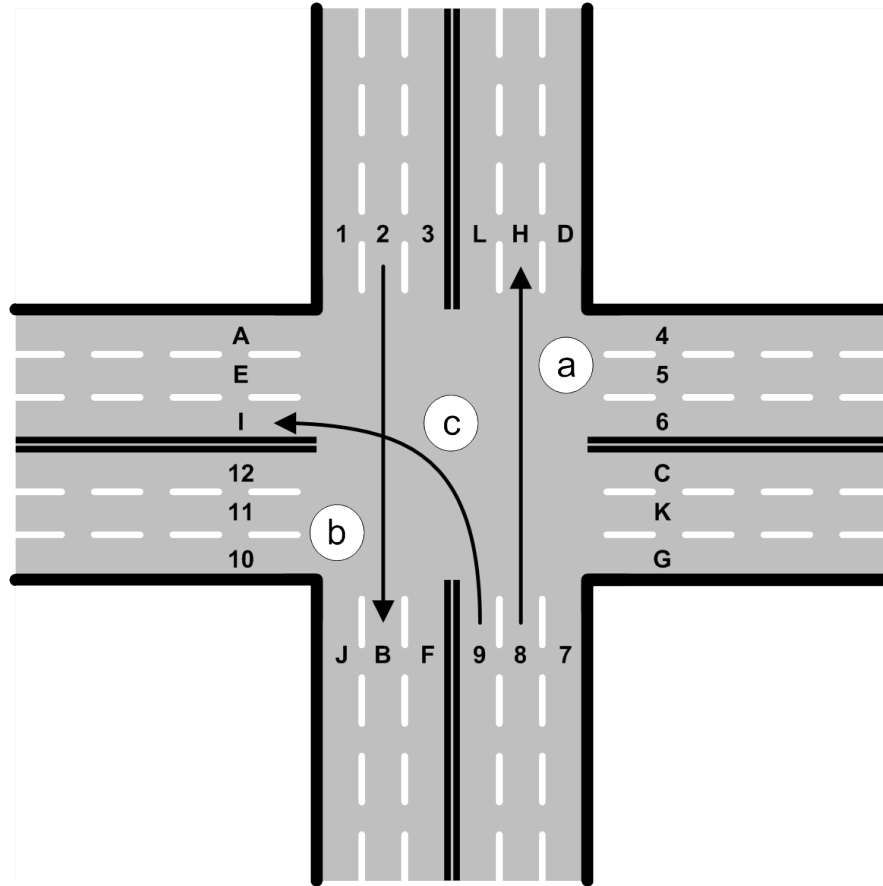
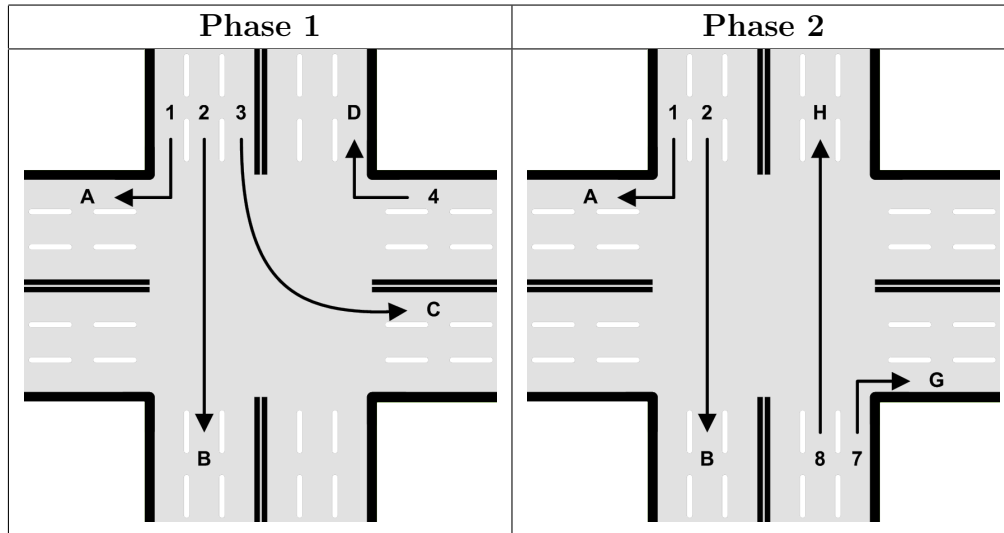


FIG. 6.3 – Intersection avec 12 voies sources et 12 voies de destination.

laisser passer les véhicules venant en direction opposée, à moins d’indication contraire. Bien entendu, une lumière verte autorise la circulation alors qu’une lumière rouge l’interdit. Ainsi, à un instant t , certaines connexions ont le droit de passage et d’autres ne l’ont pas. Cette combinaison de lumière s’appelle une phase. Chaque intersection utilise un ensemble de phases sur lequel elle peut boucler. Le tableau 6.2 montre deux phases possibles. Un cycle est une boucle complète sur toutes les phases de l’ensemble. Chaque connexion doit avoir le droit de passage au moins une fois dans le cycle. Des lumières jaunes sont utilisées entre chaque phase pour les connexions actives qui vont passer à l’état inactif.

À partir de ces définitions, il est facile de modéliser une intersection avec une approche et un langage de programmation orienté objet. Par exemple, une classe représentant une intersection pourrait avoir comme membre un vecteur de routes, un vecteur de voies sources, un vecteur de voies de destination, un ensemble de connexions, etc. La classe représentant une voie aurait un vecteur de connexions entrantes et un vecteur de connexions sortantes. Cela crée une structure semblable à un graphe et permet aux algorithmes d’exploiter facilement ces données.



TAB. 6.2 – Exemple de phases possibles.

6.1.3 Débit d'une intersection en fonction de la demande

Un aspect important de la modélisation est d'être capable de calculer le débit d'une intersection en fonction de la demande. Si le débit est égal à la demande, l'intersection ne congestionne pas. Le débit d'une intersection peut être calculé en faisant la somme des débits de toutes les connexions en fonction de la programmation de la signalisation. Nous avons vu précédemment que le débit maximal d'un itinéraire correspond à celui de son point critique. Cela veut dire que le débit maximal d'une connexion sera équivalent au débit le moins élevé entre celui de la voie source, celui de la voie de destination et celui pour traverser l'intersection.

Ainsi, selon la modélisation présentée à la sous-section 6.1.2, une phase permet de « vider » des voies sources dans des voies de destination. Cependant, même si une connexion a le droit de passage, le nombre de véhicules qui vont transiger sur cette connexion peut être moindre que la demande. Autrement dit, il se peut que des véhicules voulant utiliser cette connexion ne puissent le faire même s'ils en ont le droit, ce qui cause de la congestion. Il y a trois raisons à une telle congestion :

1. *La voie de destination est saturée* : dans ce cas, l'utilisation de la connexion courante est limitée par la vitesse d'écoulement de la connexion sortante.
2. *Les véhicules n'ont pas la priorité* : dans ce cas, le débit de la connexion est limité par l'achalandage de la connexion qui a priorité sur cette connexion (par exemple, des véhicules veulent tourner à gauche mais ne peuvent le faire que lorsqu'il y a assez d'espace entre deux véhicules venant en sens opposé).

3. *La voie source est bloquée par des véhicules ne voulant pas emprunter cette connexion.* Dans ce cas, le débit est limité par la vitesse d'écoulement de la connexion bloquante. Si la connexion bloquante n'a pas le droit de passage, l'utilisation de la connexion courante devient nulle.

Ainsi, le débit d'une connexion est déterminé par la capacité de cette connexion, qui elle est limitée par trois aspects : la capacité de la voie de sortie, les connexions prioritaires la croisant et les autres connexions ayant la même voie source. Un système de signalement doit tenir compte de tous ces facteurs pour être optimal.

6.1.4 Choix d'actions pour un agent

Il a été vu à la section 6.1.1 que l'espace d'états d'une intersection est pratiquement infini et qu'un agent doit se créer une modélisation simplifiée de cet état pour pouvoir obtenir une politique de contrôle. Simplifier cet environnement complexe est le premier défi au problème d'optimisation des feux de signalisation. Le deuxième problème est de concevoir l'ensemble des choix d'actions offert à l'agent. Le but de l'agent contrôlant les feux de signalisation à une intersection est de choisir quelles connexions sont actives et à quel moment. En d'autres mots, il doit choisir son ensemble de phases, choisir dans quel ordre il les affichera et combien de temps durera chacune des phases.

Modéliser tous ces paramètres sous forme d'actions pour un agent peut se faire de plusieurs façons. Par exemple, on pourrait choisir différents cycles possibles et permettre à l'agent d'en sélectionner un toutes les heures. Ceci lui enlèverait une certaine liberté d'action mais faciliterait la tâche d'apprentissage. À l'inverse, on pourrait demander à l'agent de choisir quelles connexions sont actives à toutes les cinq secondes, ce qui lui donnerait une flexibilité et une liberté d'action énorme mais rendrait sa tâche de décision beaucoup plus complexe. En effet, si l'espace d'états du choix d'action est trop grand, l'agent aura de la difficulté à explorer cet espace d'états pour connaître l'effet de toutes les actions possibles. Il faut donc trouver un juste milieu dans la liberté d'action offerte à l'agent.

6.1.5 Fonction de récompense

Le troisième défi est de construire une fonction de récompense pour guider l'agent dans son apprentissage. La conception de cette fonction semble facile à première vue puisqu'en tant qu'automobilistes, on pourrait opter pour :

- Attendre le moins longtemps à une lumière ;
- S’arrêter le moins souvent : on préfère un itinéraire où l’on doit attendre 2 minutes à une lumière plutôt qu’un itinéraire de même longueur avec 20 panneaux d’arrêts ;
- Ne pas attendre deux fois à la même lumière : si plusieurs véhicules nous précèdent, nous nous attendons à ce que la lumière verte soit assez longue pour que tous les véhicules traversent l’intersection sans attendre une deuxième fois ;
- Ne pas être bloqué derrière quelqu’un à une intersection lorsque nous avons le droit de passage : par exemple, nous voulons tourner à droite, la flèche verte nous en donne le droit mais le véhicule qui nous précède veut continuer tout droit et n’a pas le droit de passage ;
- Atteindre notre destination le plus rapidement possible.

Ainsi, avec ces désirs en tête, on peut facilement juger de l’efficacité des feux de signalisation lorsque l’on se promène sur le réseau routier. Je crois qu’il n’est pas exagéré de dire que nous avons tous été frustrés à un moment ou à un autre par le fonctionnement de la signalisation à une ou plusieurs intersections. Cependant, même s’il est facile de juger de l’efficacité d’une intersection en tant qu’individu, il est beaucoup plus difficile de rassembler tous ces facteurs et de les mettre en relation pour créer une fonction mathématique de récompense pour un agent contrôlant l’intersection.

De plus, la satisfaction des usagers est basée sur l’ensemble des intersections et non sur chacune d’elle : si l’on attend à une lumière, on espère ne pas avoir à arrêter aux deux prochaines intersections si elles sont suffisamment rapprochées. Ainsi, les récompenses données à un agent d’une intersection ne doivent pas uniquement refléter son environnement local mais aussi sa performance dans le réseau global.

À force de travailler le problème et de se pencher sur la question de la fonction de récompense, on en vient à résumer ces facteurs en quelques paramètres. Les principaux paramètres qui peuvent être pris en compte par la fonction de récompense sont les suivants :

1. Le délai de parcours pour chaque automobiliste ;
2. Le débit du trafic à l’intersection ;
3. Le nombre d’arrêts effectués par chaque véhicule.

En résumé, peu importe les situations que nous rencontrons sur notre itinéraire, ce qu’il nous faut c’est de s’arrêter le moins longtemps et le moins souvent possible. De plus, il faut qu’un maximum de véhicules puisse transiger par chaque intersection. Les approches présentées dans les prochaines sections utilisent des fonctions de récompense basées sur ces trois facteurs.

6.2 Apprentissage par renforcement pour la gestion de carrefours

La première approche qu'on a testée était basée sur l'apprentissage par renforcement. L'intuition derrière cette approche est de faire évoluer l'agent dans une simulation de trafic où chaque véhicule lui donne une récompense en fonction de son temps d'attente. Rappelons qu'un agent se définit comme une entité capable d'agir de façon autonome, de percevoir son environnement, d'être persistant pendant de longues périodes et de s'adapter aux changements (voir section 2.2). L'agent connaît en tout temps le nombre de véhicules qui attendent à l'intersection pour toutes les directions. Au niveau de la communication, cela implique qu'un véhicule envoie un message à l'agent lorsqu'il arrête à une intersection et envoie un autre message lorsqu'il traverse celle-ci afin de lui communiquer son temps d'attente total. Ces travaux ont fait l'objet d'une publication dans une conférence internationale sur les systèmes de transport intelligents [Grégoire *et al.*, 2007]. L'article est disponible à l'annexe A.

6.2.1 Simulation pour l'apprentissage

Pour implémenter l'agent apprenant, nous avons utilisé le même simulateur que celui présenté à la section 4.4. Il s'agit d'un simulateur continu dans l'espace et discret dans le temps. Le scénario consiste en une longue route verticale croisée par quatre autres routes. Il y a donc quatre intersections sur la route verticale et les trois tronçons entre les intersections mesurent 280 mètres chacun. Les routes horizontales ont une voie pour chaque direction alors que la route verticale a quatre voies au total, deux dans chaque direction. Des véhicules sont générés à chaque extrémité selon une distribution uniforme. Pour simplifier la simulation, les véhicules vont toujours en ligne droite et ne tournent pas sur les routes qu'ils croisent. Tous les véhicules ont une accélération de $2m/s^2$ et une capacité de freinage de $4m/s^2$. Chaque véhicule est équipé d'un contrôleur de base qui garde une distance sécuritaire avec le véhicule qui le précède et réagit convenablement aux feux de signalisation. S'il n'y a aucun véhicule à l'avant, le contrôleur ajuste la vitesse de la voiture à $14m/s$, soit $50km/h$.

Chaque intersection est équipée d'un feu de signalisation contrôlé par un agent. Il y a seulement deux phases de lumières possibles : une pour laisser passer les véhicules circulants à la verticale et une autre pour ceux à l'horizontale. La transition entre les deux phases se fait par une lumière jaune de 10 pas de temps, soit 2,5 secondes. Cela donne assez de temps aux véhicules qui s'approchent du carrefour pour arrêter et aux

véhicules déjà engagés pour le traverser en toute sécurité.

6.2.2 Modélisation sous forme de MDP

Pour faire de l'apprentissage par renforcement, nous avons modélisé le problème sous forme de MDP. Voici une description des choix d'actions, de l'espace d'états et de la fonction de récompense utilisée. Pour la fonction de transition, elle était évidemment gérée par le simulateur et inconnue de l'agent (et de nous).

Choix d'actions

À chaque étape de décision, l'agent choisi s'il change de phase ou non. S'il décide de changer de phase, les lumières passent obligatoirement au jaune puis à l'autre phase. Rappelons qu'il n'y a que deux phases possibles, soit à la verticale soit à l'horizontale. Ainsi, l'ensemble d'actions est limité à deux choix : changer ou ne pas changer de phase. Comme il n'y a que deux phases, cela revient au même que de lui faire choisir directement la phase.

Les étapes de décisions surviennent à tous les 50 pas de temps. Comme la phase de transition des lumières jaune dure 10 pas de temps, la durée minimale d'une phase normale est de 40 pas de temps, soit 10 secondes. Cependant, si l'agent décide de ne pas changer de phase, la lumière peut rester verte plus longtemps que le 10 secondes minimales. À chaque étape de décision sans changement, la lumière reste au vert 12,5 secondes de plus.

Espace d'état

L'état est représenté par quatre variables :

1. Le nombre de véhicules qui attendent à l'horizontale ;
2. Le nombre de véhicules qui attendent à la verticale ;
3. La phase courante (verte à l'horizontale ou à la verticale) ;
4. Le nombre d'étapes de décision depuis le dernier changement de phase.

On se retrouve donc avec une variable binaire et trois variables discrètes. Le nombre de voitures attendant dans chaque direction peut-être assez élevé tandis que le nombre d'étapes de décision ne peut pratiquement pas dépasser une dizaine d'étapes. En effet, après 10 étapes de décision sans changement de phase, la lumière aura été au vert pendant 135 secondes, soit 2 minutes et 15 secondes. Nous forçons donc l'agent à changer de phase s'il ne l'a pas déjà fait après 10 étapes de décision.

Comme le nombre de voitures attendant dans chaque direction peut-être assez élevé, l'espace d'états devient rapidement énorme. En effet, si l'on suppose un nombre maximum de 30 voitures pour chaque direction, on se retrouve avec un ensemble d'états de 180 000 éléments ($2 \times 10 \times 30 \times 30$). Pour travailler avec un aussi gros ensemble d'états, nous avons utilisé un algorithme de gradient de politique plutôt que le Q-Learning.

Fonction de récompense

Durant leur parcours, les véhicules sont ralentis s'ils rencontrent une lumière rouge. Dès qu'ils commencent à freiner à cause d'une lumière, ils commencent à compter et à accumuler leur délai d'attente. À chaque pas de temps, un véhicule met à jour son délai selon sa vitesse courante et sa vitesse désirée. Cela se traduit par les équations suivantes :

$$D_{t+1}(v) = D_t(v) + d(v) \quad (6.1)$$

où

$$d(v) = 0.250 * (1 - (S_{ct}/S_{dt})) \quad (6.2)$$

$D_t(v)$ est le délai cumulé au temps t de la voiture v et $d(v)$ est le délai du pas de temps courant de la voiture v . Comme notre pas de temps de simulation est fixé à 250 millisecondes, cette équation donne le délai du véhicule en secondes. En ajustant le délai de chaque pas de temps par la vitesse courante (S_{ct}) et la vitesse désirée (S_{dt}) du véhicule, nous prenons en considération l'accélération et le freinage de la voiture.

La fonction de récompense utilise ces délais pour donner des récompenses négatives à l'agent. La valeur de la récompense est calculée selon la fonction suivante :

$$R(s_t, a_t) = - \sum_{v \in V} ((D_t(v)/50)^2) * 1.5 \quad (6.3)$$

où $D_t(v)$ est le délai en secondes de la voiture v au temps t (calculé à l'équation 6.1) et où V est l'ensemble des véhicules à cette intersection. La constante (50) est utilisée pour représenter un délai acceptable d'attente à une lumière par un véhicule alors que la fonction au carré et la constante 1.5 diminuent la récompense donnée à mesure qu'un véhicule dépasse le temps d'attente acceptable.

Nous donnons également une récompense négative à chaque changement de phase dans le but d'éviter que l'agent change à chaque étape de décision.

6.2.3 Résultats d'apprentissage

Premièrement, nous avons entraîné l'agent apprenant sur un scénario ne comportant qu'une seule intersection. La probabilité de générer un véhicule est de 14% dans la direction nord/sud et de 3% dans la direction est/ouest. La figure 6.4 représente la courbe d'apprentissage de l'agent après une seule exécution de l'algorithme. Elle représente la diminution du délai des véhicules en fonction du nombre d'épisodes d'apprentissage.

Nous avons ensuite comparé la politique apprise à un contrôleur fixe. Le premier scénario utilise la même distribution de probabilité pour la génération des véhicules que dans le scénario d'apprentissage, soit 14% à la verticale et 3% à l'horizontale. Cela donne respectivement un flux de véhicule de 1008 et de 432 véhicules par heure par voie. N'oublions pas que les voies sont doubles dans la direction nord/sud. Comme la capacité de l'intersection est de 1400 véhicules par heure par voie, les paramètres de simulation sont justes en dessous de la capacité maximale.

Les contrôleurs fixes sont contraints à la même durée de phase que le contrôleur apprenant. Ainsi, la longueur de leur cycle doit être un multiple de 50 pas de temps. Le meilleur choix en fonction de la génération de véhicule est un cycle de 150 pas de temps avec une phase nord/sud de 100 pas de temps. Cela donne 66% de lumière verte à la verticale et 33% à l'horizontale.

Le premier test a été donc de comparer les contrôleurs dans le même scénario que ceux utilisant l'apprentissage, soit avec une seule intersection. Nous avons calculé le délai moyen par véhicule pour chacun des contrôleurs sur 50 épisodes de simulation

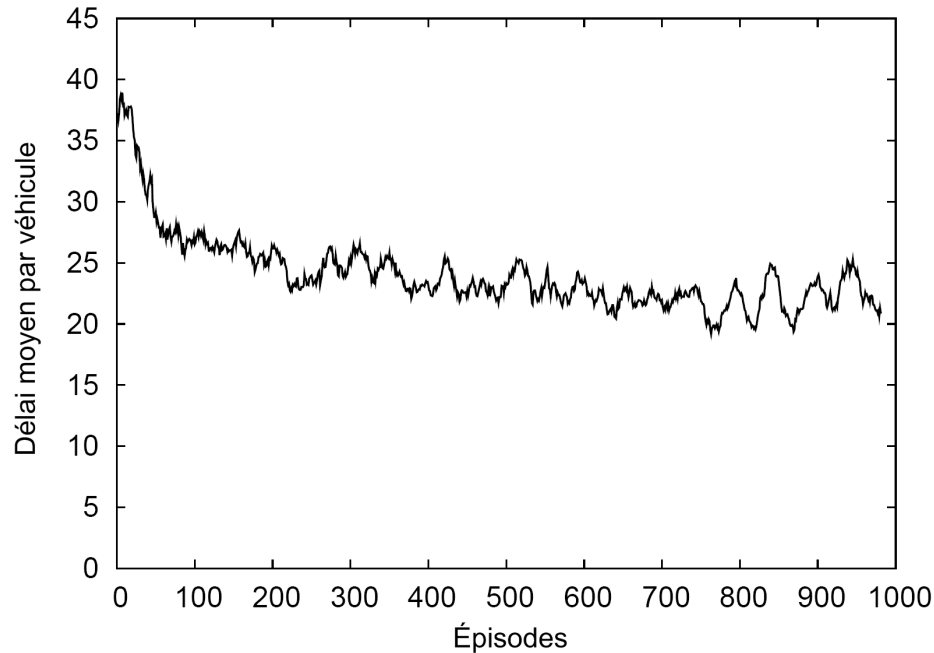


FIG. 6.4 – Apprentissage utilisant l’algorithme de gradient de politique [Grégoire *et al.*, 2007].

pour obtenir des résultats représentatifs. Nous avons aussi comparé les contrôleurs à un contrôleur fixe sous-optimal qui alloue 50% de lumière verte dans chacune des directions. Le tableau 6.3 donne le délai moyen pour les trois contrôleurs.

| Systeme | Délai moyen | Écart type |
|-----------------------------------|-------------|------------|
| À contrôleur fixe presque optimal | 22.10 | 3.86 |
| À contrôleur fixe sous-optimal | 61.59 | 5.33 |
| Apprenant une politique | 19.96 | 4.54 |

TAB. 6.3 – Moyenne et écart type du délai des véhicules pour un scénario à une intersection (Calculés sur 50 simulations).

Comme nous pouvons le voir, l’agent apprenant a appris une politique presque optimale. La petite différence de performance entre le contrôleur presque optimal et la politique apprise peut s’expliquer par le choix des phases du contrôleur fixe qui ne reflète pas exactement la génération du trafic. La génération de trafic est de 70% à la verticale et 30% à l’horizontale plutôt que 66% et 33%.

Nous avons ensuite testé les contrôleurs dans le scénario des quatre intersections. Nous avons utilisé les mêmes paramètres de génération de véhicules à l’exception que

avons diminué de moitié le nombre de véhicules circulant du sud vers le nord. Il y a donc 1008 véhicules à l'heure par voie voulant aller vers le sud et 504 voulant aller vers le nord. Nous avons aussi synchronisé les contrôleurs fixes pour avantager la circulation dans la direction de la plus forte demande, c'est-à-dire en créant des « vagues vertes » vers le sud.

| Systeme | Délai moyen | Écart type |
|-----------------------------------|-------------|------------|
| À contrôleur fixe presque optimal | 34.86 | 4.82 |
| Apprenant une politique | 38.45 | 4.74 |

TAB. 6.4 – Moyenne et écart type du délai des véhicules pour un scénario à quatre intersections (Calculés sur 50 simulations).

Le tableau 6.4 montre que la politique apprise performe quasiment aussi bien que les contrôleurs fixes, ce qui est très bon, étant donné le fait que les contrôleurs fixes sont synchronisés dans le sens du trafic.

6.2.4 Conclusion partielle

Les résultats obtenus montrent que l'apprentissage par renforcement est une solution potentiellement viable pour améliorer le contrôle des feux de signalisation. C'est une approche novatrice qui s'agence relativement bien au problème de gestion de feux et offre de bons résultats. Cependant, les algorithmes d'apprentissages sont gourmands en temps de calcul et il reste beaucoup de travail à faire pour optimiser la modélisation de l'environnement afin de diminuer un tel temps d'apprentissage.

6.3 Approche dynamique

L'approche présentée dans cette section est de concevoir un algorithme qui tire profit de la connaissance du problème d'optimisation des feux plutôt que d'appliquer des algorithmes d'apprentissage génériques. En se basant sur les méthodes d'optimisation actuelles des feux de signalisation, on conçoit un agent dynamique pour le contrôle d'une intersection.

6.3.1 Inconvénients de l'apprentissage

Comme il a été vu à la section précédente, l'apprentissage par renforcement est une approche novatrice qui permet d'obtenir de bons résultats pour l'optimisation des feux de signalisation. Cependant, elle comporte aussi quelques inconvénients. Premièrement, les algorithmes d'apprentissage sont très gourmands en ressources au niveau de l'exécution. L'agent doit apprendre en interagissant avec le monde. Pour se faire, il doit essayer à répétition toutes les actions possibles dans tous les états possibles de l'environnement pour connaître et apprendre leurs effets. Cela demande beaucoup de temps et de mémoire.

Pour faciliter ce processus, on tente généralement de simplifier le problème dans la modélisation de l'environnement. On se retrouve alors avec un agent qui apprend et performe dans un monde simplifié et qui ne prend pas nécessairement en compte toutes les facettes du problème.

De plus, une politique apprise par un agent à une intersection n'offrira pas nécessairement de bons résultats si elle est utilisée à une autre intersection. Pour qu'un agent apprenant soit performant à plusieurs intersections, il faut qu'il soit sans cesse en processus d'apprentissage et qu'il s'adapte de façon dynamique à toutes les situations, ce qui est loin d'être facile à mettre en oeuvre.

6.3.2 Intuition et description de l'approche

Les méthodes conventionnelles de conception de systèmes de signalisation sont généralement composées de trois étapes :

1. Recueil de statistiques sur la circulation à l'intersection ;
2. Calcul de la programmation optimale des lumières pour différentes heures de la journée ;
3. Implantation de la programmation dans les feux de l'intersection.

Ces méthodes donnent d'excellents résultats lorsque le trafic correspond aux données recueillies. Cependant, comme nous l'avons précédemment signalé, ces systèmes de signalisation fixes ne s'adaptent pas aux changements.

L'intuition de l'approche présentée dans cette section est de se baser sur cette méthodologie tout en la rendant plus adaptable. En utilisant les informations recueillies en temps réel par la communication route-à-véhicule, l'agent placé à l'intersection calcule la programmation optimale correspondant au trafic actuel. Le but est d'utiliser un algorithme qui s'exécute rapidement pour permettre à l'agent de calculer à nouveau la politique optimale aussi souvent que nécessaire. De cette façon, il est possible de simuler un réseau routier comptant plusieurs dizaines d'intersections. De plus, la simulation d'une heure de trafic ne nécessite que quelques minutes, voir quelques secondes.

En particulier, l'algorithme doit :

1. Être assez générique pour pouvoir calculer la programmation optimale de n'importe quelle intersection, peu importe le nombre de directions et de voies ;
2. Tenir compte de tous les aspects du problème en le simplifiant le moins possible ;
3. Offrir de bonnes performances en terme de temps d'exécution.

Une très bonne solution pour respecter ces critères est de concevoir un algorithme qui est directement lié au problème de l'optimisation des feux, c'est-à-dire de tirer avantage des caractéristiques du problème pour bâtir un algorithme spécialisé en la matière. Cela va à l'inverse des techniques utilisant l'apprentissage automatique. Celles-ci se basent en effet sur des algorithmes génériques qui peuvent être appliqués à n'importe quel problème et qui sont même spécifiquement conçus pour évoluer dans un monde inconnu. Le fait qu'ils soient aussi génériques entraîne comme conséquence un temps d'exécution élevé durant la phase d'apprentissage.

Cette approche est donc basée sur le problème plutôt que sur la théorie : en analysant plus en profondeur le problème de l'optimisation des feux, on peut arriver à une solution très performante et à une politique plus intuitive que celle apprise par un algorithme d'apprentissage.

6.3.3 Décortiquation du problème et fonctionnement de l'agent

Une bonne façon d'analyser le problème qu'on vise à mettre en oeuvre est de le décortiquer à l'aide d'un exemple. Prenons le plan illustré à la figure 6.5. Nous avons six intersections et la direction générale du trafic est vers le centre-ville. Nous pouvons supposer que cette situation correspond à l'heure de pointe du matin où les usagers se dirigent vers leur lieu de travail.

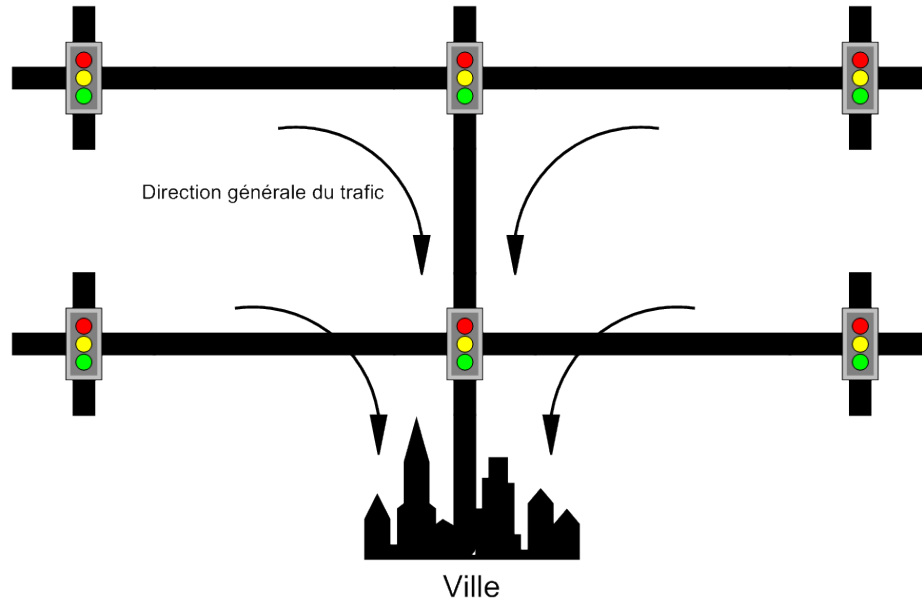


FIG. 6.5 – Réseau routier de six intersections. La direction du trafic est vers le centre-ville.

La figure 6.6 représente la demande en nombre de véhicules par heure pour chacune des routes et des intersections. L'intersection en bas au centre de la figure correspond à celle qui a la plus grosse demande de véhicules par heure.

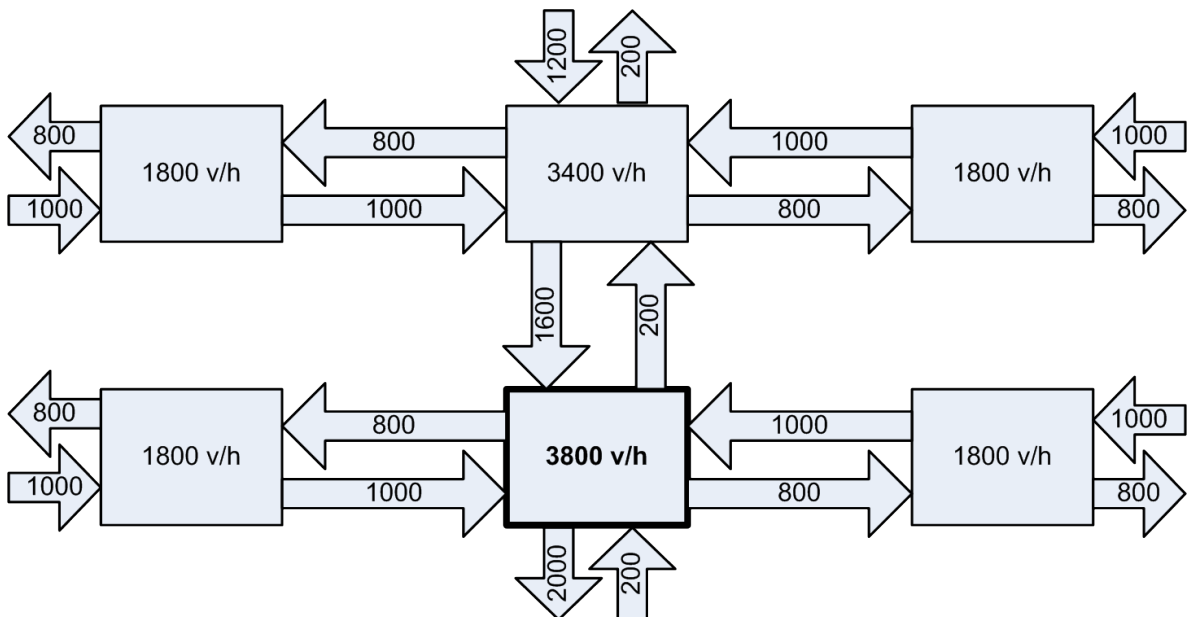


FIG. 6.6 – Demande en véhicules par heure.

Analysons de plus près cette intersection. Admettons que nous avons les 12

connexions présentées à la figure 6.7 et la répartition de la demande par connexion présentée dans le tableau 6.5. Il est important de noter que la répartition de la demande par connexion doit respecter la demande établie dans l'exemple pour chacune des routes. Par exemple, si l'on additionne la demande pour les connexions 1, 2 et 3, nous obtiendrons 1600 véhicules/heure. Il en va de même pour les connexions sortantes. Si l'on additionne la demande pour les lettres J, B et F, nous obtenons 2000 véhicules/heure.

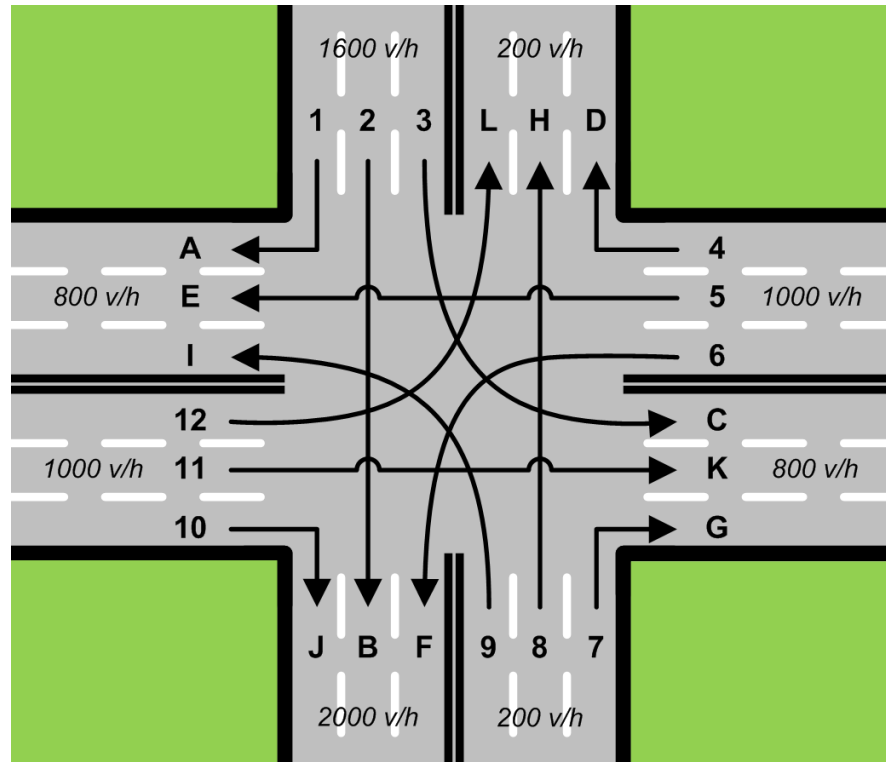


FIG. 6.7 – Les 12 connexions de l'intersection.

| | | | | | | | |
|------------|------|------------|-----|------------|-----|-------------|-----|
| 1-A | 200 | 4-D | 50 | 7-G | 50 | 10-J | 400 |
| 2-B | 1200 | 5-E | 550 | 8-H | 100 | 11-K | 550 |
| 3-C | 200 | 6-F | 400 | 9-I | 50 | 12-L | 50 |

TAB. 6.5 – Répartition de la demande pour chaque connexion. La demande est exprimée en véhicules par heure.

Lorsque l'agent est à une intersection, il reçoit l'ensemble des données physiques de celle-ci (ensembles des routes formant l'intersection, ensemble des voies sources et des voies de destination, ensemble des connexions possibles). À partir de ses données, l'agent

| Phase | Connexions | Demande | Phase | Connexions | Demande |
|-------|------------|---------|-------|------------|---------|
| P1 | 1-2-3-4 | 1650 | P10 | 1-7-10-12 | 700 |
| P2 | 1-2-4-7 | 1500 | P11 | 1-10-11-12 | 1200 |
| P3 | 1-2-7-8 | 1550 | P12 | 3-4-9-10 | 700 |
| P4 | 1-3-4-10 | 850 | P13 | 4-5-6-7 | 1050 |
| P5 | 1-4-6-7 | 700 | P14 | 4-5-7-10 | 1050 |
| P6 | 1-4-7-10 | 700 | P15 | 4-5-10-11 | 1550 |
| P7 | 1-4-10-11 | 1200 | P16 | 4-7-9-10 | 550 |
| P8 | 1-6-7-12 | 700 | P17 | 7-8-9-10 | 600 |
| P9 | 1-7-8-10 | 750 | | | |

TAB. 6.6 – Ensemble de phases possibles.

calcule les connexions conflictuelles (i.e. celles qui se croisent) et calcule l'ensemble des phases possibles.

Pour l'intersection de la figure 6.7, nous obtenons les 17 phases du tableau 6.6. Pour chaque phase, nous indiquons la liste des connexions qui ont le droit de passage ainsi que la demande correspondante à cette phase en véhicules par heure.

L'agent construit ensuite l'ensemble des cycles possibles, un cycle étant un ensemble de phases où chaque connexion se retrouve au moins une fois dans une phase du cycle. Ces deux étapes (la construction des phases et des cycles) font parti de l'initialisation. Une fois initialisé, l'agent est immédiatement opérationnel, c'est-à-dire qu'il est prêt à recevoir des données sur le trafic et à choisir les cycles optimaux.

La réception des données sur le trafic se fait grâce à la communication route-à-véhicule. Les véhicules envoient un seul message à chaque fois qu'ils entrent sur un segment et ils spécifient quelle est la prochaine route qu'ils emprunteront. L'agent sait alors quelle connexion sera utilisée par ce véhicule et met à jour la demande de cette connexion.

Pour choisir le cycle optimal, l'agent mémorise et calcule des données et des statistiques sur chaque voie et chaque connexion de l'intersection. Connaissant la demande et le débit maximum de chaque connexion, il peut calculer le pourcentage du cycle pour lequel chaque connexion doit être verte pour satisfaire la demande. Autrement dit, si le débit maximal d'une connexion est de 2000 véhicules à l'heure et que la demande pour cette connexion est de 1000 véhicules à l'heure, la connexion devra être verte au moins 50% du temps pour satisfaire la demande. Le tableau 6.7 donne ces chiffres pour l'exemple courant. Le débit maximal choisi est de 2200 véhicules à l'heure et les pourcentages ont été arrondis à l'entier supérieur.

| | | | | | | | |
|------------|-----|------------|-----|------------|----|-------------|-----|
| 1-A | 10% | 4-D | 3% | 7-G | 3% | 10-J | 19% |
| 2-B | 55% | 5-E | 26% | 8-H | 5% | 11-K | 26% |
| 3-C | 10% | 6-F | 19% | 9-I | 3% | 12-L | 3% |

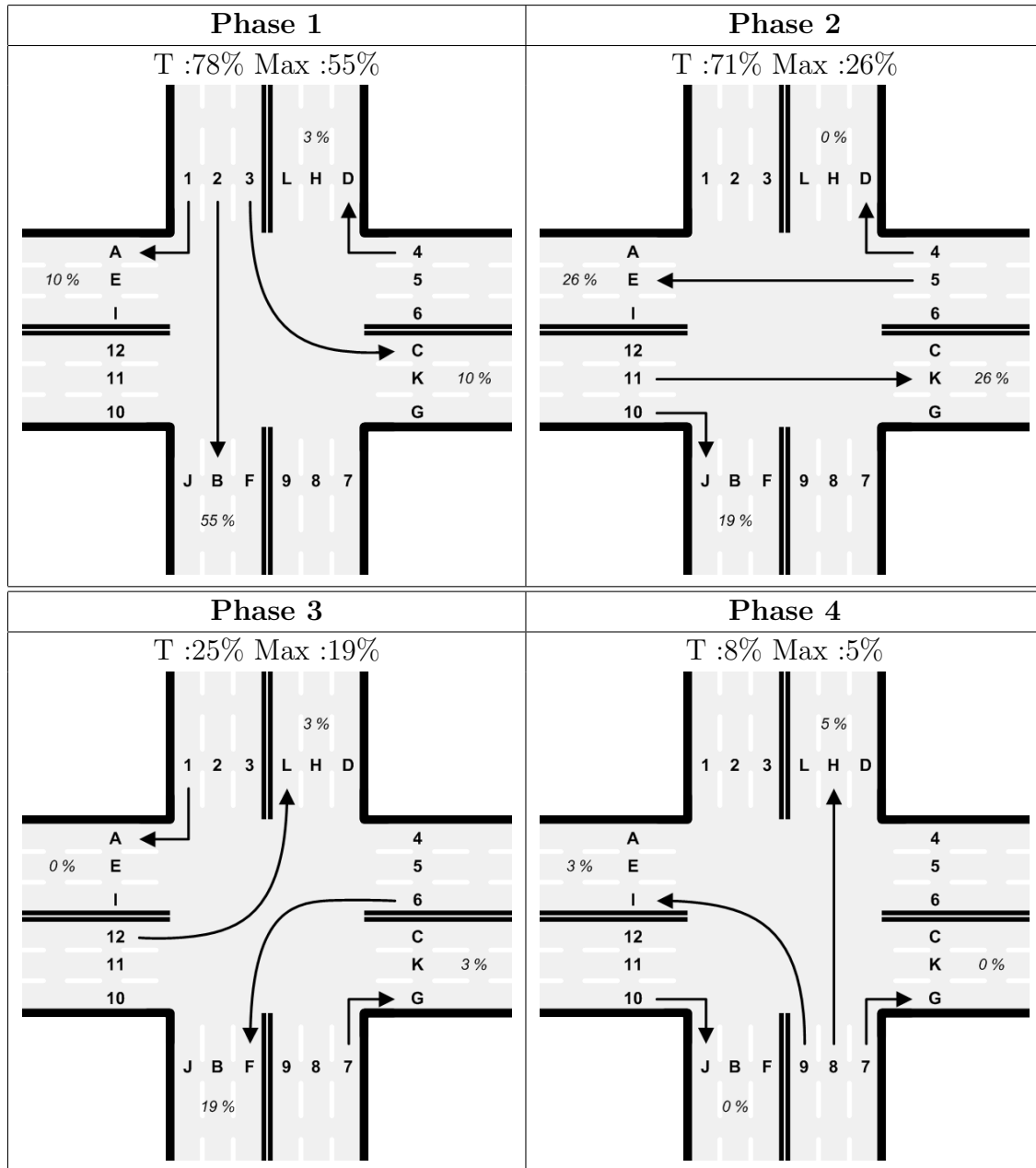
TAB. 6.7 – Pourcentage minimal du temps pour lequel chaque connexion doit avoir le droit de passage de façon à satisfaire la demande.

Pour chaque phase, l'agent calcule un chiffre représentant son utilité à satisfaire la demande. Ce chiffre correspond à la somme des pourcentages des connexions de la phase. Il classe ensuite les connexions et les phases en ordre décroissant de pourcentage et d'utilité. Pour chacune des connexions, en commençant par celle ayant le plus grand pourcentage, l'agent sélectionne la première phase (ayant la plus grande utilité) contenant cette connexion et l'ajoute au cycle optimal. Il enlève ensuite toutes les connexions de cette phase de la liste des connexions à ajouter et met à jour l'utilité des phases restantes. Le tableau 6.8 donne le cycle résultant de cet algorithme pour l'exemple courant. On peut observer que pour la deuxième phase, la connexion 4 permettant de tourner à droite est à 0% puisque cette connexion a déjà été active dans la première phase.

La dernière étape est de répartir le temps du cycle entre les phases. Pour ce faire, l'agent doit calculer le pourcentage total du cycle. Pour chaque phase, il prend la connexion ayant le plus haut pourcentage et l'ajoute au pourcentage total du cycle. Il y a ensuite deux scénarios possibles : soit que le pourcentage total est en dessous de 100%, soit qu'il est au-dessus. Dans le premier cas, la division du temps du cycle est simple : il accorde à chaque phase le pourcentage de la connexion avec le plus grand pourcentage. Le pourcentage du cycle restant est divisé équitablement entre les phases. Dans le deuxième cas, il faut réduire le pourcentage total du cycle à 100%. C'est le cas de l'exemple courant où le pourcentage total égalise 105%. Pour ce faire, l'agent normalise les pourcentages alloués à chaque phase de façon à ce que le pourcentage total égalise 100%.

Comme les véhicules envoient leurs informations de parcourt de façon régulière, l'agent de chaque intersection possède toujours les données de trafic les plus récentes. Il peut donc calculer le cycle optimal de façon régulière (par exemple à chaque 2 minutes) et s'adapter très rapidement au changement.

En résumé, l'algorithme d'adaptation à l'allure de l'algorithme 2.



TAB. 6.8 – Cycle optimal calculé par l'agent.

Algorithme 2 : fonction `ExecutionAgent(connexions, voies)`.

Précondition : `connexions` est l'ensemble des connexions de l'intersection.

Précondition : `voies` est l'ensemble des voies de l'intersection.

Précondition : `Connexion` est une classe contenant : `voieSrc`, `voieDest`, `debitCourant`, `pourcentage`.

Précondition : `Phase` est une classe contenant : `connexions`, `utilité`.

Précondition : `Cycle` est une classe contenant : `phases`, `pourcentageTotal`.

Précondition : `DEBITMAX` est le débit maximum en véhicule/heure/voie.

Précondition : `DUREECYCLE` est la durée d'un cycle en secondes.

```

1: phases = PhasesPossibles(connexions, voies)
2: tant que agentEnFonction faire
3:   donneesTrafic = LectureCanalCommunication()
4:   MAJDebitCourant(connexions, donneesTrafic)
5:   pour tous connexion ∈ connexions faire
6:     connexion.pourcentage ← (connexion.debitCourant / DEBITMAX)
7:   fin pour
8:   pour tous phase ∈ phases faire
9:     phase.utilité ← SommePourcentage(phase.connexions)
10:  fin pour
11:  TriPourcentageDecroissant(connexions)
12:  TriUtiliteDecroissant(phases)
13:  connexionsTemp ← connexions
14:  phasesTemp ← phases
15:  cycleOptimal.Effacer()
16:  tant que connexionsTemp n'est pas vide faire
17:    connexionMax ← connexionsTemp[0]
18:    phaseMax = phasesTemp.TrouverPremiereContenant(connexion)
19:    cycleOptimal.Ajoute(phaseMax)
20:    pour tous connexion ∈ phaseMax.connexions faire
21:      connexionsTemp.Enleve(connexion)
22:      pour tous phase ∈ phasesTemp faire
23:        phase.connexions.Enleve(connexion)
24:        phase.utilité ← SommePourcentage(phase.connexions)
25:      fin pour
26:    fin pour
27:  fin tant que
28:  pour tous phase ∈ cycleOptimal.phases faire
29:    cycleOptimal.pourcentageTotal += phase.PourcentageMax()
30:  fin pour
31:  cycleOptimal.RepartirTemp(DUREECYCLE)
32:  Executer(cycleOptimal)
33: fin tant que

```

6.3.4 Synchronisation des lumières et coordination entre les agents

Il convient de préciser que l'efficacité du réseau routier ne repose pas uniquement sur l'efficacité de chaque agent mais aussi sur leur collaboration. Les deux règles de coordination proposées sont les suivantes :

1. Il ne faut jamais envoyer plus de véhicules sur une route que ce que la prochaine intersection de cette route peut prendre.
2. Il faut toujours être synchronisé avec la lumière ayant la plus forte demande du réseau.

En respectant la première règle, les routes qui sont à l'intérieur du réseau ne sont jamais congestionnées. Les seules routes qui peuvent être congestionnées sont celles qui représentent des sources pour le réseau. Le but de cette approche est de rendre la circulation fluide en ville et à encourager le transport en commun pour ceux qui veulent entrer en ville. Autrement dit, si les intersections en périphérie de la ville contrôlent le nombre de véhicules qui y entre pour ne pas saturer le réseau interne, il se formera de la congestion à ces endroits clés du réseau routier, par exemple aux entrées des ponts. Il devient alors très facile d'y construire des stationnements et des terminaux d'autobus ou de train pour encourager le transport en commun.

Cette première règle peut facilement être respectée par nos agents. Une fois que chacun a calculé son cycle optimal, la demande et le débit de chaque route sont misent à jour. L'agent peut ensuite vérifier le débit sortant qu'il génère et diminuer celui-ci s'il excède le débit entrant maximal de l'autre intersection.

La deuxième règle permet de synchroniser les feux du réseau tout en laissant le plus de flexibilité aux intersections les plus en demande. L'agent contrôlant l'intersection la plus chargée du réseau devient donc le « chef d'orchestre » et donne le rythme aux autres intersections voisines. Il choisit la direction de la synchronisation en fonction de la plus forte demande (par exemple du nord au sud) et communique avec les autres agents pour leur indiquer le délai à respecter. Ceux-ci communiquent à leur tour aux autres intersections et le mouvement de synchronisation se propage à travers le réseau. Le délai à respecter entre chaque intersection se calcule facilement selon la longueur du segment les séparant et la vitesse des véhicules y circulant.

6.3.5 Simulateur

Étant donné que notre simulateur développé pour concevoir le régulateur de vitesse adaptatif et collaboratif (voir chapitre 4) est axé davantage sur la simulation de quelques véhicules que sur la simulation d'un réseau routier, il était préférable d'en choisir un autre pour simuler la gestion du trafic. Pour s'assurer que la solution proposée est réaliste, il faut faire les tests dans un simulateur s'approchant le plus possible de la réalité pour le problème donné. Dans le cas de cette maîtrise, le simulateur adopté est SUMO [Krajzewicz *et al.* \[2006\]](#), un simulateur de trafic à code ouvert (open source).

SUMO est un simulateur microscopique continu dans l'espace de trafic véhiculaire. Cela signifie que chaque véhicule a sa propre route et est simulé individuellement. Pour simuler les mouvements du véhicule dans le réseau, un modèle développé par [Krauß \[1998\]](#) est utilisé. Ce modèle est rapide tout en offrant des résultats assez précis [Brockfeld *et al.* \[2004\]](#). La discrétisation dans le temps se fait par des pas de temps d'une seconde. De plus, SUMO est également très rapide en temps d'exécution : environ 100 000-200 000 véhicules peuvent être simulés en temps réel sur un ordinateur personnel, incluant la gestion des feux de signalisation, les priorités de passage et les changements de voie. La figure 6.8 montre l'interface graphique de SUMO. On y voit le réseau routier de la ville de Québec ainsi que deux vues approchées démontrant la précision du simulateur au niveau de la gestion des routes et des voies. Le code de SUMO est en C++ est regroupe une série de bibliothèques et d'exécutables. Le simulateur peut fonctionner en mode console ou en mode graphique.

6.3.6 Résultats

Pour tester l'approche présentée dans cette section, une bibliothèque de contrôle des feux a été développée et intégrée à l'architecture du simulateur SUMO. Cette bibliothèque est complètement indépendante du simulateur et pourrait être réutilisée facilement dans tout autre système. La situation testée se base sur l'intersection présentée via l'exemple de la section 6.3.3.

Le premier test consiste à calculer le débit maximal pour chacune des connexions en utilisant SUMO. Le but est de savoir si le débit maximal est influencé par le type de connexion (virage à droite, virage à gauche, en ligne droite). Comme l'intersection est symétrique, le test a été fait uniquement pour les véhicules provenant du nord. Pour se faire, un grand nombre de véhicules ont été générés ayant comme itinéraires les connexions 1, 2 et 3. L'intersection utilisait un cycle fixe de quatre phases allouant

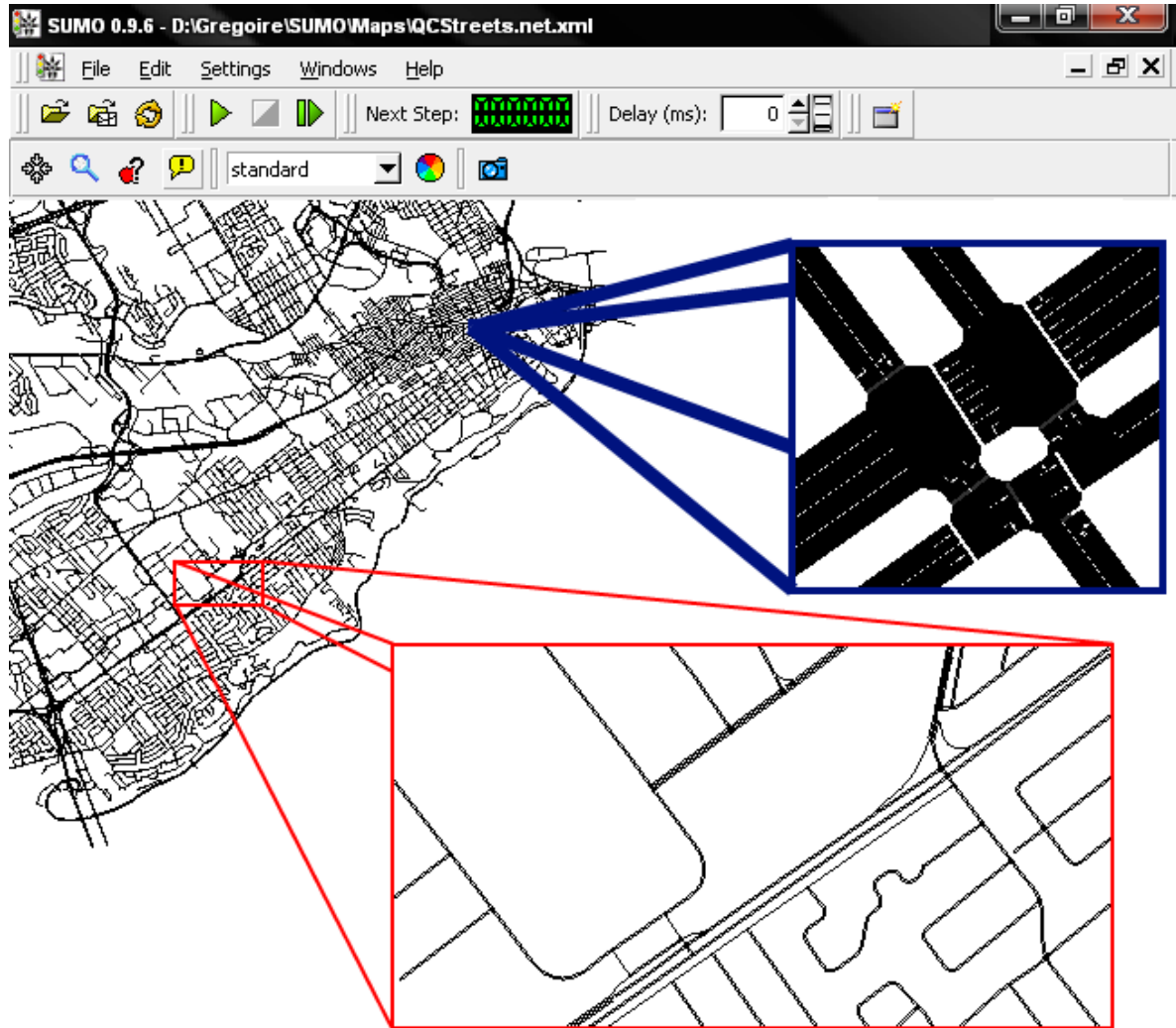


FIG. 6.8 – Réseau routier de la ville de Québec vu dans le simulateur SUMO.

60 secondes pour les véhicules provenant de chaque direction. La demande pour chaque connexion était de 2000 véhicules à l'heure. Le tableau 6.9 montre le débit des trois connexions en fonction du temps.

Lorsque la lumière est au rouge pour les véhicules provenant du nord, le débit est évidemment à zéro. Dès que la lumière passe au vert, le débit des trois connexions monte en flèche (sauf pour la première fois où le nombre de véhicules accumulés était moindre). Comme nous pouvons le voir, le débit maximal atteint est pratiquement le même pour les trois connexions, soit environ 2150 véhicules par heure. Ce chiffre peut donc être utilisé comme valeur par défaut par l'agent pour le débit maximal de chaque connexion, peu importe le type de celle-ci. Bien sûr, si le débit maximal observé en

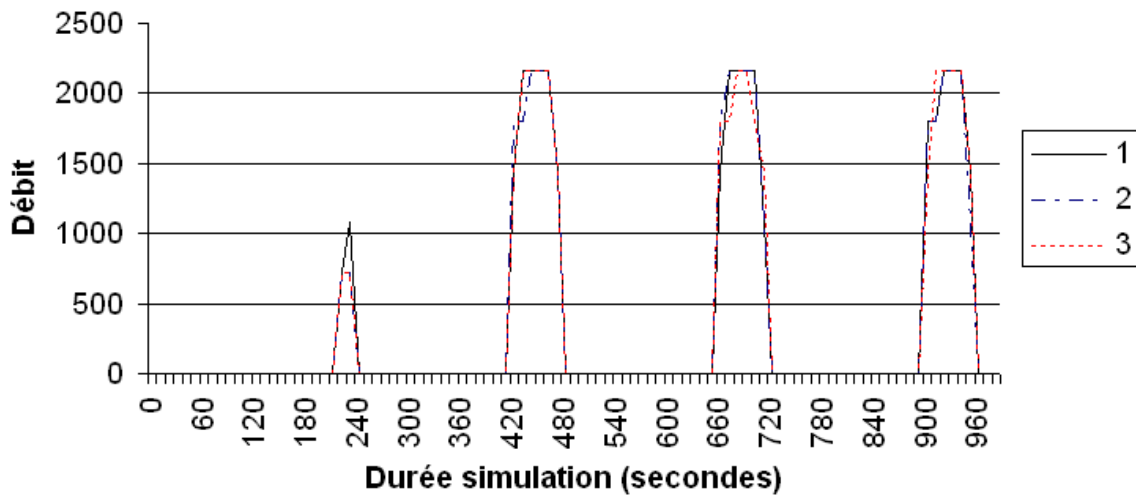


FIG. 6.9 – Débit maximum en fonction du temps pour les connexions 1, 2 et 3.

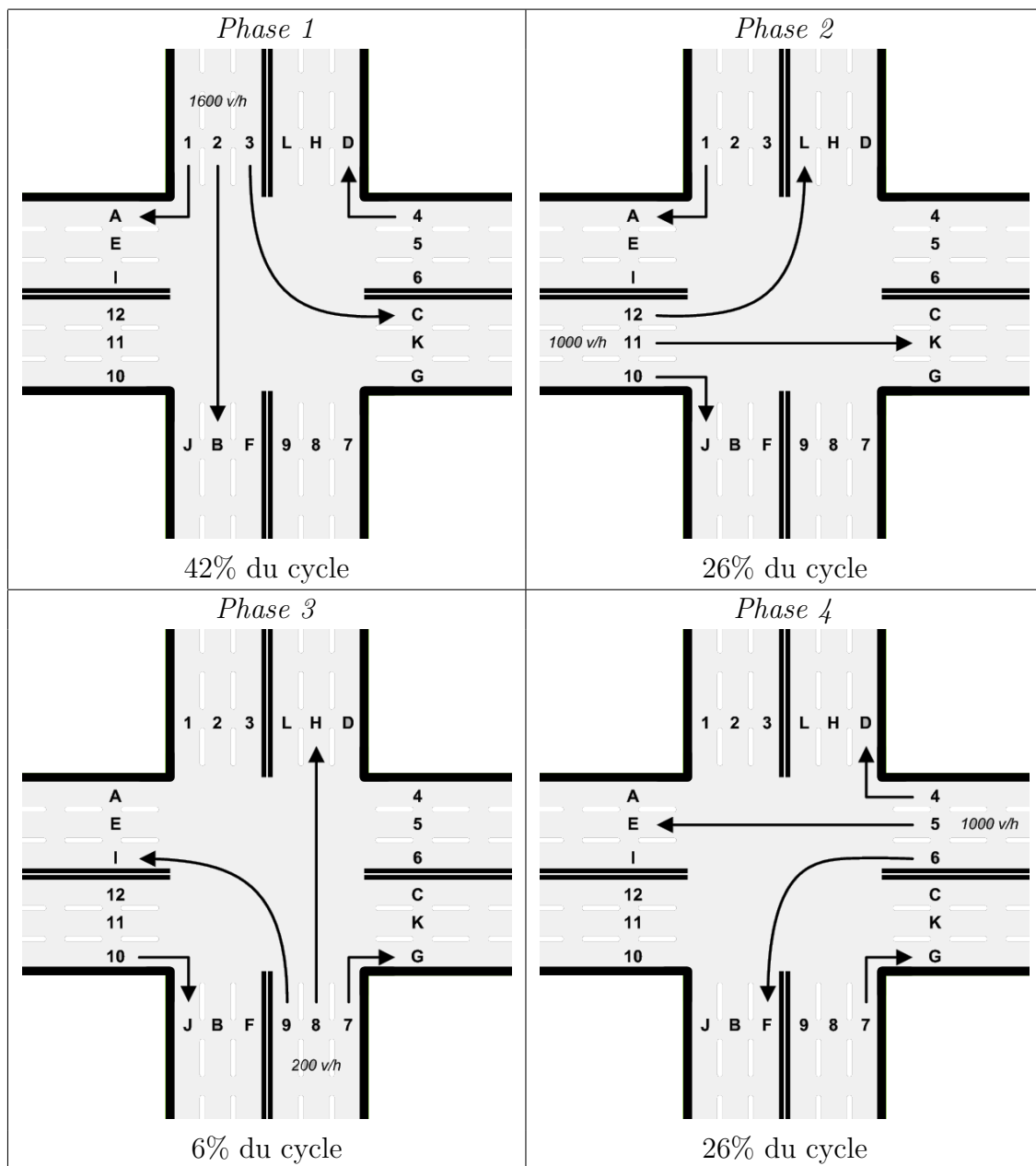
vient à dépasser ce nombre, l'agent conservera en mémoire le nouveau débit maximal.

Comparons maintenant les performances de notre agent par rapport à un contrôleur fixe. Le cycle choisi pour le contrôleur fixe est représenté dans le tableau 6.9. Il s'agit d'un cycle de 100 secondes allouant un pourcentage équitable à chaque direction en fonction de la demande.

La génération de véhicules s'effectue sur une période de deux heures. La simulation s'exécute jusqu'à ce que tous les véhicules aient atteint leur destination. Comme la demande est plus forte que la capacité de l'intersection, une congestion se forme sur les routes menant à l'intersection. Une fois la génération terminée, la simulation continue jusqu'à ce que tous les véhicules accumulés complètent leur parcours. La durée de la simulation est donc supérieure à deux heures.

Le tableau 6.10 montre le temps total de simulation pour les deux contrôleurs (fixe et agent) ainsi que la durée moyenne de parcours des véhicules pour chaque contrôleur. On peut voir que l'agent réussit à écouler le trafic plus rapidement que le contrôleur fixe, de près de 36 minutes. De plus, le temps de parcours des véhicules est diminué de plus de quatre minutes. La figure 6.10 montre la différence de congestion entre les deux contrôleurs.

Les résultats obtenus par l'agent montre que le cycle choisit par celui-ci est mieux adapté au modèle du trafic. Bien sûr, nous aurions pu calculer le cycle optimal de la



TAB. 6.9 – Cycle du contrôleur fixe.

| Contrôleurs | Temps de simulation | Temps moyen de parcourt par véhicule |
|-------------|---------------------|--------------------------------------|
| Fixe | 3 h. 39 min. | 22,7 minutes |
| Agent | 3 h. 03 min. | 18,4 minutes |

TAB. 6.10 – Temps de simulation et temps de parcourt moyen pour les deux contrôleurs.

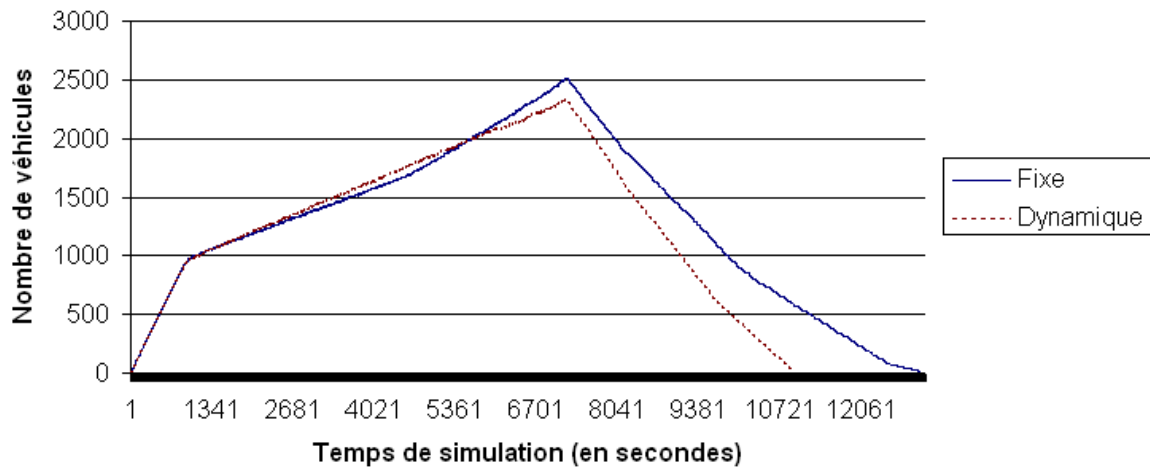


FIG. 6.10 – Nombre de voitures dans le réseau en fonction du temps.

même façon que l'agent et l'implanter dans le contrôleur fixe. Cependant, le but était d'utiliser un cycle simple et intuitif au niveau du contrôleur fixe et de voir l'amélioration obtenue par l'agent en utilisant un cycle optimal.

Le deuxième test consiste à tester les réactions de notre agent dans une situation où le modèle de trafic change dynamiquement. Nous utilisons un scénario où le modèle de trafic change à chaque heure. Les trois modèles utilisés sont présentés dans le tableau 6.11. Dans chacun des cas, la demande approche de très près la capacité de l'intersection. Le contrôleur doit donc choisir le cycle optimal pour ne pas créer de congestion.

La figure 6.11 montre les choix de cycles de l'agent en fonction du temps. La description des cycles se trouve dans le tableau 6.12. Nous pouvons voir que l'agent utilise un total de 10 cycles différents au cours de la simulation de trois heures. Pour chaque modèle de trafic, l'agent utilise d'un à trois cycles différents. Cela est dû au fait que certains cycles n'écoulent pas le trafic de chaque connexion de façon parfaitement uniforme. On se retrouve donc, après un certain temps, avec des connexions plus chargées que d'autres. L'agent s'adapte alors en choisissant un autre cycle pour équilibrer la demande de chaque connexion.

La figure 6.12 montre le nombre de véhicules en fonction du temps. Nous pouvons voir qu'au cours de la première heure, l'agent a de la difficulté à écouler le trafic dû à la plus forte demande. Cependant, lorsque le modèle de trafic change à la deuxième heure, l'agent réussit à écouler le trafic accumulé et à gérer en même temps la nouvelle

| Modèle 1 | | | | | | | |
|------------|------|------------|------|------------|------|-------------|-----|
| 1-A | 400 | 4-D | 50 | 7-G | 400 | 10-J | 50 |
| 2-B | 1200 | 5-E | 500 | 8-H | 1200 | 11-K | 500 |
| 3-C | 50 | 6-F | 0 | 9-I | 50 | 12-L | 0 |
| Modèle 2 | | | | | | | |
| 1-A | 50 | 4-D | 400 | 7-G | 50 | 10-J | 50 |
| 2-B | 500 | 5-E | 1200 | 8-H | 500 | 11-K | 50 |
| 3-C | 0 | 6-F | 400 | 9-I | 0 | 12-L | 50 |
| Modèle 3 | | | | | | | |
| 1-A | 50 | 4-D | 300 | 7-G | 50 | 10-J | 300 |
| 2-B | 50 | 5-E | 300 | 8-H | 50 | 11-K | 300 |
| 3-C | 800 | 6-F | 300 | 9-I | 800 | 12-L | 300 |

TAB. 6.11 – Répartition de la demande pour chaque connexion de chaque modèle. La demande est exprimée en véhicules par heure.

demande. Nous pouvons d’ailleurs remarquer qu’il crée un cycle uniquement pour cette période de transition (cycle numéro 3) et qu’il ne le réutilise plus par la suite. Au cours de la deuxième et troisième heure, l’agent maintient un niveau de trafic assez bas en fonction de la demande.

Ces résultats montrent que l’agent est non seulement capable d’adopter un cycle optimal en fonction du modèle de trafic mais s’adapte aussi très rapidement aux changements.

6.3.7 Conclusion

Les résultats obtenus dans ce chapitre montrent que l’utilisation de la communication route-à-véhicule de pair avec l’utilisation d’agents dynamiques aux intersections permet de traiter le problème d’optimisation des feux de signalisation de façon efficace. Les tests effectués pour une intersection donnent de très bons résultats. De plus, le temps d’exécution des simulations est minime comparativement à l’exécution de simulations impliquant de l’apprentissage par renforcement. Par exemple, simuler un scénario de trois heures impliquant une intersection et des milliers de véhicules s’effectue en moins de trente secondes avec cette approche, alors qu’obtenir une politique à l’aide de l’apprentissage par renforcement pouvait prendre plusieurs dizaines de minutes.

La deuxième étape serait d’implémenter et de tester les règles de coordination

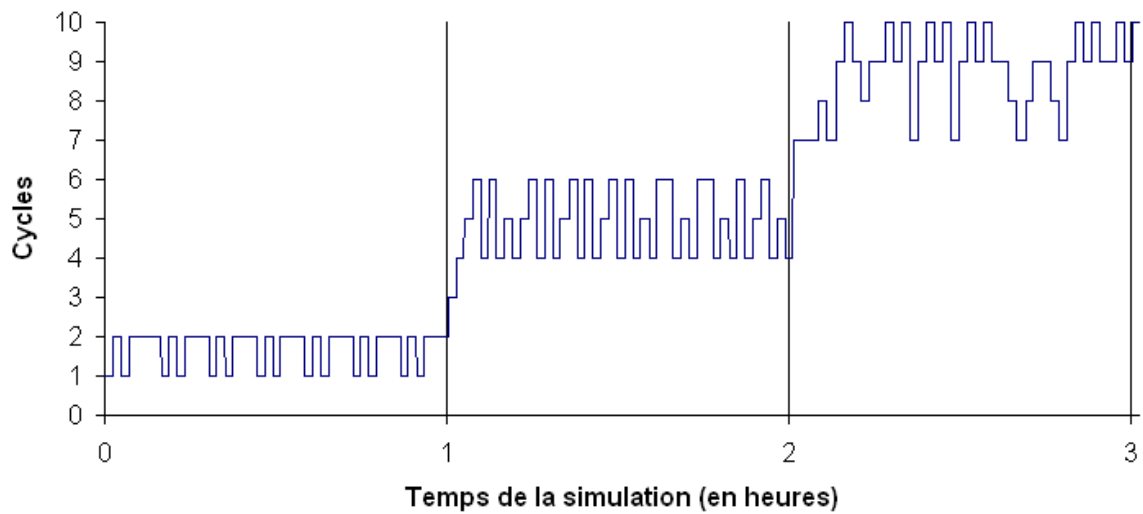


FIG. 6.11 – Choix du cycle par l'agent.

décrites dans cette section. Une bonne partie de ce travail a été fait au cours de cette maîtrise et les résultats préliminaires étaient encourageants. Cependant, le temps manque pour présenter des résultats plus approfondis.

| Cycle | Phase 1 | Phase 2 | Phase 3 | Phase 4 |
|-------|-----------|----------|------------|------------|
| 1 | 4-5-10-11 | 1-2-7-8 | | |
| 2 | 4-5-10-11 | 4-3-9-10 | 1-2-7-8 | |
| 3 | 4-5-10-11 | 6-1-7-12 | 1-2-7-8 | |
| 4 | 4-5-6-7 | 1-2-7-8 | | |
| 5 | 4-5-6-7 | 4-5-7-10 | 1-2-7-8 | 1-10-11-12 |
| 6 | 4-5-6-7 | 1-2-7-8 | 1-10-11-12 | |
| 7 | 4-5-10-11 | 4-3-9-10 | 6-1-7-12 | 1-2-7-8 |
| 8 | 4-5-10-11 | 4-3-9-10 | 6-1-7-12 | |
| 9 | 4-5-6-7 | 4-3-9-10 | 1-2-7-8 | 1-10-11-12 |
| 10 | 4-5-6-7 | 4-3-9-10 | 1-10-11-12 | |

TAB. 6.12 – Cycles choisis par l'agent, chaque phase est représentée par la liste de connexions qu'elle contient.

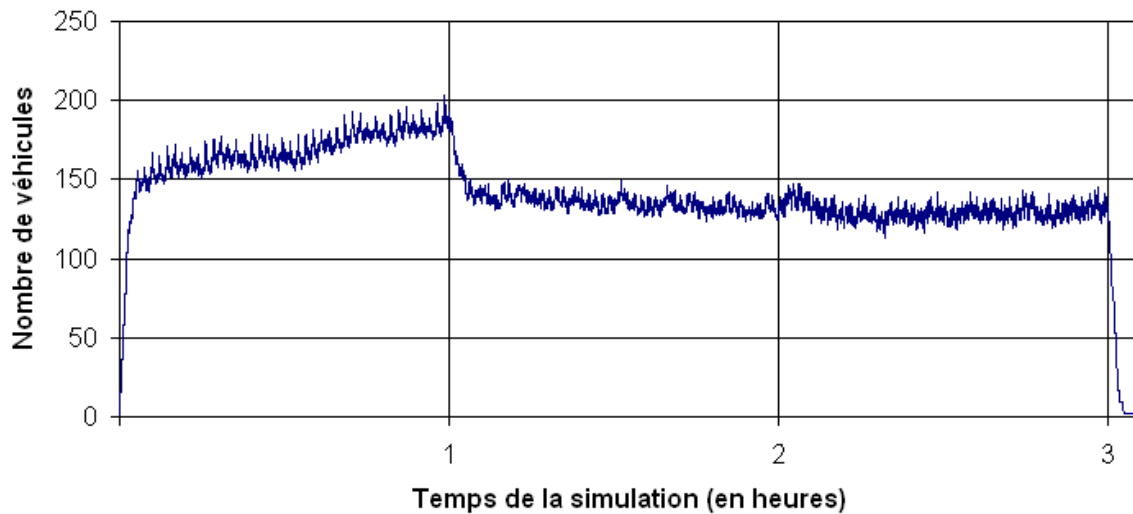


FIG. 6.12 – Nombre de véhicules en fonction du temps.

Chapitre 7

Conclusion

Ce mémoire commence avec une brève présentation des systèmes de transport intelligents et la description des problèmes qui sont abordés dans cette maîtrise. La motivation d'intégrer la communication inter-véhicules et route-à-véhicule dans les systèmes de transport intelligents est ensuite expliquée suivie des principaux objectifs de ce mémoire et de la présentation du projet Auto21. Le chapitre suivant survole le champ de l'intelligence artificielle pour mieux cibler la position du mémoire dans ce domaine complexe.

Ensuite, les principales notions de communications sont expliquées selon l'ordre suivant : (i) la communication entre ordinateurs, (ii) la communication sans fil et (iii) les réseaux mobiles et la communication inter-véhicules. Après l'explication de ces notions, nous entrons dans le chapitre portant sur le premier but de ce mémoire : l'apprentissage de la communication inter-véhicules. En premier lieu, on y trouve la description du système de régulateur de vitesse adaptatif et collaboratif dans lequel la communication sera implantée. Viennent ensuite l'explication des processus décisionnels de Markov et l'implémentation de la communication véhiculaire dans le système de régulateur de vitesse. Le chapitre conclut avec la présentation de l'environnement de test et les résultats d'apprentissage de la communication inter-véhicules.

La deuxième partie du mémoire porte sur le problème de la congestion urbaine et sur l'optimisation des feux de signalisation à l'aide de la communication route-à-véhicule. On y voit d'abord l'ampleur du problème de congestion dans les grandes villes du monde ainsi que les principales notions relatives au trafic et à la circulation sur les réseaux routiers. L'approche proposée dans le mémoire est ensuite présentée : utiliser la communication route-à-véhicule ainsi que la théorie multi-agents pour optimiser le fonctionnement des feux de signalisation. Le dernier chapitre explique le problème d'optimisation des feux et la modélisation qui peut être utilisée pour traiter ce problème. Deux méthodes

sont ensuite présentées, une utilisant l'apprentissage par renforcement et l'autre utilisant une approche dynamique. Les deux méthodes sont analysées et présentées avec leurs résultats respectifs.

7.0.8 Contributions

Ce mémoire apporte plusieurs contributions au niveau des systèmes de transport intelligents ainsi qu'au niveau de l'apprentissage machine. Ces contributions peuvent être présentées selon les objectifs énoncés dans la section 1.3.

- Plusieurs techniques d'apprentissage machine ont été étudiées et analysées de façon à les appliquer au domaine de la communication inter-véhicules dans les transports. Il a été démontré que les algorithmes d'apprentissage machine s'intègrent particulièrement bien à ce domaine.
- Une solution de communication pour les systèmes d'aide à la conduite a aussi été proposée. Plus précisément, une solution utilisant l'apprentissage par renforcement a été appliquée à un système de régulateur de vitesse adaptatif et collaboratif pour y intégrer la communication inter-véhicules et améliorer l'efficacité du système. Les résultats obtenus sont très prometteurs et viennent enrichir l'état de l'art en la matière.
- Finalement, deux solutions ont été proposées pour le contrôle des feux de signalisation dans le but de diminuer la congestion urbaine. La première utilise l'apprentissage par renforcement et est présentée en détail dans [Grégoire *et al.* \[2007\]](#). La deuxième méthode utilise une approche dynamique et donne d'excellents résultats tout en offrant des temps de calcul très faibles.

7.0.9 Travaux futurs

Les travaux faits au cours de cette maîtrise forment une très bonne base à l'étude de la communication inter-véhicules. Ils démontrent que l'apprentissage par renforcement est une solution viable au problème de communication dans le domaine des transports. Cependant, il serait intéressant de continuer les recherches en utilisant d'autres techniques d'apprentissage machine et plus particulièrement en intégrant l'incertain dans les algorithmes d'apprentissage. En effet, les réseaux mobiles sont très complexes et la propagation des ondes est loin d'être un domaine déterministe et complètement observable. Dans bien des cas, les messages sont perdus ou parviennent de façon altérée aux destinataires.

Pour ce qui est de l'optimisation des feux de signalisation, il s'agit d'un champ de recherche très important et actuel puisque la congestion urbaine est un problème de plus en plus présent dans nos villes. Il serait intéressant de continuer le développement de l'approche dynamique en intégrant dans l'algorithme de la collaboration entre les agents gérants les intersections. Il serait alors possible d'offrir un réseau de signalisation efficace qui optimiserait la synchronisation entre les intersessions. De plus, il faudrait étendre les travaux pour inclure les transports en commun puisqu'ils sont probablement la seule solution efficace à long terme pour diminuer le trafic urbain.

Annexe A

Urban Traffic Control Based on Learning Agents

Bibliographie

- AGARWAL, M., MAZE, T. H. et SOULEYRETTE, R. R. (2005). Impacts of weather on urban freeway traffic flow characteristics and facility capacity. *In Proceedings of the 2005 Mid-Continent Transportation Research Symposium*.
- ARMSTRONG, L. (2005). Dedicated short range communications (dsrc) home. <http://www.learmstrong.com/DSRC/DSRCHomeset.htm>. [En ligne - Page disponible le 20 janvier 2008].
- ASTM (2008). Astm international standards worldwide. <http://www.astm.org/>. [En ligne - Page disponible le 30 novembre 2007].
- AUTO21 (2008). The automobile of the 21st century. http://www.auto21.ca/home_f.html. [En ligne - Page disponible le 16 avril 2008].
- BAZZAN, A. L. C. (2005). A distributed approach for coordination of traffic signal agents. *Autonomous Agents and Multi-Agent Systems (AAMAS'05)*, 10:131–164.
- BISHOP, R. (2005). *Intelligent Vehicle Technology and Trends*. Artech House.
- BORGONOVO, F., CAMPPELLI, L., CESANA, M., COLETTI, L. et di MILANO, P. (2003). Mac for ad-hoc inter-vehicle network : services and performance. *In Proceedings of the Vehicular Technology Conference (VTC'03)*, pages 2789–2793.
- BROCKFELD, E., KÜHNE, R. D. et WAGNER, P. (2004). Calibration and validation of microscopic traffic flow models. *Transportation Research Record*, 1876:62–70.
- BULL, A. (2004). *Traffic Congestion : the problem and how to deal with it*. United Nations Publications.
- BURGHOUT, W. (2004). *Hybrid microscopic-mesosopic traffic simulation*. Royal Institute of Technology.
- BUTTON, K. J. et HENSHER, D. A. (2001). *Handbook of Transport Systems and Traffic Control*. Elsevier Pergamon.
- CONTI, M., MASELLI, G., TURI, G. et GIORDANO, S. (2004). Cross-layering in mobile ad hoc network design. *Computer - the flagship publication of the IEEE Computer Society*, 37(2):48–51.

- da SILVA, B. C., de OLIVERIA, D., BAZZAN, A. L. C. et BASSO, E. W. (2006). Adaptive traffic control with reinforcement learning. *In Conference on Autonomous Agents and Multiagent Systems (AAMAS'06)*, pages 80–86.
- DERYCKE, P.-H. (1997). *Le péage urbain : histoire - analyse - politiques*. Economica.
- DESJARDINS, C. (2008). Cooperative adaptive cruise control : A learning approach. Mémoire de D.E.A., Université Laval.
- DRESNER, K. et STONE, P. (2006). Traffic intersections of the future. *In American Association for Artificial Intelligence (AAAI'06)*, pages 1593–1596.
- ERNST, T. et INRIA, K. U. (2002). Connecting automobiles to the internet. *In The 3rd International Workshop on ITS Telecommunications (ITST'02)*, pages 61–66.
- FESTAG, A., FÜSSLER, H., HARTENSTEIN, H., SARMA, A. et SCHMITZ, R. (2004). Fleetnet : Bringing car-to-car communication into the real world. *In Proceedings of the 11th World Congress on ITS*, pages 1–8, Nagoya, Japan.
- FLEETNET (2002). About fleetnet. <http://www.et2.tu-harburg.de/fleetnet/english/about.html>. [En ligne - Page disponible le 23 février 2008].
- GÜNTER, Y. et GROSSMANN, H. P. (2005). Usage of wireless lan for inter-vehicle communication. *In IEEE Conference on Intelligent Transportation Systems (ITSC'05)*, pages 296–301.
- GRÉGOIRE, P.-L., DESJARDINS, C., LAUMÔNIER, J. et CHAIB-DRAA, B. (2007). Urban traffic control based on learning agents. *In Proceedings of Intelligent Transportation Systems Conference (ITSC'07)*.
- HARTENSTEIN, H., BOCHOW, B., EBNER, A., LOTT, M., RADIMIRSCH, M. et VOLLMER, D. (2001). Position-aware ad hoc wireless networks for inter-vehicle communications : the fleetnet project. *In Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, pages 258–262.
- HEUNG, T. H., HO, T. K. et FUNG, Y. F. (2005). Coordinated road-junction traffic control by dynamic programming. *IEEE Transaction on Intelligent Transportation Systems*, 6(3):341–350.
- IEEE (2006). Ieee-1609 - family of standards for wireless access in vehicular environments (wave). http://www.standards.its.dot.gov/fact_sheet.asp?f=80. [En ligne - Page disponible le 14 janvier 2008].
- IEEE (2008a). Ieee 802.11 - the working group setting the standards for wireless lans. <http://ieee802.org/11/>. [En ligne - Page disponible le 17 août 2008].
- IEEE (2008b). Ieee web site. <http://www.ieee.org>. [En ligne - Page disponible le 17 août 2008].
- IMPROTA, G. et CANTARELLA, G. E. (1984). Control system design for an individual signalized junction. *Transportation research - Part B : methodological*, 18B(2):147–168.

- JATINDER PAL SINGH, Nicholas Bambos, B. S. et CLAWIN, D. (2004). Wireless lan performance under varied stress conditions in vehicular traffic scenarios. *In The First Annual International Conference on Mobile and Ubiquitous Systems : Networking and Services (MobiQuitous'04)*, pages 114–123.
- KRAJZEWICZ, D., BONERT, M. et WAGNER, P. (2006). The open source traffic simulation package sumo. *In RoboCup 2006 Infrastructure Simulation Competition*, Germany, Bremen.
- KRAUSS, S. (1998). *Microscopic Modeling of Traffic Flow : Investigation of Collision Free Vehicle Dynamics*. Thèse de doctorat, Mathematisches Institut, Universität zu Köln.
- LAUMONIER, J. (2008). *Méthodes d'apprentissage de la coordination multiagent : application au transport intelligent*. Thèse de doctorat, Département d'Informatique et de Génie Logiciel, Université Laval, Québec, Canada.
- LIANG, C.-Y. et PENG, H. (2000). String stability analysis of adaptive cruise controlled vehicles. Rapport technique, University of Michigan.
- MOSKE, M., FUSSLER, H., HARTENSTEIN, H. et FRANZ, W. (2004). Performance measurements of a vehicular ad hoc network. *In Vehicular Technology Conference (VTC'04)*, volume 4, pages 2116–2120.
- OCDE (1994). *Gérer la congestion et la demande de trafic routier*. Éditions OCDE.
- PAPAGEORGIU, M. et KOTSIALOS, A. (2000). Freeway ramp metering : an overview. *In Proceedings of Intelligent Transportation Systems Conference (ITSC'00)*, pages 228–239.
- ROBERTSON, D. I. et BRETHERTON, R. D. (1991). Optimizing networks of traffic signal in real time – the SCOOT method. *IEEE Transaction on Vehicular Technology*, 40(1):11–15.
- RUSSELL, S. J. et NORVIG, P. (2003). *Artificial Intelligence : A Modern Approach*. Pearson Education.
- SCHRANK, D. et LOMAX, T. (2007). The 2007 urban mobility report. Rapport technique, Texas Transportation Institute.
- TANENBAUM, A. (2003). *Réseaux*. Pearson Education France.
- TANK, T. et LINNARTZ, J.-P. M. G. (1997). Vehicle-to-vehicle communications for avcs platooning. *IEEE Transaction on Vehicular Technology*, 46:528–536.
- THORPE, T. (1997). Vehicle traffic light control using sarsa. Rapport technique, Colorado State University.
- THORPE, T. L. et ANDERSON, C. W. (1996). Traffic light control using sarsa with three state representations. Rapport technique, IBM Corporation.

- TRANSPORT QUÉBEC (2008a). Transports québec : Carefours giratoires. http://www.mtq.gouv.qc.ca/portal/page/portal/grand_public/vehicules_promenade/reseau_routier/carrefours_giratoires. [En ligne - Page disponible le 24 janvier 2008].
- TRANSPORT QUÉBEC (2008b). Transports québec : Feux de gestion d'accès. http://www.mtq.gouv.qc.ca/portal/page/portal/grand_public/vehicules_promenade/reseau_routier/signalisation/signaux_lumineux/feux_gestion_acces. [En ligne - Page disponible le 24 janvier 2008].
- TRANSPORTS CANADA (2006). Le coût de la congestion urbaine au canada. Rapport technique, Transports Canada.
- TURCOTTE, M. (2005). Le temps pour se rendre au travail et en revenir. Enquête sociale sur l'emploi du temps : cycle 19 89-622-XIF, Statistique Canada.
- VINCENT, R. et YOUNG, C. (1986). Self-optimising traffic signal control using micro-processors - the TRRL 'MOVA' strategy for isolated intersections. *Traffic Engineering and Control*, 27(7):385–387.
- WALL, B. (2008). Computer chess history. <http://www.geocities.com/SiliconValley/Lab/7378/comphis.htm>. [En ligne - Page disponible le 10 novembre 2007].
- WIERING, M. (2000). Multi-Agent Reinforcement Learning for Traffic Light Control. *In Seventeenth International Conference on Machine Learning and Applications (ICMLA '00)*, pages 1151–1158.
- WIKIPÉDIA (2007). Hoy no circula. <http://en.wikipedia.org/w/index.php?title=Routing&oldid=183412465>. [En ligne - Page disponible le 23 janvier 2008].
- WIKIPÉDIA (2008a). Routing. <http://en.wikipedia.org/w/index.php?title=Routing>. [En ligne - Page disponible le 16 janvier 2008].
- WIKIPÉDIA (2008b). Révolution industrielle. http://fr.wikipedia.org/w/index.php?title=R%C3%A9volution_industrielle&oldid=27083869. [En ligne ; Page disponible le 7-mars-2008].
- YANG, X., LIU, J., ZHAO, F. et VAIDYA, N. H. (2004). A vehicle-to-vehicle communication protocol for cooperative collision warning. *In The First Annual International Conference on Mobile and Ubiquitous Systems : Networking and Services (MobiQuitous'04)*.
- YIN, J., ELBATT, T., YEUNG, G., RYU, B., HABERMAS, S., KRISHNAN, H. et TALTY, T. (2004). Performance evaluation of safety applications over dsrc vehicular ad hoc networks. *In Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks (VANET'04)*.
- ZHANG, H. M. et RITCHIE, S. G. (1997). Freeway ramp metering using artificial neural networks. *Transportation Research - Part C : Emerging Technologies*, 5:273–286.