



**HAL**  
open science

# Contrôle et gestion du trafic routier urbain par un réseau de capteurs sans fil

Sébastien Faye

► **To cite this version:**

Sébastien Faye. Contrôle et gestion du trafic routier urbain par un réseau de capteurs sans fil. Infrastructures de transport. Télécom ParisTech, 2014. Français. NNT : 2014ENST0061 . tel-01398839

**HAL Id: tel-01398839**

**<https://pastel.hal.science/tel-01398839>**

Submitted on 17 Nov 2016

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

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



École doctorale Informatique, Télécommunications et Électronique (Paris)

## Doctorat ParisTech

# THÈSE

pour obtenir le grade de docteur délivré par

**TELECOM ParisTech**

**Spécialité « Informatique et Réseaux »**

*présentée et soutenue publiquement par*

**Sébastien FAYE**

le 13 octobre 2014

## **Contrôle et gestion du trafic routier urbain par un réseau de capteurs sans fil**

Encadrant de thèse : **M. Claude CHAUDET**  
Directeur de thèse : **Mme Isabelle DEMEURE**

### Jury

**M. Anthony BUSSON**, Professeur, École Normale Supérieure de Lyon (France)

**M. Marco FIORE**, Maître de conférences, CNR - IEIIT (Italie)

**Mme Bénédicte LE GRAND**, Professeur, Université Paris 1 Panthéon - Sorbonne (France)

**M. Nicolas SAUNIER**, Maître de conférences, Polytechnique Montréal (Canada)

**M. Claude CHAUDET**, Maître de conférences, Télécom ParisTech (France)

**Mme Isabelle DEMEURE**, Professeur, Télécom ParisTech (France)

Rapporteur

Rapporteur

Examinateur

Examinateur

Encadrant de thèse

Directeur de Thèse

**TELECOM ParisTech**

école de l'Institut Mines-Télécom - membre de ParisTech

46 rue Barrault 75013 Paris - (+33) 1 45 81 77 77 - [www.telecom-paristech.fr](http://www.telecom-paristech.fr)



*L'imagination est la meilleure compagnie de transport au monde*

Roger Fournier



---

## Résumé

---

Les transports terrestres occupent une place majeure dans notre société, notamment en ville où les ralentissements aux heures de pointe peuvent avoir un impact notable sur l'organisation des activités, l'économie ou encore l'écologie. Les nouvelles technologies de l'information et de la communication ont permis, en l'espace de quelques années, de mettre en œuvre des *systèmes de transport intelligents*. À l'aide de multiples points de mesures répartis sur le territoire, un opérateur peut dénombrer les usagers et en déduire la charge du réseau. Il peut, par ailleurs, réagir aux variations du trafic en régulant dynamiquement les flux de véhicules. Les équipements tels que les feux de circulation deviennent alors des éléments actifs dans la réduction des embouteillages et dans la gestion de la sécurité, améliorant globalement l'expérience des usagers.

Les infrastructures routières sont généralement coordonnées par un centre de contrôle, responsable du maintien des équipements, de leurs réglages initiaux et de la gestion des incidents (matériels ou humains). Toutefois, centraliser les informations présente de nombreuses limites. Tout d'abord, cette logique impose à la plupart des équipements routiers d'être reliés en permanence à un centre de contrôle. Cela complique le déploiement de systèmes sur des zones isolées, en plus de nécessiter un câblage conséquent. Ensuite, la tolérance aux pannes est faible, car le réseau repose sur une entité centrale unique, au mieux sur une poignée. Enfin, d'une manière générale, utiliser une architecture centralisée implique un temps de latence qui peut restreindre la capacité des applications à répondre à certains événements.

Cette thèse vise à étudier l'emploi de systèmes distribués afin de mettre en œuvre des systèmes de transport intelligents grâce à un *réseau de capteurs sans fil*. Couplés à une unité de détection (p. ex., un magnétomètre), les capteurs communicants peuvent réagir au passage d'un véhicule en étant déployés, par exemple, sur les voies. Ils sont également capables de coopérer et de s'affranchir d'une entité centrale, rendant tout ou partie d'une zone urbaine totalement indépendante. D'autre part, ces réseaux peuvent fonctionner de manière autonome et tolèrent mieux les pannes, car aucun élément n'est indispensable au fonctionnement global du système. Enfin, les éléments de ces réseaux sont petits, peu coûteux, et communiquent en sans fil, ce qui leur permet d'être déployés et redéployés rapidement et de manière dense.

Cette thèse détaille trois contributions principales, visant à démontrer la pertinence et l'intérêt que de tels systèmes peuvent représenter, face aux contraintes fixées par l'infrastructure routière actuelle.

1. Tout d'abord, nous présentons un modèle de graphe de terrain et analysons le déploiement d'un réseau de capteurs sans fil sur plus d'une cinquantaine de villes. Nous montrons en particulier que la distribution des degrés des graphes générés suit une loi gamma et qu'ils présentent de nombreuses composantes connexes. Cette étude peut servir de base à l'évaluation de stratégies de routage ou à l'évaluation de déploiement de stations de base. Pour cette thématique, une plateforme de simulation a notamment été développée [Fay14a].
2. Cette thèse présente ensuite plusieurs contributions relatives à la gestion des feux de circulation. Dans les implantations actuelles, les feux de circulation suivent généralement une séquence fixe, prédéterminée et indépendante de l'état du réseau. Ceci a tendance à créer de nombreux ralentissements, alors qu'une gestion plus dynamique permettrait de fluidifier le trafic, en allant jusqu'à prévenir ou déconstruire des embouteillages. Nous présentons deux algorithmes de contrôle des feux de circulation, TAPIOCA et BASALTE, qui agissent localement au niveau des intersections. Le premier est conçu pour les situations où le trafic est normal tandis que le second est dédié aux cas de saturation. Les simulations réalisées montrent le potentiel de ces approches décentralisées et évaluent leurs coûts et performances face à des solutions plus classiques.
3. Enfin, nous considérons les aspects réseaux, en étudiant l'influence de l'utilisation d'un support de communication sans fil, dont la fiabilité est imparfaite, sur nos algorithmes. Nous proposons notamment un mécanisme d'interpolation permettant de compenser les pertes d'informations. Nous évaluons ce mécanisme à l'aide d'une co-simulation entre SUMO (un simulateur transport) et OMNeT++ (un simulateur réseau).

---

## Abstract

---

Road traffic has a significant effect on metropolitan activities, especially during peak hours when it impacts on areas such as the economy and the environment. During the past few years, new technologies in the fields of information and communication have led to the introduction of *intelligent transportation systems*. Using multiple measurement points distributed across a country, an operator can count road users and calculate the network load. An operator can also respond to changes in traffic by dynamically regulating the flow of vehicles. Roadside objects like traffic lights and other signs and signals become active elements in reducing congestion, managing safety, and generally improving the road-user's experience.

Road infrastructure is typically coordinated from a control centre that is responsible for maintaining not only its equipment but also their initial settings and incident management (both material and human). However, the centralization of information has a number of drawbacks. First of all, most roadside infrastructure must be permanently connected to a control centre, which complicates the deployment of systems in isolated areas and requires expensive cabling. Furthermore, fault tolerance is low because the network relies on a single central entity – or at most a handful of entities. Finally, the use of centralized architecture generally leads to delays that restrict the ability of applications to respond to certain events.

The aim of this thesis is to study the use of distributed systems in order to implement intelligent transportation systems via a *wireless sensor network*. Coupled to a detection unit (e.g., a magnetometer), the interconnected sensors can respond to the passage of a vehicle when deployed, for example, along the road. They can also work together without recourse to a central entity – rendering all or part of an urban area totally independent. Furthermore, these networks can operate autonomously and are less susceptible to breakdown, because the overall running of the system is not affected by the failure of individual components. Finally, components are small and cheap, and they operate wirelessly, which means they can be deployed and redeployed both rapidly and densely.

The three main contributions of this thesis aim to demonstrate the relevance and the value of these systems in the face of the constraints of existing road infrastructure.

1. First, we present a terrain graph model and analyze the deployment of wireless sensor networks in more than fifty cities. In particular, we demonstrate that the degree distribution of the generated graphs follows a gamma distribution with many connected components. This study can be used as a basis for assessing

routing strategies or evaluating the deployment of base stations. A simulation platform has been developed to this purpose [Fay14a].

2. This thesis also presents several contributions that focus on traffic light management. In their conventional role, traffic lights generally follow a fixed sequence – predetermined and independent of the network state, and which has a tendency to create holdups. More dynamic management, on the other hand, can improve traffic flow by averting or easing traffic congestion. We propose two algorithms for controlling traffic lights, TAPIOCA and BASALTE, both of which act locally at intersections. The first is designed for situations where traffic flow is normal, while the second was specifically created for instances of saturation. Simulations have demonstrated the great potential of these decentralized approaches and favourably assessed their costs and performance against those of more traditional solutions.
3. Finally, we consider networking aspects by studying the influence of the use of wireless communication, whose reliability is not perfect in our algorithms. In particular, we suggest an interpolation mechanism that compensates for loss of data. We evaluate this mechanism using a co-simulation between SUMO (a transport simulator) and OMNeT++ (a network simulator).

---

## Remerciements

---

J'aimerais avant tout adresser ma gratitude aux deux personnes sans qui cette thèse n'aurait pas été possible, et qui ont su m'apporter autonomie, motivation et l'idée qu'un travail de recherche était loin d'être un travail solitaire. Tout d'abord, à mon encadrant de thèse, CLAUDE CHAUDET, qui a été présent à chaque étape durant ces trois dernières années, en me guidant et en m'apprenant énormément sur le plan scientifique. Ensuite, à ma directrice de thèse, ISABELLE DEMEURE, qui a toujours porté d'un œil bienveillant et encourageant ce travail, en m'apportant de nombreux conseils et une certaine manière de m'organiser (ce qui n'était pas gagné d'avance).

Mes remerciements vont également à ANTHONY BUSSON et à MARCO FIORE, qui ont accepté la dure tâche de rapporter ce manuscrit. De même, je tiens à remercier BÉNÉDICTE LE GRAND et NICOLAS SAUNIER, pour avoir accepté d'évaluer ce travail en tant que membres du jury.

Je souhaite remercier toutes les personnes qui ont œuvré à la mise en place de cette thèse, fruit d'une collaboration entre Télécom ParisTech, l'Edite de Paris et le CNAM. Je remercie particulièrement CHRISTOPHE PICOULEAU, en souvenir de débuts bien difficiles.

Certaines réflexions et résultats présentés dans ce manuscrit sont le fruit de rencontres. Je remercie THIERRY DELAPORTE, responsable du poste de régulation du trafic d'Amiens, pour m'avoir ouvert les yeux sur la réalité du terrain. Je remercie également la communauté et les contributeurs des outils utilisés dans cette thèse.

D'une manière générale, j'adresse mes remerciements à toutes les personnes ayant pu participer, de près ou de loin, à la réalisation de cette thèse et que j'ai pu rencontrer durant ces trois dernières années, notamment les membres du LINCS et de la salle 50, qui se reconnaîtront.

Sur un plan plus personnel, je souhaiterais remercier ma famille et mes amis, qu'ils soient Amiénois, Cannois, Compiégnois, Limougeauds ou Toulousains. J'adresse une pensée toute particulière à ma sœur et à mes parents, JULIE, DANIEL et PASCALE, ainsi qu'à ma très chère grand-mère, NICOLE, pour leur présence et encouragements.

*Last but not least*, une mention spéciale à CLAIRE pour son soutien, son affection sans failles et sa patience, qui font d'elle quelqu'un d'exceptionnel.

*La reconnaissance est la mémoire du cœur* – Henri Lacordaire



---

## Table des matières

---

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Un réseau complexe et critique . . . . .	1
1.2	Contrôle et suivi des usagers à l'aide de systèmes de transport intelligents . . . . .	3
1.3	Le cas particulier des feux de circulation . . . . .	4
1.4	De nouvelles problématiques . . . . .	5
1.5	Diviser pour mieux régner . . . . .	6
1.6	Contribution . . . . .	6
<b>2</b>	<b>Des systèmes de transport intelligents pour gérer le trafic routier</b>	<b>9</b>
2.1	Modélisation du trafic routier urbain . . . . .	9
2.1.1	Modèles microscopiques . . . . .	11
2.1.2	Modèles macroscopiques . . . . .	12
2.1.3	Le diagramme fondamental du trafic . . . . .	13
2.2	Le cas particulier des feux de circulation . . . . .	14
2.2.1	Fonctionnement traditionnel d'une intersection . . . . .	14
2.2.2	Gestion des séquences de feux . . . . .	15
2.2.3	Gestion du temps de feux . . . . .	17
2.3	Contrôle du trafic routier urbain . . . . .	22
2.3.1	Métriques et paramètres traditionnels . . . . .	23
2.3.2	Types de contrôles . . . . .	23
2.3.3	Modes de fonctionnement d'un contrôleur de feux . . . . .	25
2.3.4	Les systèmes de régulation du trafic . . . . .	26
2.3.5	Les équipements de détection . . . . .	28
2.4	Réseaux de communication . . . . .	30
2.4.1	Les réseaux statiques . . . . .	30
2.4.2	Les réseaux véhiculaires . . . . .	31
2.4.3	Les systèmes coopératifs . . . . .	31
2.5	Distribution du réseau de communication . . . . .	33
2.5.1	Les systèmes distribués . . . . .	33
2.5.2	Les réseaux de capteurs sans fil . . . . .	34
<b>3</b>	<b>Déploiement et caractérisation de la topologie d'un réseau distribué pour la gestion routière</b>	<b>37</b>
3.1	Topologie du réseau . . . . .	38
3.2	Stratégies de déploiement . . . . .	39

3. 2. 1	Sur une intersection . . . . .	39
3. 2. 2	Sur une large zone . . . . .	41
3. 3	Modélisation d'un déploiement de capteurs à large échelle . . . . .	42
3. 3. 1	Stratégie de base . . . . .	42
3. 3. 2	Réduction du nombre de capteurs . . . . .	42
3. 3. 3	Création du graphe de connectivité . . . . .	45
3. 4	Création du jeu de données . . . . .	46
3. 4. 1	Méthode et outils . . . . .	46
3. 4. 2	Sélection de six scénarios représentatifs . . . . .	47
3. 5	Analyse du graphe de connectivité . . . . .	51
3. 5. 1	Degré des nœuds . . . . .	51
3. 5. 2	Coefficient de clustering . . . . .	54
3. 5. 3	Rapport avec la création de graphes aléatoires . . . . .	56
3. 6	Analyse du réseau . . . . .	57
3. 6. 1	Partitionnement du réseau . . . . .	57
3. 6. 2	Composantes connexes . . . . .	58
3. 6. 3	A l'intérieur des composantes connexes . . . . .	59
3. 6. 4	Analyse de fiabilité . . . . .	62
3. 7	Amélioration du déploiement . . . . .	63
3. 8	Analyse des autres stratégies de déploiement . . . . .	66
3. 8. 1	Degré des nœuds . . . . .	66
3. 8. 2	Partitionnement . . . . .	68
3. 8. 3	Largeur des composantes connexes . . . . .	68
3. 9	Outils développés . . . . .	70
3. 10	Conclusion . . . . .	71
<b>4</b>	<b>Contrôle des feux de circulation distribué sur plusieurs intersections</b>	<b>73</b>
4. 1	Algorithmes de contrôle des feux de circulation . . . . .	74
4. 2	TAPIOCA : organisation du réseau . . . . .	76
4. 2. 1	Architecture . . . . .	77
4. 2. 2	Distance entre les nœuds $SN$ et $DN$ . . . . .	78
4. 3	TAPIOCA : le cas d'une intersection isolée . . . . .	78
4. 3. 1	Classement de mouvements sur la base de deux objectifs . . . . .	79
4. 3. 2	Sélection d'une phase . . . . .	81
4. 3. 3	Politique avec des mouvements faiblement en conflit . . . . .	83
4. 3. 4	Passages piétons . . . . .	83
4. 3. 5	Durée de phase . . . . .	83
4. 3. 6	Évaluation sur une intersection isolée (Amiens) . . . . .	85
4. 4	TAPIOCA : le cas de plusieurs intersections . . . . .	87
4. 4. 1	Cas 1 : amélioration du score . . . . .	87
4. 4. 2	Cas 2 : synchronisation indépendante . . . . .	87
4. 5	TAPIOCA : algorithme détaillé . . . . .	88
4. 6	Simulations . . . . .	91
4. 6. 1	Scénario 1 : ville d'Amiens . . . . .	91
4. 6. 2	Scénario 2 : TAPASCologne . . . . .	93

4. 6. 3	Scénario 3 : grilles d'intersections . . . . .	93
4. 7	Conclusion . . . . .	96
<b>5</b>	<b>Des réseaux de capteurs sans fil au service des infrastructures routières</b>	<b>99</b>
5. 1	Délais et problèmes de transmission . . . . .	99
5. 2	Prédiction des données perdues . . . . .	100
5. 2. 1	Prédiction sur le nœud <i>DN</i> . . . . .	100
5. 2. 2	Prédiction sur le nœud décision . . . . .	101
5. 3	Gestion de plusieurs intersections . . . . .	101
5. 4	Inventaire des messages . . . . .	102
5. 5	Simulations . . . . .	104
5. 5. 1	Description des scénarios . . . . .	104
5. 5. 2	Evaluation des problèmes réseaux . . . . .	104
5. 5. 3	Effet du mécanisme de prédiction . . . . .	105
5. 6	Conclusion . . . . .	106
<b>6</b>	<b>Caractérisation et gestion des embouteillages</b>	<b>111</b>
6. 1	État de l'art des systèmes embouteillés . . . . .	111
6. 1. 1	États de trafic et formation des embouteillages . . . . .	111
6. 1. 2	Contrôle des feux de circulation dans des conditions saturées	114
6. 1. 3	Limites de la littérature . . . . .	116
6. 2	<i>BASALTE : a distriButed and Adaptive Signalization ALgorithm for saTurated intErsections</i> . . . . .	117
6. 3	Architecture . . . . .	118
6. 4	Algorithme de contrôle des feux de circulation . . . . .	119
6. 4. 1	Étape 1 : métriques . . . . .	119
6. 4. 2	Étape 2 : agrégation et prise de décision distribuée . . . . .	122
6. 4. 3	Étape 3 : mise en place des feux de circulation . . . . .	124
6. 4. 4	Résumé des messages . . . . .	124
6. 5	Simulations . . . . .	125
6. 6	Perspectives . . . . .	125
6. 7	Conclusion . . . . .	128
<b>7</b>	<b>Conclusion générale et perspectives</b>	<b>129</b>
<b>A</b>	<b>Principales notations</b>	<b>133</b>
<b>B</b>	<b>Liste des sigles</b>	<b>135</b>
<b>C</b>	<b>Liste des principaux outils et simulateurs</b>	<b>137</b>
<b>D</b>	<b>Liste des publications</b>	<b>139</b>
	<b>Bibliographie</b>	<b>143</b>



---

## Liste des figures

---

1.1	Évolutions historiques et projetées de la population mondiale de véhicules motorisés [SG09]. 1950 – 2030. . . . .	2
1.2	Coûts annuels liés au trafic routier aux États-Unis, par automobiliste [Lit13]. . . . .	2
1.3	Temps moyen annuel passé par un automobiliste dans les embouteillages, dans les 10 villes considérées comme étant les plus embouteillées par l’institut Inrix. . . . .	4
2.1	Exemple d’une grille d’intersections. . . . .	10
2.2	Les quatre classes de modèles du trafic routier. De gauche à droite : macroscopique, mesoscopique, microscopique et sous-microscopique. . . . .	11
2.3	Diagrammes débit-densité. . . . .	13
2.4	Modèle de carrefour généralement utilisé dans la littérature. . . . .	14
2.5	Exemple de matrice des conflits. . . . .	16
2.6	NEMA - Diagramme de phases. . . . .	18
2.7	NEMA - Diagramme en anneaux. . . . .	18
2.8	Temps inter-véhiculaire, séparant deux véhicules successifs. . . . .	19
2.9	Évolution du temps inter-véhiculaire à la sélection d’une file. . . . .	19
2.10	Illustration des différents temps de feu vert pour une phase. . . . .	21
2.11	Paramètres et métriques généraux des STI. . . . .	24
2.12	Exemple d’utilisation de CAUTOVE. À gauche, nous définissons les voies à prendre en compte. À droite, une détection de véhicule, avec ce que perçoit le logiciel. . . . .	29
2.13	Détection d’un véhicule par un capteur. Les mesures sont des exemples issus de [Che+05]. Le premier graphique affiche, en fonction du temps, les variations brutes recueillies par un magnétomètre échantillonné à 128 Hz au passage d’un véhicule. Le deuxième graphique retranscrit, en fonction du temps, l’enregistrement d’un signal de détection : ce dernier apparaît lorsque 10 échantillons successifs sont enregistrés et dépassent un certain seuil. . . . .	30
2.14	Les systèmes interactifs. . . . .	32
3.1	Intersection à quatre directions où toutes les voies sont équipées de capteurs. . . . .	39
3.2	Deux autres exemples d’architecture. . . . .	40

3.3	Stratégies de déploiement de capteurs illustrés sur une intersection entre une route majeure (4 voies), une rue mineure (1 voie, 1 direction) et l'intersection elle-même. . . . .	43
3.4	Cas de figures où les capteurs ne sont pas intégrés sur les cartes. Le premier cas résulte d'une longueur de voie trop courte. Le deuxième cas de l'absence de plus de deux directions. . . . .	44
3.5	Évolution du nombre de capteurs en fonction de la distance minimale séparant deux intersections. . . . .	44
3.6	Comparaison du nombre de capteurs déployés en suivant différentes stratégies, sur six scénarios représentatifs de notre jeu de données. . . . .	44
3.7	Pondération des arêtes en fonction de la distance entre les nœuds. . . . .	46
3.8	Différence entre les graphes que nous générons et les graphes intersections-rues. . . . .	46
3.9	Caractérisation du jeu de données. . . . .	49
3.10	Distribution des distances entre les intersections sur des villes représentatives. . . . .	50
3.11	Les six cartes représentatives de notre jeu de données, détaillées dans ce chapitre. . . . .	52
3.12	Degré moyen sur notre jeu de données complet. . . . .	53
3.13	Degré moyen sur les six cartes. . . . .	53
3.14	Distribution des degrés pour le scénario de Paris (échelle log-log). . . . .	53
3.15	Distributions des degrés empiriques et ajustés. . . . .	54
3.16	Diagramme quantile-quantile de la distribution des degrés contre les distributions Gamma. . . . .	55
3.17	Coefficient de clustering. . . . .	56
3.18	Un réseau de quatre intersections gérées par un capteur par voie. . . . .	57
3.19	Nombre de composantes connexes. . . . .	58
3.20	Pourcentage de nœuds isolés. . . . .	59
3.21	CDF de la distance entre les composantes connexes proches. . . . .	59
3.22	Représentation d'un diagramme de Voronoï et d'une triangulation de Delaunay. . . . .	60
3.23	Diamètre des composantes connexes. . . . .	61
3.24	Analyse de la composante maximale. . . . .	61
3.25	Nombre de composantes connexes et biconnexes. . . . .	62
3.26	CDF du nombre de points d'articulation (pour chaque composante constituée d'au moins trois nœuds). . . . .	62
3.27	Paris. Distributions empiriques et ajustées lors de l'ajout de 4 relais (au plus) entre les composantes connexes. . . . .	64
3.28	Amélioration de la connectivité des réseaux. . . . .	65
3.29	Degré moyen sur notre jeu de données complet, pour un déploiement à un nœud par route. . . . .	66
3.30	Degré moyen sur notre jeu de données complet, pour un déploiement à un nœud par intersection. . . . .	67
3.31	Diagrammes quantile-quantile de la distribution des degrés contre les distributions Gamma. . . . .	67
3.32	Nombre de composantes connexes. . . . .	68

3.33	Coefficient de clustering moyen. . . . .	69
3.34	CDF de la distance séparant tous les couples de nœuds de la composante connexe maximale. . . . .	69
3.35	Aperçu de notre outil, développé sous licence LGPL. . . . .	70
4.1	Simulations extraites de [FCD12a] montrant l'évolution du temps d'attente en fonction de la pondération de deux critères : (file d'attente, famine). $S(y)$ représente une fonction de score équivalente à celle définie dans les sections suivantes. . . . .	75
4.2	Combinaisons sans conflits possibles pour un carrefour à 4 directions. . . . .	76
4.3	Une intersection à quatre directions et son architecture. . . . .	77
4.4	Exemple d'une matrice des conflits. . . . .	81
4.5	Exemple de phase avec deux mouvements en conflit léger. . . . .	82
4.6	Simulations sur une intersection isolée de la ville d'Amiens. . . . .	86
4.7	Simulations sur trois intersections adjacentes de la ville d'Amiens. . . . .	86
4.8	Exemple de scénario où des messages de synchronisation envoyés entre les intersections. . . . .	89
4.9	Diagramme de collaboration de TAPIOCA. . . . .	91
4.10	Comparaison de performances : plusieurs intersections de la ville d'Amiens. . . . .	92
4.11	Comparaison de performances : le scénario TAPASCologne. . . . .	94
4.12	Le scénario TAPASCologne. . . . .	95
4.13	Comparaison de performances : grilles d'intersections. . . . .	95
4.14	Simulation confrontant TAPIOCA (à gauche) à des plans de feux statiques (à droite). . . . .	96
5.1	Représentation des flux de communication et des principales procédures de notre algorithme. . . . .	102
5.2	Simulations sur une intersection isolée (1/2). . . . .	105
5.3	Simulations sur une intersection isolée (2/2). . . . .	107
5.4	Simulations sur dix intersections adjacentes avec $\lambda = \frac{1}{3}$ et $\Delta_{SN} = \Delta_{DN} = 8$ . . . . .	108
6.1	Les différents états de trafic. . . . .	113
6.2	Le réseau routier est bloqué, aucune place n'est libre entre les deux intersections. Ce cas de figure étend logiquement l'importance de l'embouteillage. . . . .	113
6.3	Présence d'un interblocage : les véhicules doivent faire marche arrière pour débloquer la situation. . . . .	114
6.4	Architecture de BASALTE. . . . .	118
6.5	Diagramme de collaboration de BASALTE pour le cas d'un réseau multi-sauts. . . . .	119
6.6	Résultats en fonction de la fréquence d'échantillonnage. . . . .	126
6.7	Performances du système de transport intelligent. . . . .	126
6.8	Exemple d'une synchronisation des feux verts. . . . .	127



---

## Liste des tableaux

---

2.1	Exemple d'orientation pour le choix d'un mode, en fonction de $V/C$ ( <i>espace occupé sur capacité</i> , la charge d'une intersection) [SBG98]. . .	26
3.1	Comparaison qualitative des architectures. . . . .	40
3.2	Paramètre des distributions Gamma. . . . .	54
3.3	Comparaison de quelques modèles de graphes aléatoires. . . . .	56
3.4	Paramètres des distributions gamma ajustées dans un déploiement avec amélioration de la connectivité. . . . .	64
5.1	Résumé des communications. . . . .	103
6.1	Principales métriques utilisées. . . . .	120
6.2	Messages transmis sur le réseau, fréquence et taille des paquets. . . .	124



# CHAPITRE 1

---

## Introduction

---

### 1.1 Un réseau complexe et critique

Le développement des infrastructures et des machines qui permettent notre déplacement, nos correspondances et notre approvisionnement a toujours été au cœur de nos préoccupations. Si l'une des premières routes aménagées de l'histoire fut construite il y a plus de 5 000 ans, l'un des premiers véhicules automobile, le fardier à vapeur, est lui bien plus récent. Il fut conçu par Joseph Cugnot (1725 – 1804) en 1770 et pouvait transporter une charge allant jusqu'à 5 tonnes. Il pouvait rouler à une vitesse maximale de 4 km/h, sur plus de quinze minutes. Les années passant, l'industrie automobile s'est développée au point de devenir ces dernières décennies un élément de société majeur. Le trafic routier – notamment urbain – est au cœur de nombreuses problématiques et est devenu un aspect essentiel de la vie quotidienne. Ce dernier s'est amplifié en l'espace de quelques années, engendrant de nombreux problèmes qui coûtent quotidiennement temps, argent, santé et qualité environnementale, que ce soit par l'intermédiaire des embouteillages, des accidents ou encore des infractions.

Le graphique 1.1 [SG09] nous montre, par exemple, que le nombre de véhicules motorisés est destiné à doubler d'ici la prochaine décennie, en dépassant les deux milliards. Le graphique 1.2 [Lit13] nous montre une estimation des coûts annuels liés au trafic routier aux États-Unis. Comme nous pouvons le constater, la plupart des coûts listés sont liés à des imprévus ou au facteur humain. Nous pouvons penser que les embouteillages, la pollution ou encore les accidents seraient, *a priori*, régulables si un meilleur contrôle leur était réservé. Ceci nous laisse supposer qu'une meilleure gestion des flux routiers nous permettrait d'améliorer l'expérience des usagers et de réduire les coûts globaux.

Dans cette thèse, nous nous intéressons à l'utilisation des technologies de l'information et de la communication afin d'améliorer la circulation du trafic routier dans un environnement urbain. En effet, le cadre urbain est celui qui engendre le plus de problèmes (ralentissements, engorgements, etc.), mais est aussi le plus complexe. En

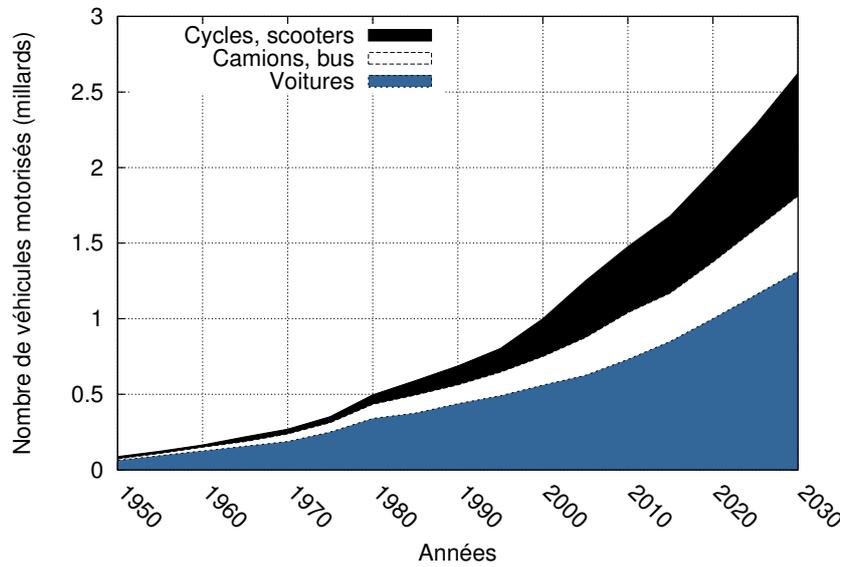


FIGURE 1.1 – Évolutions historiques et projetées de la population mondiale de véhicules motorisés [SG09]. 1950 – 2030.

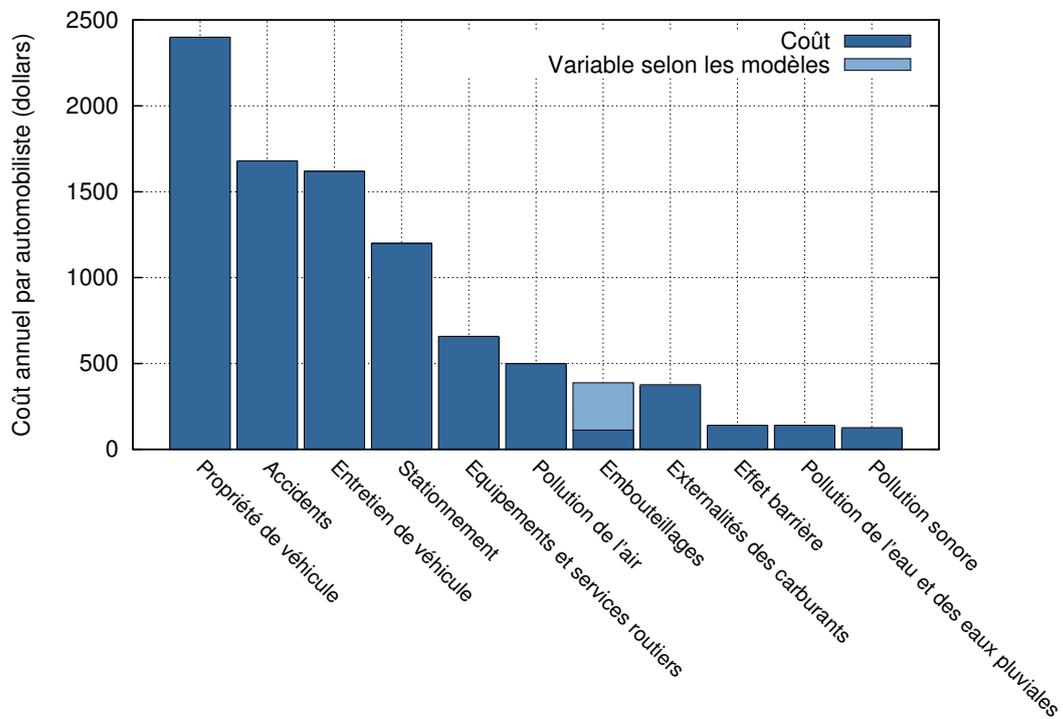


FIGURE 1.2 – Coûts annuels liés au trafic routier aux États-Unis, par automobiliste [Lit13].

outre, il offre la plus grande possibilité de surveillance et de contrôle par un système de transport intelligent.

## 1. 2 Contrôle et suivi des usagers à l'aide de systèmes de transport intelligents

Une étude menée par IBM en juin 2011 montre une évolution de 8% à 28%, entre 2010 et 2011, du nombre de New-Yorkais pour qui les transports auraient grandement nui à leur travail ou études [IBM11]. D'une part, ceci pourrait nous laisser supposer que les automobilistes sont de plus en plus exigeants. D'autre part, ceci pourrait également nous indiquer que les flux de circulation sont plus importants, ou que l'effort pour les absorber n'est plus suffisant. De plus, les barres en bleu foncé sur le graphique 1.3 montrent l'importance de la perte de temps engendrée par les embouteillages dans dix villes mondiales en 2010, selon Inrix<sup>1</sup>. Face à des chiffres qui impressionnent (p. ex., 94 heures sont perdues annuellement par automobiliste dans Bruxelles), il devient de plus en plus nécessaire de réagir rapidement à chaque situation de crise pouvant intervenir sur le réseau routier. Les ingénieurs l'ont bien compris, en développant des systèmes dits intelligents, qui sont devenus une nécessité ces dernières années. Les barres en bleu clair sur la figure 1.3 montrent une baisse significative de la perte de temps en 2013. Le nombre de véhicules en circulation allant en évoluant, nous pouvons conclure que cette baisse est *a priori* liée à de nouveaux aménagements ou à l'implantation de tels systèmes.

**Définition 1. Les systèmes de transport intelligents (STI)** apparaissent comme étant l'application des nouvelles technologies de l'information et de la communication au domaine des transports [Weba]. Ces nouveaux systèmes, apparus progressivement depuis les années 1960, se basent essentiellement sur des technologies permettant de détecter ou de visionner le comportement du trafic et de ses usagers. Ils permettent de réagir plus rapidement – parfois en temps-réel – aux variations du trafic routier.

En milieu urbain, les systèmes de transport intelligents s'étendent à de nombreuses applications. Une manière de les catégoriser serait de les répartir dans les domaines d'application qui suivent [KTYR] : gestion et contrôle du trafic, péages et tarification routière, sécurité routière et application de la loi, transports publics, informations et guides aux conducteurs, gestion du fret et des flottes, et enfin sécurité du véhicule. Ces systèmes agissent également de manière plus ou moins directe sur des enjeux modernes tels que la pollution, en permettant de réduire l'émission de gaz à effet de serre (conséquence d'une régulation cohérente du trafic). Le but visé par de tels systèmes est d'optimiser au mieux les coûts globaux liés au trafic routier et d'améliorer l'expérience de l'utilisateur sur des réseaux souvent complexes.

---

1. Société spécialisée dans l'analyse du trafic routier, collectant des données de tout pays et fournissant services, applications et outils pour faciliter la navigation. Les données utilisées dans cette thèse sont issues de : <http://scorecard.inrix.com/scorecard/>

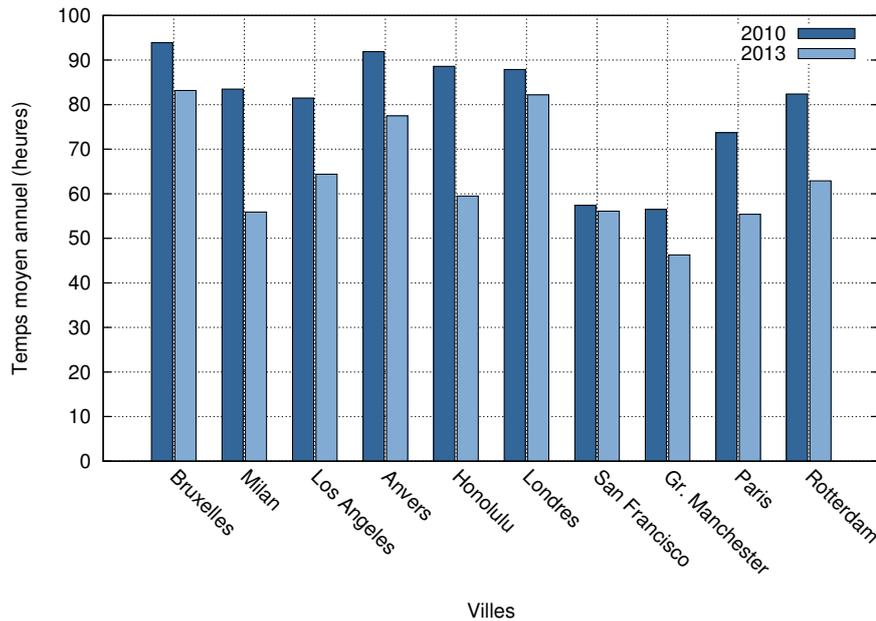


FIGURE 1.3 – Temps moyen annuel passé par un automobiliste dans les embouteillages, dans les 10 villes considérées comme étant les plus embouteillées par l’institut Inrix.

En ville, les feux de circulation sont des éléments clés qui peuvent contrôler le trafic routier et influencer, par exemple, la formation ou la résorption d’embouteillages. Pour ces raisons, nous choisissons, dans cette thèse, de présenter des applications essentiellement liées à la fluidification du trafic, notamment par l’intermédiaire de ce type de contrôle.

### 1.3 Le cas particulier des feux de circulation

L’intérêt des feux de circulation en milieu urbain n’est plus à démontrer. En premier lieu, ils servent à la sécurité des utilisateurs, sur des carrefours (ou intersections dans la suite du document) subissant une charge suffisamment importante pour être considérée dangereuse. Sans ces systèmes, si plusieurs flux de véhicules importants venaient à se croiser sur un intersection, le comportement individualiste des utilisateurs rendrait la traversée des carrefours périlleuse et augmenterait le temps perdu. La mise en place de séquences temporisées permettant de régir le passage des véhicules a donc été une nécessité très tôt dans l’histoire de la circulation routière. Le premier feu de circulation aurait ainsi été installé à Londres le 10 décembre 1868, sous la forme d’une lanterne à gaz pivotante aux couleurs rouge et verte. Pour fonctionner, il était nécessaire qu’un agent de police soit présent pour la manœuvrer. En France, il aura fallu attendre 1920 pour que Léon Foenquinos (1889 – 1954) décrive les feux tricolores : *"on installera, aux angles des croisements de rues, des poteaux ayant trois mètres de hauteur, sur lesquels seront fixés des signaux électriques lumineux et sonores (...)"*. Aujourd’hui, et dans de nombreux pays, il est courant de voir des feux de circulation dont les ampoules sont remplacées par des diodes électrolu-

minuscules et qui affichent le décompte des secondes restant avant un changement d'état.

L'application des nouvelles technologies de l'information et de la communication à ce type d'équipement représente une réelle opportunité. Les séquences habituellement utilisées ne sont pas toujours en adéquation avec la nature temps-réel du trafic routier. De plus, la présence de plusieurs feux de circulation successifs peut rapidement devenir inefficace si l'automobiliste doit s'y arrêter à chaque fois. Comme le montre [Mac+08], la gestion intelligente des feux de circulation est au cœur de nombreuses problématiques STI et est capable de fluidifier le trafic routier, en plus de servir indirectement l'environnement.

## 1. 4 De nouvelles problématiques

Qu'il s'agisse de routes, d'intersections, de feux de circulation ou de simples places de stationnement, l'infrastructure routière urbaine est aujourd'hui devenue comparable à un véritable réseau de communication. L'utilisation de systèmes de transports intelligents pour gérer cette infrastructure rend ce parallèle d'autant plus intéressant. Ainsi, si rendre l'infrastructure routière intelligente est un objectif développé depuis bien des années, il nous faut aujourd'hui intégrer les problématiques propres aux liaisons entre les différents équipements.

Actuellement, en France, les réseaux routiers urbains sont en majorité gérés par des *centres d'ingénierie et de gestion du trafic* (CIGT). Le rôle de tels organismes, généralement responsables d'une zone géographique bien particulière, est de coordonner au mieux les différents éléments routiers, et de faire face aux situations quotidiennes (travaux, accidents, gestion du trafic routier et des pics d'affluence, etc.). Ces centres, instaurés à la fin des années 1990 en France, possèdent généralement un poste de contrôle (PC) permettant l'exploitation d'un ensemble de technologies placées sur le terrain. Ces systèmes sont au centre d'une infrastructure urbaine. Citons par exemple le *PC Lutèce* situé au cœur de Paris, qui est relié à plus de 1 800 feux de circulation. Les missions des CIGT sont multiples : recueillir les données en provenance des véhicules, usagers et routes afin de remplir un rôle de superviseur et d'agir en cas de problème, gérer le trafic en cas d'imprévu ou encore informer les usagers. Remarquons que dans les pays anglo-saxons, la gestion du trafic est généralement divisée en plusieurs domaines qui sont gérés par des organismes tels que l'*Institute of Transportation Engineers* en charge des aspects suivants : caractéristiques du trafic, planification des transports, conception des infrastructures, contrôle, et maintenance organisationnelle, administrative et matérielle [HKD92 ; MW90].

Le développement des villes intelligentes nous impose toutefois d'acquérir des informations en temps-réel sur l'infrastructure routière. Nous sommes en droit de nous poser deux questions fondamentales. Tout d'abord, cette volonté de centralisation est-elle un frein pour le développement des transports intelligents ? Ensuite, quels mécanismes pouvons-nous proposer afin de rendre certaines composantes du système autonomes ? Nous avons l'intuition que tirer avantage de certains systèmes informa-

tiques, tels que les systèmes distribués, pourrait amener un élément de réponse à ces questions.

## 1.5 Diviser pour mieux régner

Dans un système centralisé, la plupart des informations sont localisées et gérées par la même entité. Dans cette thèse, nous nous intéressons au cas des systèmes distribués, dont le développement depuis plusieurs années – particulièrement depuis les années 2000 – représente des enjeux intéressants pour la gestion du trafic routier. Le principe est simple : revoir le rôle d’une entité centrale au profit d’une coopération entre plusieurs entités rattachées à l’infrastructure routière. Cette coopération permettrait de surveiller le trafic et de résoudre des problèmes localement, sans nécessiter, *a priori*, de communications régulières avec un centre de contrôle. Les intérêts peuvent sembler évidents : distribution des coûts de calcul, de communication et de décision, tolérance aux pannes ou encore autonomie.

Les éléments des systèmes distribués, répartis sur le terrain, sont des machines indépendantes connectées en réseau et communiquant par son intermédiaire. Dans un cas de figure général, nous pouvons imaginer que ces dispositifs sont interconnectés par un réseau filaire. Toutefois, il nous semble plus vraisemblable d’utiliser une connexion sans fil. Le temps de déploiement, ainsi que les coûts de mise en place, seraient potentiellement réduits. Sur le terrain, les réseaux de capteurs sans fil semblent pouvoir répondre à ces besoins. De plus, les éléments de ce type de réseau possèdent généralement des capacités en termes de calcul et de mémoire, leur permettant d’exécuter des procédures simples. Ces capacités sont généralement limitées mais ceci est au profit de leur prix, de leur fabrication et d’un déploiement rapide et massif. Enfin, leur intégration à un réseau urbain aurait pour conséquence de les inscrire dans des axes appartenant à l’Internet des objets.

Cette thèse tente de répondre à deux principales questions. Tout d’abord, la problématique générale de la thèse est la suivante : **que peut apporter le contrôle adaptatif et distribué d’une population de véhicules par une infrastructure fixe ?** En effet, si sur le papier ce type de système semble attrayant et prometteur, nous devons définir ce qu’il est possible de faire avec, et si ils sont capables de concurrencer des systèmes plus traditionnels. En voyant plus loin, il nous faut également définir **de quelle manière nous pouvons mettre en œuvre ce type de système**. En d’autres termes, étudier des modèles et des algorithmes exploitant ces systèmes, ainsi que leur déploiement.

## 1.6 Contribution

Les contributions réalisées dans le cadre de cette thèse sont détaillées au travers de cinq chapitres.

Dans le chapitre 2, nous présentons, tout d’abord, un état de l’art des systèmes de transport intelligents, afin de mieux comprendre l’existant, ses problématiques, et

le contexte dans lequel se déroule cette thèse [Fay12]. Dans le chapitre 3, nous étudions le déploiement de réseaux de capteurs sans fil à l'échelle d'une ville entière. En particulier, nous caractérisons la topologie des réseaux créés par ce type de déploiement [FC14a]. Dans le chapitre 4, nous proposons et étudions un algorithme de gestion des feux de circulation sur plusieurs intersections, baptisé TAPIOCA [FCD12a; FCD12b; FCD14a]. Dans le chapitre 5, nous étudions l'impact des communications sans fil et des pertes de paquets sur TAPIOCA et proposons notamment un mécanisme d'interpolation [FCD13a]. Dans le chapitre 6, nous étudions le cas des embouteillages, dont les propriétés diffèrent fondamentalement des situations plus générales pour lesquelles TAPIOCA est conçu. Nous proposons ainsi un algorithme répondant à ce scénario, nommé BASALTE. Nous concluons au chapitre 7.

Soulignons que nos contributions incluent le développement de deux plateformes de visualisation et de génération de résultats<sup>2 3</sup>. La première plateforme est mise en place sur deux sites [Fay13; Fay14b]. Elle présente l'intégralité des résultats abordés aux chapitres 4, 5 et 6. La deuxième plateforme [Fay14a] offre un outil d'analyse topologique, développé sous licence LGPL et permettant de reproduire les résultats présentés au chapitre 3.

Enfin, précisons que les principales notations utilisées dans ce document sont schématisées et résumées en annexe A, les sigles en annexe B et les principaux outils et simulateurs en annexe C. Une liste des publications réalisées, dans le cadre de cette thèse ou non, est donnée en annexe D.

---

2. <http://tapioca.sfaye.com/>, <http://basalte.sfaye.com/>  
3. <http://g.sfaye.com/>



## CHAPITRE 2

---

### Des systèmes de transport intelligents pour gérer le trafic routier

---

Dans ce chapitre, nous présentons un état de l'art de la gestion du trafic routier à l'aide de systèmes de transport intelligents. Nous axons volontairement cette étude sur la gestion des feux de circulation puis, sur des aspects touchant aux systèmes distribués. Les classes de modèles de trafic existant sont tout d'abord décrites en section 2.1. Puis, nous détaillons le fonctionnement d'une intersection à feux de circulation en 2.2 et définissons le modèle classique ainsi que les métriques et paramètres qui sont traditionnellement utilisés (p. ex., le temps de feu vert). Ensuite, dans la section 2.3, nous caractérisons les systèmes de transport intelligents et leur application aux systèmes de gestion des feux de circulation. Les équipements permettant la détection des véhicules – entre autres – y sont également abordés. Enfin, nous terminons avec l'étude des réseaux de communication en section 2.4 et le cas des systèmes distribués, avec l'exemple de réseaux de capteurs sans fil, en section 2.5.

#### 2.1 Modélisation du trafic routier urbain

Un réseau routier est conçu afin de permettre à ses usagers de se déplacer d'un point à un autre. Dans ce contexte, il est composé d'un ensemble de routes avec plus ou moins de voies, en fonction des zones d'activités et des lieux d'habitation. Les croisements entre les routes étant inévitables en milieu urbain, de par leur concentration, des intersections permettent de gérer les flux de véhicules entrant en conflit, pour leur sécurité et afin d'éviter les interblocages et de permettre aux automobilistes de changer de route. Ces intersections, ou dans certains cas des carrefours giratoires, régulent le trafic et appliquent généralement des règles de priorité à droite ou sont équipés de feux de circulation pour gérer les situations particulièrement dangereuses. Le réseau routier urbain peut donc être vu comme étant une collection d'intersections, liées les unes aux autres par des routes, auxquelles peuvent venir se greffer des éléments opérationnels (panneaux, feux de circulation, etc.).

Sur un réseau de plusieurs intersections, les axes sont généralement classifiés en *artères* principales ou secondaires, en fonction de leur utilisation et de leur capacité. À titre d'exemple, une grille d'intersections est représentée sur la figure 2.1. L'intersection centrale est située sur deux axes (artères). Les numéros indiquent le nombre de sauts qui séparent cette intersection des autres intersections. Ainsi, une ligne droite formée de plusieurs intersections consécutives et accueillant une majorité du trafic disponible est désignée comme une artère principale, tandis que les autres routes, accessibles sur chacune des intersections, sont désignées comme étant secondaires. Cette différenciation est utile pour la coordination des feux de circulation, qui s'effectuent dans la majorité des systèmes traditionnels sur les artères principales [US 08]. Elle est également utile à la compréhension de la formation des embouteillages, qui ont tendance à se former là où le trafic est plus fort. À ce stade, il n'est toutefois pas utile de décrire ces phénomènes, nous les introduisons toutefois dans le chapitre 6.

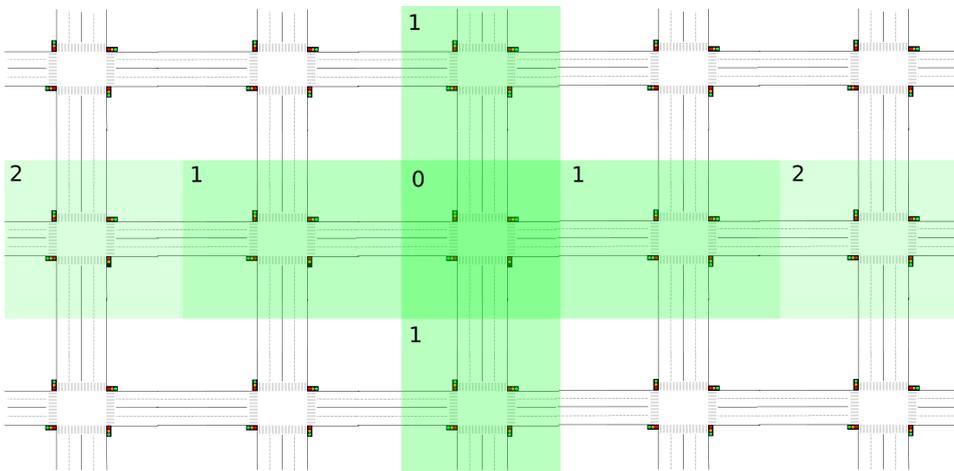


FIGURE 2.1 – Exemple d'une grille d'intersections.

Décrire le trafic routier existant sur ce type de réseau est crucial. Non seulement, ce type d'étude permet de planifier les évolutions à entreprendre sur le réseau routier (p. ex., travaux), mais elle permet également – de nos jours – de réaliser des simulations numériques permettant d'anticiper certaines difficultés, telles que les embouteillages ou accidents.

Le trafic routier a ainsi été décrit au travers de nombreux modèles [Dim12], notamment à partir des années 1930 où Greenshields et *al* [GCM+35] se sont intéressés à étudier les termes de densité (de véhicules), de capacité (des voies – routes) et de vitesse. La recherche de nouveaux modèles s'est ensuite accélérée dans les années 1950. Greenshields et *al* se sont intéressés aux performances des intersections routières [GSE46]. Chandler et *al* [CHM58] ont étudié le suivi de véhicules et les comportements que les conducteurs peuvent avoir les uns par rapport aux autres. Lighthill et *al* [LW55] ont, pour leur part, étudié des notions de débit dès les années 1955, tandis que des travaux se basant sur la théorie des files d'attente apparaissent [War52].

Ces travaux ont, entre autres, défini des modèles et des notions fondamentales de la gestion du trafic routier. Seulement, depuis leur publication, les conditions de circulation ont grandement évolué, de même que les technologies et les théories touchant aux mathématiques ou aux systèmes informatiques. Aujourd’hui, de nombreuses modélisations du trafic routier sont nées, se basant sur d’anciens modèles ou profitant de cette ascension. Pour cette raison, ces modèles sont généralement classifiés en deux principales familles [HB01], que nous décrivons ci-après. Tout d’abord, les modèles *microscopiques*, qui décrivent les véhicules individuellement. Ensuite, les modèles *macroscopiques*, qui décrivent la circulation à l’échelle d’un ensemble de véhicules sur une plus grande longueur de route.

Notons que deux autres classes de modélisations existent, mais sont moins courantes que les modèles microscopiques ou macroscopiques. D’une part, les modèles *mesoscopiques* sont une classe de modèles intermédiaire [PMN97 ; HB01]. Ils permettent de modéliser les véhicules par paquets, qui sont représentés par des pelotons pouvant s’échanger des véhicules. D’autre part, les modèles *sous-microscopiques* sont à un niveau en dessous des modèles microscopiques. Ils considèrent que chaque véhicule peut se diviser en plusieurs sous-structures (p. ex., modélisation de la vitesse de rotation du moteur par rapport à la vitesse du véhicule). Ces quatre classes de modèles sont représentés par la figure 2.2.

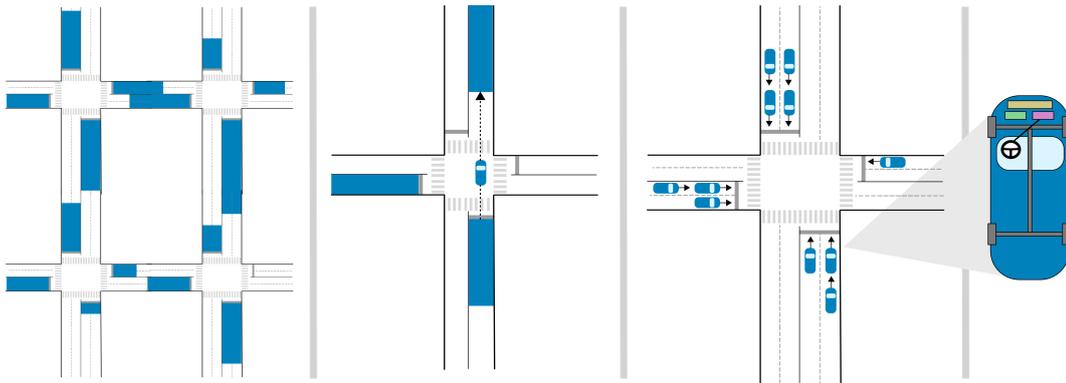


FIGURE 2.2 – Les quatre classes de modèles du trafic routier. De gauche à droite : macroscopique, mesoscopique, microscopique et sous-microscopique.

### 2. 1. 1 Modèles microscopiques

Les modèles microscopiques permettent de décrire le comportement individuel de chaque véhicule et conducteur sur le réseau routier. Beaucoup de paramètres sont généralement à prendre en compte, mais ils permettent de réaliser des simulations numériques réalistes (p. ex., simulateur SUMO [BBEK11 ; KEBB12]). Dans ces modèles, le comportement d’un véhicule est perçu de façon individuelle par ses voisins.

La physique statistique permet d’étudier le comportement de systèmes physiques comprenant un nombre élevé de particules. Autrement dit, il est possible de décrire un système macroscopique à partir des caractéristiques de ses éléments, qui sont microscopiques. Le rapprochement avec le trafic routier urbain est évident : les véhicules sont les éléments mobiles et sont donc les particules du système. Les interactions

entre les particules sont déterminées par la façon dont les véhicules influencent le mouvement des uns et des autres [CSS00]. L'un des modèles suivant ce principe et représentatif des modèles microscopiques est le modèle de suivi de véhicules (*car following model* [CHM58]), qui décrit le comportement d'un véhicule en suivant un autre.

Les modèles à base d'automates cellulaires sont apparus plus récemment et sont également adaptés à la modélisation microscopique du trafic routier [NS92]. Ils sont composés d'une grille régulière de cellules pouvant chacune posséder un état, choisi parmi un ensemble fini et qui peut évoluer au cours du temps. L'état d'une cellule à un instant est fonction de l'état d'un nombre fini de cellules voisines à l'instant précédent. À chaque unité de temps, les règles sont appliquées sur chacune des cellules de la grille, produisant ainsi de nouvelles générations de cellules héritant directement de la génération précédente. Nagel et *al* [NS92] ont proposé au début des années 1990 un modèle basé sur des automates cellulaires permettant de représenter l'évolution du trafic sur une autoroute. Chaque cellule représente une portion de l'autoroute, pouvant être occupée ou non par un véhicule roulant à une vitesse variable. Chaque véhicule avance donc à chaque changement d'état avec une distance proportionnelle à sa vitesse. D'autres modèles considèrent le temps de réaction des conducteurs, rendant les états métastables [BSS98]. Ces temps de réactions sont intégrés en faisant dépendre une probabilité de freinage à la vitesse du véhicule. Certains modèles permettent aux véhicules d'anticiper la vitesse des véhicules qui les précèdent [KSS00]. Notons que les automates cellulaires trouvent également utilité dans les applications liées au trafic, comme la gestion des feux de circulation, où nous pouvons imaginer dédier une cellule à une intersection qui se coordonnerait par rapport à ses voisines.

## 2. 1. 2 Modèles macroscopiques

Les modèles macroscopiques servent à la théorie analytique et permettent de décrire le comportement de flux de véhicules, sans distinguer les véhicules individuellement. Le trafic est ainsi représenté de manière agrégée. Ces modèles sont parmi les plus anciens et considèrent le trafic comme étant un phénomène continu [Whi11 ; LW55 ; GCM+35]. Ils sont généralement dérivés de la mécanique des fluides, plus particulièrement de la dynamique des fluides (c.-à-d., fluides en mouvement). L'exemple le plus parlant est celui des autoroutes homogènes, où aucune distinction n'est opérée sur les véhicules. Un autre exemple d'application est le cas des carrefours giratoires. [LL11] décrit par exemple les flux de circulation comme étant des fluides en opposition. Le constat établi est que généralement, le plus gros flux (en termes de véhicules) prend le dessus (et donc la priorité) sur le flux de plus faible importance. Ces modèles sont toutefois plutôt utilisés pour des études de densité de trafic et de capacité.

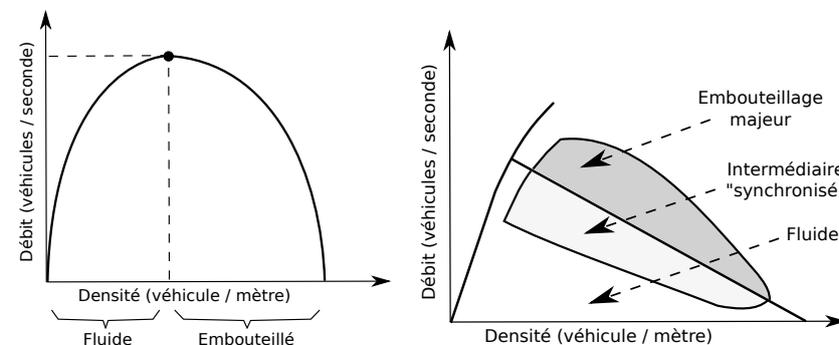
La théorie des files d'attente rentre dans la catégorie des modèles macroscopiques et se veut bien adaptée au cas du trafic routier [War52 ; Akç97 ; YAKS10]. Cette théorie appartient au domaine des probabilités et permet une description détaillée des files d'attente de véhicules, réparties sur un ensemble de voies. Dans le cas des intersections, une file d'attente se crée lorsque le nombre d'automobilistes (clients) dépasse un certain seuil ou lorsque, par exemple, un feu est au rouge (serveur). Il est

particulièrement facile avec cette théorie de calculer des valeurs telles que le nombre moyen de véhicules en attente, en service, le temps moyen d'attente ou encore de séjour dans le système. Ceci, en prenant généralement pour modèle une file M/M/1, où l'unique serveur est un feu de circulation. Notons par exemple  $\lambda$  la fréquence moyenne d'arrivée et  $\mu$  la fréquence moyenne de service. La charge du système est décrite par  $C = \lambda/\mu$ . Si  $C < 1$ , nous pouvons obtenir le temps moyen d'attente avec  $C/(\mu \cdot (1 - C))$ , le nombre moyen de véhicules en attente par  $C/(1 - C)$ , ou encore le temps moyen de séjour dans le système par  $(1/\mu) \cdot (1/(1 - C))$ .

Dans la suite de cette thèse, les algorithmes que nous définissons ont plutôt tendance à privilégier des modèles macroscopiques. Dans le cas des feux de circulation, nous pouvons par exemple considérer qu'un feu doit passer au vert si l'une de ses files contient un nombre trop important de véhicules. À l'inverse, les véhicules pourraient être considérés individuellement, comme par exemple lorsqu'il s'agit de détecter une fraude (p. ex., passage au feu rouge) ou un obstacle. Le principal simulateur utilisé dans cette thèse, SUMO [BBEK11 ; KEBB12], possède une vision microscopique du trafic routier, afin de modéliser les comportements individuels des automobilistes et de se rapprocher d'un cas de figure réel.

### 2. 1. 3 Le diagramme fondamental du trafic

Le diagramme fondamental du trafic est une alternative aux modélisations précédentes et permet de décrire la relation entre le débit et la densité d'un réseau routier [CSS00]. Il est observé théoriquement, et plusieurs variations sont identifiables expérimentalement [Ker00]. Il distingue deux états de trafic différents : le cas où le trafic est fluide (c.-à-d., vitesse proche de la vitesse libre) et le cas où il est embouteillé (c.-à-d., les véhicules se déplacent à une vitesse inférieure à la vitesse libre). La figure 2.3(a) représente la forme générale de ce diagramme. Des variantes existent, telles que celle suggérée par la théorie des trois phases du trafic, de Kerner [Ker00]. Elle considère un état supplémentaire d'embouteillage, comme représenté sur la figure 2.3(b).



(a) Diagramme fondamental du trafic. (b) Théorie des trois phases du trafic.

FIGURE 2.3 – Diagrammes débit-densité.

La modélisation du trafic routier peut suivre différentes théories. Toutefois, dans la plupart des cas, il est question de quantifier les flux de véhicules. En d'autres termes, le fonctionnement d'un système de transport dépend fortement de sa charge. Il y a une nécessité de contrôler – et donc de mesurer – les variations du trafic routier afin d'optimiser le fonctionnement du système de transport. Dans la partie suivante, nous étudions plus particulièrement le fonctionnement des feux de circulation, afin d'appuyer ces remarques. Nous étudions ensuite les systèmes intelligents mis en place de nos jours.

## 2.2 Le cas particulier des feux de circulation

### 2.2.1 Fonctionnement traditionnel d'une intersection

Le modèle d'intersection classique est représenté sur la figure 2.4. Il est composé d'un ensemble de quatre *directions* ou *routes* possibles notées  $D$  (p. ex.,  $E$  sur la figure 2.4). Ce modèle est celui utilisé dans la plupart des travaux de la littérature, mais rien n'interdit de l'étendre. Chaque direction peut être décomposée en une *direction entrante* et une *direction sortante*, ou bien des deux. Une direction entrante permet aux véhicules, répartis sur une ou plusieurs *voies*, de traverser l'intersection vers une direction sortante, qui fait le lien avec une intersection voisine. Sur chaque direction entrante, les véhicules tournant à gauche utilisent la voie la plus à gauche, tandis que la voie la plus à droite est généralement utilisée pour les véhicules allant tout droit ou tournant à droite. Nous nous plaçons dans un scénario où les conducteurs conduisent du côté droit de la route.

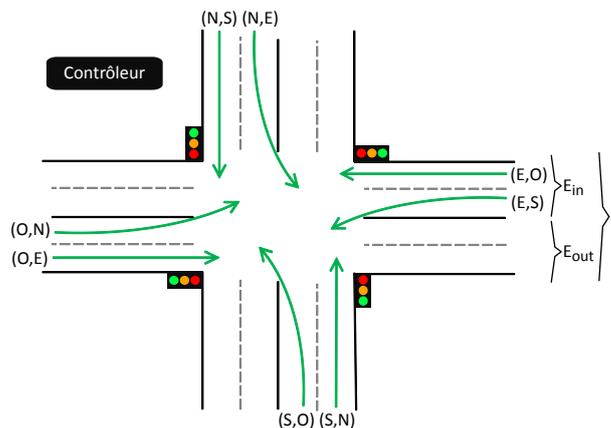


FIGURE 2.4 – Modèle de carrefour généralement utilisé dans la littérature.

Typiquement, un véhicule entre sur l'intersection par une direction  $a \in D$  et ressort par une direction  $b \in D$  (les cas où  $a = b$  sont possibles). Chaque possibilité d'entrée-sortie décrit par le couple  $(a, b)$  est un *mouvement* possible sur l'intersection. Sur la figure 2.4, le mouvement  $(E, S)$  est par exemple le mouvement allant de la direction  $E$  à la direction  $S$ . Un mouvement peut être accessible depuis plusieurs voies (p. ex., deux voies qui permettent d'aller tout droit). À l'inverse, une voie peut

proposer plusieurs mouvements possibles (p. ex., aller tout droit ou tourner à droite). Nous supposons ici qu'un feu de circulation contrôle un mouvement ou, au moins, une direction entrante. Enfin, notons que les mouvements allant à droite ne sont pas représentés car sont confondus avec les mouvements allant tout droit. Il paraît évident de dire que si un mouvement allant à droite rentre en conflit avec un autre mouvement, alors cela est nécessairement le cas pour le mouvement allant tout droit qui y est associé.

Afin de régir le fonctionnement des feux de circulation, des mécanismes combinatoires et temporels spécifiques sont utilisés. Nous les introduisons ci-après et fournissons l'annexe A en guise de support et d'illustration.

## 2. 2. 2 Gestion des séquences de feux

### 2. 2. 2. 1 Vocabulaire

**Définition 2.** Un **contrôleur de feux** est un équipement permettant de veiller au bon fonctionnement des feux de circulation. Il applique une stratégie de signalisation en enclenchant séquentiellement les feux. Il est traditionnellement rattaché à un seul carrefour mais peut, dans certains cas, en gérer plusieurs. Typiquement, un contrôleur possède les éléments principaux suivants [Min06] :

- Une *unité d'énergie*.
- Une *unité de contrôle*, donnant l'ordre d'enclenchement des feux.
- Une *unité d'avertissement rapide*, réagissant en cas d'erreur critique (p. ex., en mettant un feu orange clignotant sur l'ensemble des feux).
- Une *unité de gestion des conflits*, qui est programmée avec les combinaisons de feux verts autorisés et qui vérifie les données envoyées par l'unité de contrôle. Elle utilise généralement une *matrice de conflits* afin de savoir quels mouvements peuvent se dérouler simultanément. Différents types de conflits existent et peuvent être triés en fonction de leur gravité. Les conflits entre mouvements latéraux sont, par exemple, généralement interdits. A l'inverse, certaines configurations autorisent le déroulement de deux mouvements conflictuels, plus légers, simultanément (p. ex., mouvements tournant à gauche). La figure 2.5 donne un exemple d'une telle matrice, qui pourrait être utilisée sur l'intersection décrite en 2. 2. 1. Sur cet exemple, nous pouvons constater que le mouvement  $(N, S)$  peut avoir le feu vert avec le mouvement  $(S, N)$ , mais pas avec le mouvement  $(E, 0)$ , pour des raisons de sécurité. Cette unité fait appel à l'unité précédente en cas d'erreur ou de faute constatée sur l'un des feux.
- Une *unité d'administration*, pour prendre le contrôle du carrefour (p. ex., par la police, ingénieurs).
- Dans certains cas, une *unité de détection*, connectée à des *détecteurs* de véhicules.

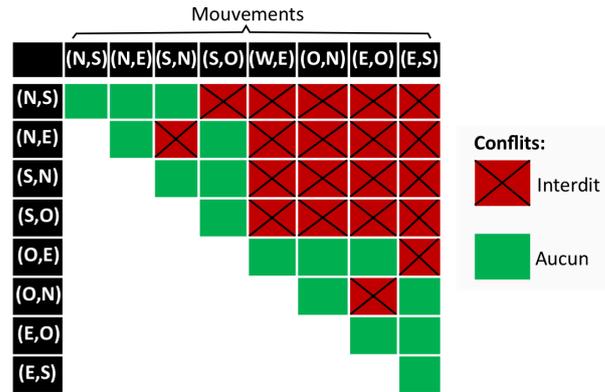


FIGURE 2.5 – Exemple de matrice des conflits.

**Définition 3.** Un **détecteur** est une technologie permettant de récolter et transmettre des informations environnementales à un nœud traitant.

**Définition 4.** Une **phase** est un intervalle de temps durant lequel une combinaison de feux verts autorisée par l'unité de gestion des conflits est activée. Les phases sont déterminées à partir des mouvements que chaque direction peut effectuer. Dans la mesure du possible, elles ne doivent pas mettre de mouvement en *conflict*, le principe même des feux de circulation étant d'assurer une certaine sécurité entre les flux d'usagers.

**Définition 5.** Un **cycle** correspond à l'enchaînement d'un ensemble de phases successives. Il déroule l'ensemble des phases et mouvements possibles, de manière à ce que toutes les voies aient au moins une fois le feu au vert. Lorsque l'intersection est équipée de dispositifs de mesure, cette règle n'est pas nécessairement appliquée (par exemple, les voies sans véhicules peuvent ne pas avoir le feu vert pendant un ou plusieurs cycles).

**Définition 6.** Un **plan de feux** correspond à la description d'un cycle donné, et définit les différentes phases à dérouler ainsi que leur durée. Généralement, un plan de feu est conçu pour une tranche horaire spécifique, en fonction du niveau de trafic moyen à absorber (p. ex., un plan de feu plus court pour la nuit, un autre plus long en heure de pointe). Exemple donné en annexe A où le plan de feux est constitué d'un cycle de trois principales phases.

### 2. 2. 2. 2 Création d'un plan de feux

Face à la multitude de constructeurs et de spécifications, plusieurs organisations se sont regroupées dans le but de mettre en place des nomenclatures, afin de favoriser le développement des STI et d'éviter les incompatibilités techniques et fonctionnelles. Aux États-Unis par exemple, dès le début des années 1990, la NEMA (*National Electrical Manufacturers Association*) s'est chargée de la rédaction de telles normes. Nous prenons ici l'exemple de cet organisme en détaillant la manière dont les plans

de feux peuvent être créés et la logique de fonctionnement standard des contrôleurs de feux, probablement l'une des plus répandues traditionnellement [Uni ; Gor+05 ; Nat98]. Notons que certains pays possèdent une logique similaire, comme le cas de la France avec la norme DIASER (*dialogue standard pour les équipements de régulation*).

Un *diagramme de phases* rassemble l'ensemble des phases d'un cycle, où chaque phase est représentée par un bloc. Afin de dérouler un cycle sans conflit entre les phases, il suffit de dérouler ce diagramme. La figure 2.6 décrit par exemple un diagramme de phases possédant trois phases que nous nommons  $\phi A$ ,  $\phi B$  et  $\phi C$ .

Un *diagramme en anneaux* permet de définir chacune des phases d'un diagramme de phases, en s'assurant de la bonne compatibilité entre les mouvements existant, comme représenté sur la figure 2.7. Les diagrammes en anneaux sont constitués de barrières qui isolent les mouvements appartenant aux rues en conflit latéral d'une intersection : les mouvements d'un côté de la barrière ne doivent pas se dérouler en même temps que les mouvements de l'autre côté. Ainsi, une phase représente un ensemble de mouvements sans barrières. La sélection des mouvements de chaque phase se fait à l'aide des anneaux présents de part et d'autre des blocs : chaque anneau sélectionne un mouvement et leur assemblage forme une phase. Sur une intersection à 4 directions, la barrière permet par exemple de séparer les mouvements de deux rues orthogonales. Selon le nombre de mouvements utilisés dans le diagramme en anneaux, plusieurs diagrammes de phases peuvent être créés. Notons qu'une matrice des conflits est un bon complément à ce type de diagramme (sec. 2. 2. 2. 1). Lorsque des phases sont appliquées, une matrice des conflits permet de vérifier, en toute circonstance, que leur composition respecte bien les règles de sécurité. En dehors du cadre de la NEMA, nous pouvons également imaginer composer des phases en se basant uniquement sur cette matrice.

Pour finir, soulignons que le nombre de phases dans un cycle est un élément important. Si il y a trop de phases, leur durée est plus courte ou le temps d'attente est plus long, ce qui peut être mal perçu par les usagers. Si le nombre de phases est moins important, chacune autorise plus de mouvements simultanés : l'intervalle entre deux sélections du même mouvement est donc réduit. Il y a un compromis entre la capacité, sécurité et attente. Selon [Gor+05], depuis plusieurs années déjà, le nombre de phases est limité à huit dans la plupart des contrôleurs de feux, afin de ne pas provoquer des arrêts inutiles ou un rallongement du temps de cycle. Il est toutefois courant pour des intersections d'avoir deux à quatre phases.

## 2. 2. 3 Gestion du temps de feux

### 2. 2. 3. 1 Temps inter-véhiculaire

Soit  $\tau^h$ , la moyenne du temps séparant le passage de deux véhicules successifs à un feu de circulation, comme représenté par la figure 2.8. Ce temps peut être utilisé, par exemple, pour estimer le temps que peut mettre une file d'attente à passer un feu vert. Dans des conditions ne prenant pas en compte les ralentissements liés au début ou à la fin d'une phase, ce temps est estimé à 2 secondes [Gor+05 ; US 08].

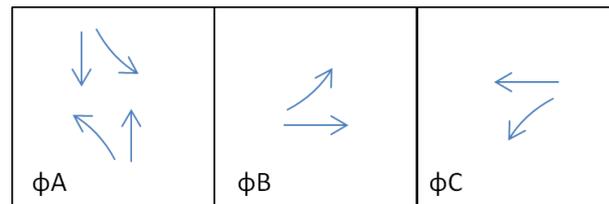


FIGURE 2.6 – NEMA - Diagramme de phases.

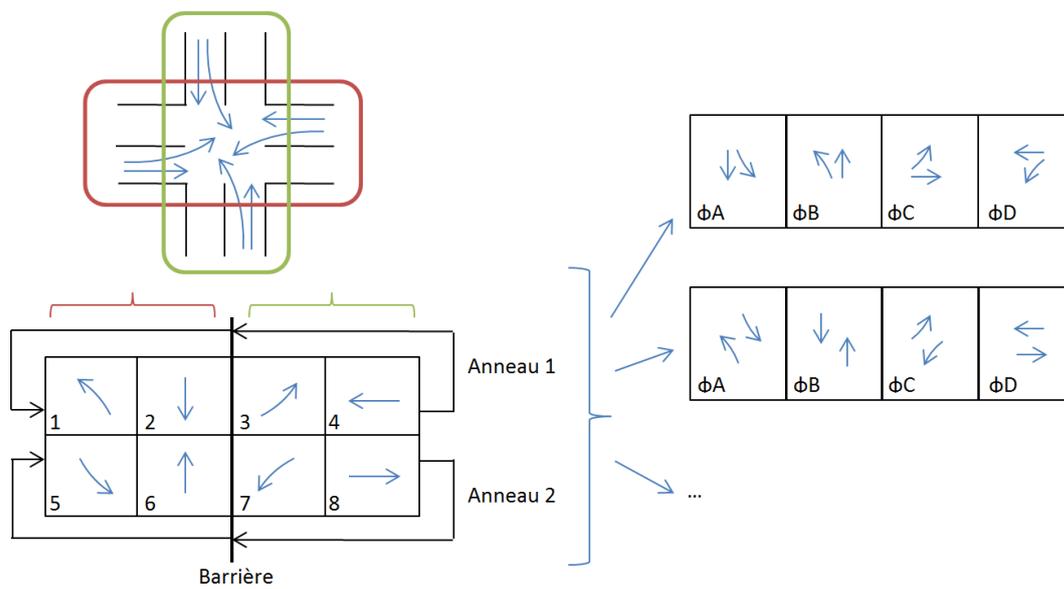


FIGURE 2.7 – NEMA - Diagramme en anneaux.

Notons que le temps inter-véhiculaire permet généralement de déterminer la capacité des routes.

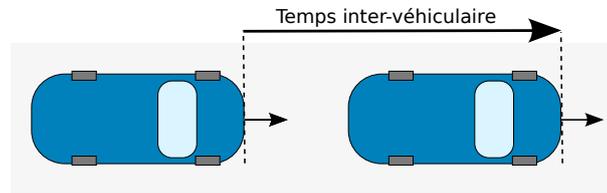


FIGURE 2.8 – Temps inter-véhiculaire, séparant deux véhicules successifs.

### 2. 2. 3. 2 Temps perdu

Le temps perdu  $\tau^{LT}$  est une mesure du temps non utilisé sur une phase, lorsqu'aucun feu n'est vert. Généralement, il est la somme de trois valeurs, estimées par [Gor+05 ; US 08 ; Min09]. 1) Le feu orange, marquant l'arrêt d'une phase, dure en moyenne 3 secondes en zone urbaine. 2) Le feu rouge de sécurité, qui correspond à un temps de dégagement séparant deux phases, varie selon la taille de l'intersection. 3) Le temps de démarrage au début d'une phase (c.-à-d., la somme du temps de réaction et du temps d'accélération). Ce dernier provoque en moyenne 4 secondes de délai, réparties généralement sur les quatre premiers véhicules de chaque file. Le graphique 2.9 représente le comportement du temps de démarrage, se stabilisant au fur et à mesure vers le temps inter-véhiculaire,  $\tau^h$ .

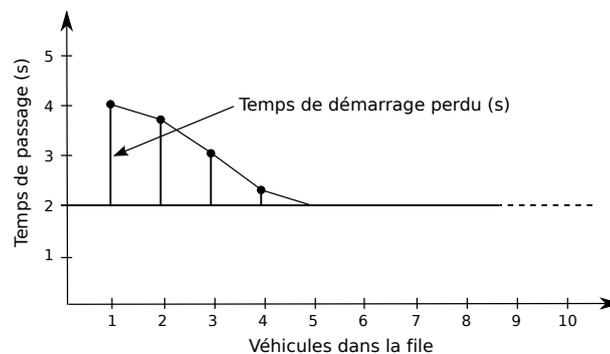


FIGURE 2.9 – Évolution du temps inter-véhiculaire à la sélection d'une file.

### 2. 2. 3. 3 Temps de cycle

La détermination d'un temps de cycle, noté par la suite  $\tau^c$ , a fait l'objet de nombreuses théories [Gor+05 ; US 08 ; CM00]. Il peut suivre divers objectifs, tels que la minimisation du temps d'attente ou la maximisation du débit. Les intersections possèdent généralement toutes un temps de cycle égal, qui sert de base à leur coordination car il représente une relation temporelle fixe. Un temps de cycle suffisamment large permet aux feux verts de chaque phase qu'il englobe d'être variables. La méthode classique de Webster [Web58] permet d'obtenir un temps de cycle optimal en

minimisant le retard subi par les véhicules, sur les  $p$  phases constituant un cycle :

$$\tau^c = \frac{(1,5 \cdot \tau^{LT} \cdot p + 5)}{1 - \sum_i y_i} \quad (2.1)$$

Notons  $V_i$ , le volume de la voie critique, comme étant le plus grand nombre de véhicules reçus par une voie lors de la phase  $i$ . Notons  $s = 3600/\tau^h$ , le débit de saturation, comme étant le nombre théorique de véhicules capables de traverser un feu qui reste au vert durant une heure. La charge de la phase  $i$ , noté  $y_i$  et utilisé dans la méthode de Webster, s'obtient en divisant  $V_i$  par  $s$ .

Le temps de cycle est typiquement fixé au minimum à 45 secondes et ne dure pas plus de 90 secondes pour éviter un temps d'attente trop long [Uni]. D'autres méthodes que celle de Webster existent et permettent d'obtenir des temps de feux minimisant, par exemple, le degré de saturation moyen [Yin08]. Certaines méthodes s'axent sur de lourds calculs prédéterminés afin d'obtenir un temps de cycle correspondant à un réseau d'intersections. D'autres fixent manuellement un nombre, comme par exemple à Los Angeles, où tous les feux de circulation sont à 60 secondes [US 08]. Enfin, de plus en plus de méthodes modernes partent du principe que la notion de cycle n'est pas nécessaire, et qu'un enchaînement dynamique des phases suffit à maintenir un système [US 08].

### 2. 2. 3. 4 Temps de feu vert

Un cycle se décompose en plusieurs phases, qui ont chacune une durée de feu vert qu'il est primordial de bien régler. En effet, une phase trop courte peut ne pas laisser suffisamment de temps à une file de se décharger. À l'inverse, une phase trop longue augmente le temps d'attente des automobilistes présents sur les autres phases, sans être utile.

Le temps de feu vert  $\tau_i^g$  pour une phase  $i$  est généralement calculé proportionnellement au temps de cycle, en fonction de différentes contraintes (p. ex., statistiques connues de débit ou d'attente).

Dans certains systèmes, ce temps est calculé dynamiquement et est borné entre un temps minimal  $\tau_i^{g^{min}}$  et un temps maximal  $\tau_i^{g^{max}}$  [Gor+05 ; US 08 ; AM97]. Une fois le temps de feu vert minimal écoulé, un temps additionnel  $\tau_i^{g^{add}}$  peut rallonger la phase, jusqu'à éventuellement atteindre  $\tau_i^{g^{max}}$ . À chaque nouveau véhicule qui arrive assez proche du précédent,  $\tau_i^g$  est incrémenté de quelques secondes (paramètre généralement proche de  $\tau^h$  secondes), laissant ainsi le temps au nouveau véhicule de passer. Ceci est illustré au travers de la figure 2.10.

La somme des durées de chaque phase, en tenant compte des temps intermédiaires, doit être égal au temps de cycle. Ceci est vérifié par la relation suivante :

$$\tau^c = \sum_i (\tau_i^{g^{min}} + \tau_i^{g^{add}} + \tau^{LT}) \quad (2.2)$$

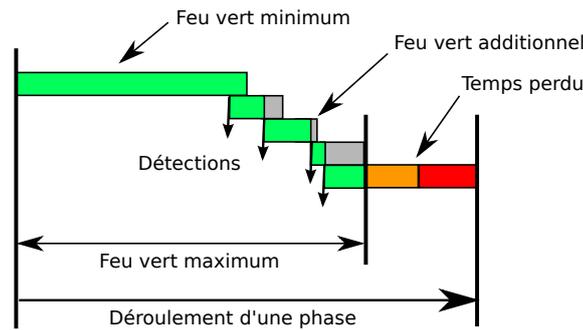


FIGURE 2.10 – Illustration des différents temps de feu vert pour une phase.

Notons que le temps de feu vert maximal a fait l'objet de nombreuses études. Kell et Fullerton [KF98] observent qu'il doit se situer entre 30 et 60 secondes. Orcutt [L.93] suggère que ce temps maximal doit être assez long pour laisser passer 1,3 fois la longueur moyenne de la file concernée. Courage [GZE89] indique qu'un temps de feu vert maximal élevé n'a que peu d'impact sur un système adaptatif (sec. 2. 3. 2) si le trafic est peu important.

### 2. 2. 3. 5 Coordination des feux entre intersections

*Coordonner* ou *synchroniser* les feux de plusieurs intersections consécutives revient à créer une *bande de progression*, *vague verte* ou *onde verte*, sur laquelle un flux de véhicule évolue sans être stoppé. Si le système ne possède pas de coordination, la progression n'est pas constante et les arrêts, les délais et la consommation d'essence ont tendance à augmenter [HHRC01 ; Nts]. De plus, à chaque fois que des voitures s'arrêtent, du temps de feu vert est perdu et le délai est augmenté.

Dans les systèmes traditionnels, la mise en place d'une coordination se fait dans la plupart des cas en fixant un temps de cycle identique pour chaque intersection d'une artère principale, voire en fixant des temps de phase identiques, mais décalés [US 08 ; AM97]. En clair, un système de signalisation est généralement synchronisé sur un seul axe. Pour chaque intersection, une coordination est appliquée à une seule phase par cycle, incluant les mouvements venant et menant à l'artère principale. Cette phase est généralement mise en dernière dans le cycle afin de pouvoir lui allouer éventuellement plus de temps que prévu, dans le cas où les phases non coordonnées viendraient à se terminer plus tôt (demande inférieure au temps prévu). Toutefois, des travaux récents, tels que ceux présentés dans cette thèse, considèrent que les intersections peuvent collaborer de proche en proche, et qu'une coordination de feux peut s'effectuer aussi bien sur les axes principaux que sur les autres routes (chapitre 4).

Les feux de circulation possèdent un *décalage* (*offset*) [US 08]. Le décalage est le temps – généralement fixe – séparant les phases coordonnées de deux intersections successives. Considérons deux intersections,  $I^1$  et  $I^2$ , liées par une route commune  $a$ , mesurant  $|a|$  mètres et dont les véhicules peuvent aller à une vitesse maximale de  $v$  km/h. Le décalage optimal est approximativement zéro lorsque  $|a|$  est trop petit [AM97] ou lorsque la file d'attente présente sur ce lien dépasse approximativement 22% de la capacité de stockage en longueur de ses voies (preuve ci-dessous). De plus,

dans un système congestionné, il est souvent considéré par la littérature que ce décalage peut être réglé à zéro [LRKS86 ; Rat88 ; AM97]. Sinon, il est égal au temps de traversée théorique entre les deux intersections :

$$\tau_a^{cross} = \frac{|a|}{v/3.6} \quad (2.3)$$

Si nous considérons le temps de démarrage de 4 secondes (sec. 2.2.3.2), un véhicule en tête de la file d'attente ayant le feu vert en  $I^1$  met au minimum  $\tau_a^{cross} + 4$  secondes pour aller rejoindre l'intersection voisine. A noter que  $\tau_a^{cross}$  peut être ajusté sur des bases réelles en installant des détecteurs au début et à la fin de la route.

**Preuve 1** Le temps théorique nécessaire à vider les  $n$  véhicules présents sur la route  $a$  est approximativement de  $n \cdot \tau^h$  secondes (en ne tenant pas compte du temps de démarrage). Le nombre de véhicules  $n$  peut être exprimé par rapport à la longueur de la route  $a$ , en posant  $n = |a|/(p \cdot L_{veh})$ .  $n$  représente alors le nombre de véhicules dont la longueur cumulée est  $p$  fois inférieure à la longueur de  $a$ . Nous cherchons la valeur de  $p$  pour laquelle le temps de traversée des  $n$  véhicules est égal au temps de traversée sans ralentissements sur la route  $a$  :

$$n \cdot \tau^h = \frac{|a|}{v} \Leftrightarrow \frac{|a|}{p \cdot L_{veh}} \cdot \tau^h = \frac{|a|}{v} \Leftrightarrow p = \frac{v \cdot \tau^h}{L_{veh}} \quad (2.4)$$

En prenant  $\tau^h = 2$  secondes,  $v = 13,9$  mètres/seconde et  $L_{veh} = 6$  mètres [US 08], nous obtenons  $p = 4,6$ . Cela signifie que si une file sur  $I^2$  fait plus de  $100/4,6 = 22\%$  de la taille de sa route, alors elle mettra davantage de temps à s'évacuer que les nouveaux véhicules de  $I^1$  mettront à arriver.

Au niveau d'une intersection, des mouvements combinés en phases forment des cycles qui se répètent en sélectionnant chaque mouvement au moins une fois. La composition des phases, leur ordonnancement et leur durée sont définis en fonction des objectifs d'écoulement du trafic et peuvent être dynamiques. Dans la section suivante, nous étudions les systèmes de transport intelligents actuellement mis en place en zone urbaine et qui permettent d'introduire ce caractère dynamique.

## 2.3 Contrôle du trafic routier urbain

Depuis les années 1960 et plus particulièrement au début des années 1970 [ACCJ06], les systèmes de régulation du trafic se sont rapidement développés afin d'améliorer les conditions de mobilité, en réduisant notamment les facteurs pouvant provoquer de l'attente et des embouteillages. L'apparition progressive de systèmes de gestion centralisés, en charge des carrefours d'une zone géographique donnée, s'est révélée déterminante. De plus, la recherche couverte par la presse montre un intérêt certain pour la gestion du trafic, ceci dans le but de supprimer l'instabilité dans les flux de trafic [Hor13b] et pour réduire le niveau d'embouteillage. Les systèmes de régulation

des accès (*ramp metering*), qui sont largement déployés, montrent qu’une gestion active des feux de circulation peut réduire drastiquement les embouteillages, même si des exemples en place, tels qu’à Minneapolis, sont contestés par les conducteurs qui ont le *sentiment* d’attendre plus longtemps.

### 2. 3. 1 Métriques et paramètres traditionnels

La mise en place d’un système de transport intelligent nécessite, *a priori*, de connaître à la fois les paramètres sur lesquels il peut jouer et les métriques à optimiser. D’une part, les paramètres sont des valeurs qui régissent le fonctionnement du système. Ils sont décrits en bleu sur la figure 2.11 et décrivent des valeurs ou comportements variables subis par le système, dans un cas général. Le style de conduite dépend par exemple en partie de l’utilisateur, mais a une influence majeure sur la manière dont peut réagir le système (p. ex., passer un feu rouge, provoquer un accident, accélération soudaine). D’autre part, les métriques sont des valeurs qu’il est possible de mesurer sur le terrain, découlant du comportement du réseau routier, de ses utilisateurs et de ses paramètres. Elles sont décrites en rouge sur la figure 2.11. Nous pouvons par exemple citer le temps moyen d’attente (*average waiting time – AWT*), qui est généralement représentatif des performances d’un système de gestion des feux de circulation.

Sur la figure 2.11, notons que les paramètres et métriques sont décomposés en deux échelles, l’une globale (ville, zone urbaine, système) et l’autre locale (voie, route, intersection). Cette distinction est importante dans la suite de cette thèse, les systèmes centralisés ayant un niveau de contrôle plutôt global, tandis que les systèmes distribués ont une vision locale. Nous remarquons qu’utiliser une vision locale plutôt que globale n’affecte que la manière dont un opérateur peut avoir accès aux informations. Il serait donc possible de s’y fier et d’imaginer des algorithmes utilisant ces paramètres et métriques. Chaque échelle peut également se décomposer en deux sous-ensembles : ce qui est en rapport avec l’utilisateur (p. ex., son comportement, le type de véhicule) et avec l’infrastructure (p. ex., feux de circulation). Cette distinction est également importante car dans le cas des paramètres, un STI subit généralement ceux en rapport avec l’utilisateur, tandis qu’il nous est possible de fixer ceux en rapport avec l’infrastructure.

### 2. 3. 2 Types de contrôles

Au fil des années, les systèmes de régulation se sont diversifiés, et ont connu globalement trois générations de contrôle. Aujourd’hui, ces générations peuvent être utilisées chacune en fonction des moyens mis en place sur l’infrastructure, et de la connaissance de cette dernière :

- *Première génération - contrôle à temps fixe* : en fonction de l’heure et parfois du jour, le système utilise un plan de feux prédéfini. Exemple : une configuration est appliquée le midi, tandis qu’une plus souple et équitable pour l’ensemble des voies d’un carrefour est appliquée la nuit. Classiquement, trois configurations

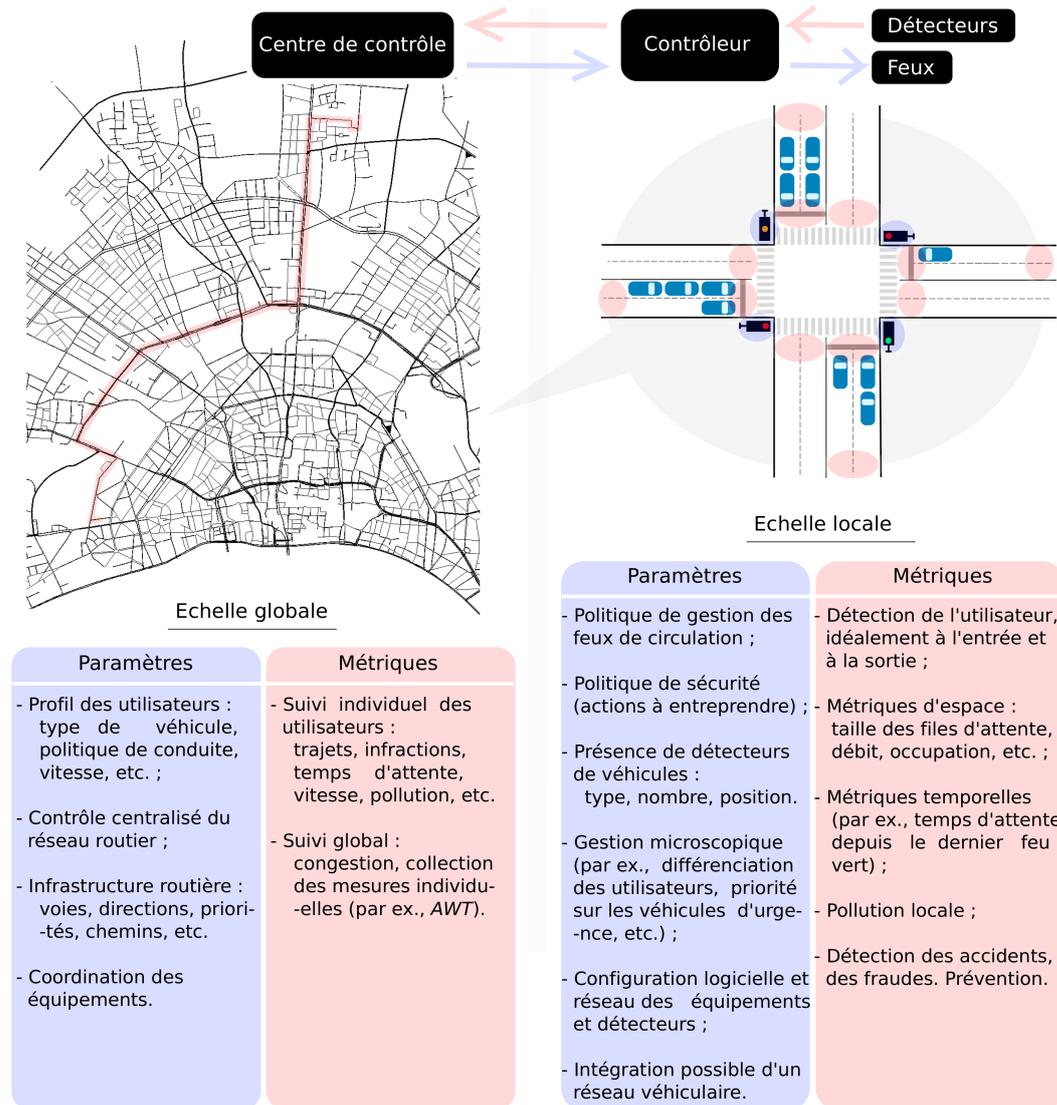


FIGURE 2.11 – Paramètres et métriques généraux des STI.

existent : pointes du matin, de l'après-midi, et le reste (hors pointes) [Webb]. Nous pouvons imaginer une formulation plus complète prenant en compte des plans spécifiques aux heures de pointe.

- *Deuxième génération - contrôle à temps dynamique - réactif* : des détecteurs sont utilisés afin de recueillir les données du trafic périodiquement (plusieurs minutes ou cycles). Ces données sont ensuite utilisées pour mettre en place un plan de feux en réponse aux informations reçues. Cette méthode est simple à mettre en place, mais nécessite toutefois une très bonne connaissance du système afin d'être efficace. Il s'agit de la première méthode à être apparue aux États-Unis à la fin des années 1980 avec l'apparition des UTCS (*Urban traffic control software*).
- *Troisième génération - contrôle à temps réel - adaptatif* : reprend le même principe que la deuxième génération mais cette fois-ci en temps réel. Ce type de contrôle programme dynamiquement les plans de feux en se servant des paramètres recueillis sur le terrain. Il est possible de calculer des valeurs telles que le temps de cycle, les phases ou encore leur ordonnancement. L'opération est effectuée de manière adaptative, c'est à dire en réaction directe au trafic. L'avantage de cette méthode est qu'elle peut s'adapter à de multiples situations, mais reste la plus compliquée à mettre en place (nécessité de cerner les informations à utiliser, comment les utiliser et se baser sur des théories parfois coûteuses en calculs ou mémoire).

### 2. 3. 3 Modes de fonctionnement d'un contrôleur de feux

Sur le terrain, les contrôleurs de feux peuvent être mis en place avec plusieurs modes de fonctionnement.

Tout d'abord, les *contrôles prédéterminés*, où l'enchaînement des phases s'effectue en marge du trafic routier, c'est à dire toujours dans le même ordre et avec un temps prédéfini en fonction de l'importance connue des voies. Les plans de feux mis en place peuvent être générés à l'avance : il est commun que des méthodes d'optimisation souvent longues soient employées afin d'obtenir le meilleur rendu possible. Il est en effet connu que trouver une séquence optimale pour un réseau de plusieurs intersections est un problème NP-difficile [HMB11].

Ensuite, les *contrôles semi-adaptatifs*, où des détecteurs sont mis sur certaines voies (exemple : les voies jugées peu importantes) afin d'ajuster des paramètres tels que les temps de feux ou encore l'ordonnancement des phases.

Enfin, les *contrôles adaptatifs*, qui découlent de l'émergence des nouvelles technologies et qui se calibrent en temps réel ou quasi temps réel sur le trafic, à l'aide de détecteurs au sol ou caméras, ainsi que de méthodes heuristiques.

Quelques auteurs ont essayé d'orienter les ingénieurs dans le choix d'un mode, par exemple [Cha96] ou encore [SBG98] décrivent leurs avantages et inconvénients, en fonction du matériel disponible et des données connues de l'intersection. Le tableau 2.1 donne l'exemple d'un choix de mode, se basant notamment sur l'état du

trafic, c'est à dire la demande en trafic par rapport à la capacité. Elle est notée par le rapport  $V/C$  (*espace occupé (volume) sur capacité*).

Configuration du réseau	Charge de l'intersection ( $V/C$ )	Phases		
		2	4	8
Croisement	0.80	Prédéterminé	Semi-adaptatif	Semi-adaptatif
	> 0.80	Prédéterminé	Prédéterminé	Prédéterminé
Réseau dense	0.80	Adaptatif	Semi-adaptatif	Adaptatif
	> 0.80	Prédéterminé	Semi-adaptatif	Adaptatif

TABLE 2.1 – Exemple d'orientation pour le choix d'un mode, en fonction de  $V/C$  (*espace occupé sur capacité*, la charge d'une intersection) [SBG98].

Notons au passage l'existence du *contrôle par ordinateur*, où les phases sont ajustées à distance depuis des CIGT ou directement sur le contrôleur de feux (p. ex., influence en cas d'imprévu, police, travaux).

## 2. 3. 4 Les systèmes de régulation du trafic

En Europe, trois grandes approches – complémentaires – de régulation du trafic routier se sont principalement développées et sont représentatives de ce qui se fait à travers le monde.

### 2. 3. 4. 1 L'approche Anglaise

La première approche, d'origine Anglaise, se base sur des principes mathématiques. Le modèle TRANSYT [CL87] est notamment né de cette démarche, puis a rapidement évolué vers TRANSYT 7 avant de devenir le système SCOOT [RB91]. Aujourd'hui, cette approche est l'une des plus populaires et compte de nombreux autres systèmes de régulation. SCOOT et SCATS, notamment, sont les deux principaux systèmes de gestion du trafic routier utilisés dans le monde [ZCZW].

SCOOT [RB91] (*Split Cycle Offset Optimization Technique*) est un système de contrôle développé par le TRL (*Traffic research laboratory*, Angleterre). À l'aide de détecteurs placés sur le terrain, SCOOT se base notamment sur un indice de performance afin de générer des plans de feux en fonction de la demande des utilisateurs. Cet indice est calculé par rapport au délai d'attente moyen, à la longueur des files d'attente et des arrêts sur le réseau. Cet aspect dynamique est réalisé à l'aide d'un aller-retour régulier de mesures et de décisions entre les équipements sur le terrain et un centre de contrôle. Cette centralisation et ce suivi régulier de la circulation impliquent un passage à l'échelle limité, car de gros besoins en calcul sont nécessaires et car tous les détecteurs doivent être interconnectés. Cela limite leur déploiement aux plus grands carrefours.

SCATS [SD80] (*Sydney Coordinated Adaptive Traffic System*) a été à l'origine développé pour Sydney et d'autres villes Australiennes. Il utilise une notion de hiérarchie (ce qui forme une certaine distribution sur le réseau). En effet, entre le recueil des données sur le terrain et le centre de contrôle, des contrôleurs intermédiaires sont

insérés. Ceci permet d'alléger la charge globale du système et d'avoir un contrôle découpé en plusieurs zones, l'ensemble des acteurs utilisant des communications synchronisées. De manière similaire à SCOOT, ce système ajuste le temps des cycles et autres paramètres en fonction des données recueillies afin de diminuer le délai et les arrêts. Il n'utilise toutefois pas la même stratégie : les valeurs recueillies permettent la sélection de plans de feux parmi une large bibliothèque, sur lesquels le système se base pour proposer des plans ajustés. Notons que les conventions utilisées par SCATS sont éloignées des standards NEMA, ce qui ne facilite pas son intégration sur les réseaux urbains [ACCJ06].

Selon [SS09], l'installation de l'une de ces solutions prendrait en moyenne 365 heures (630h pour SCOOT), coûterait au total en moyenne 55 000 \$ par intersection, et nécessiterait un temps d'entraînement moyen de 41 heures (dont 60h pour SCATS). Des chiffres considérables expliqués en premier lieu par l'équipement utilisé [MK07]. Les auteurs de [DFMN02] estiment que par rapport à un système fixe, SCATS réduit globalement les temps de déplacement de 8%, les retards de 28% et les arrêts de 42%. SCOOT pour sa part réduit globalement les temps de déplacement de 8%, les retards de 22% et les arrêts de 17%.

Tandis que SCATS et SCOOT sont conçus afin de ne prendre qu'une décision par cycle, d'autres systèmes plus modernes tels que OPAC [GPA01], RHODES [HMS92] ou encore InSync [Cha10] – pour ne citer que eux – analysent le trafic en temps réel et prennent des décisions seconde après seconde afin d'adapter dynamiquement un cycle. Nous pouvons également citer CRONOS [Boi+92 ; BMP06] et Prodyn [HFT83], les deux principaux systèmes agissant sur les carrefours Français [DP07]. Le premier a été mis au point au début des années 1990 par l'INRETS et permet de gérer un carrefour en se basant notamment sur des images en temps réel en provenance de caméras, qui fournissent des informations telles que l'occupation de la chaussée. Le deuxième a été développé par le CERT et a la particularité de se baser sur des mécanismes classiques de cycles et phases. Hélas, le coût et la durée d'installation de ces systèmes constituent un frein à leur développement.

#### 2. 3. 4. 2 L'approche Allemande

La deuxième approche, d'origine Allemande, se base sur des coordinations horaires pointues. Cette approche a su faire ses preuves et a essentiellement été mis en œuvre par Siemens jusqu'aux années 1990, notamment dans certains pays du nord de l'Europe où le style de conduite appelait à un trafic plus régulier. Cette solution a été par la suite abandonnée au profit de systèmes tels que SCOOT, ou de modèles temps réels comme MOTION, installé à Cologne.

#### 2. 3. 4. 3 L'approche Française

La troisième approche, d'origine Française, part d'un principe plus simple : *"aucune machine ne remplacera jamais ce qui est l'essence même de l'homme : l'esprit (...) et la compréhension"* (Louis V. Gerstner). Les modèles mathématiques sont ici mis de côté au profit de l'expérience humaine.

Les ingénieurs des ponts et chaussées mettent ainsi en place des stratégies au cas par cas. Sur Amiens par exemple, ceci se fait par le biais de systèmes centraux développés par THALES.

D'une manière plus générale, ils mettent également en avant des systèmes de supervision des réseaux de transports. Citons par exemple CLAIRE-SITI [SL13]. Ce système, d'initiative Française, est destiné à intégrer et à traiter des données issues de part et d'autres du réseau routier (trafic routier, stationnement, transport en commun, etc.), afin de fournir des indicateurs ayant trait à *la ponctualité (le retard)*, à *la régularité (l'attente)*, à *la fiabilité (la vitesse)*, à *au trafic (la demande, le débit, la concentration)*, à *la correspondance (les temps de transfert)*, à *l'environnement (l'émission de polluants)*<sup>1</sup>.

### 2. 3. 4. 4 Recherche Européenne

Notons également que la recherche Européenne pousse à ce genre d'initiative, allant vers l'analyse du trafic routier. Instant Mobility est, par exemple, un projet de recherche Européen étudiant la manière dont peut être améliorée la mobilité urbaine et la gestion du trafic routier<sup>2</sup>. Ce projet souhaite notamment créer un écosystème entièrement connecté agrégeant les données de trafic en provenance de diverses sources (transports publics, réseaux sociaux, flottes et camions commerciaux, etc.), afin de proposer des informations et services aux automobilistes.

### 2. 3. 5 Les équipements de détection

Les détecteurs de véhicules, répartis généralement autour des intersections en zone urbaine, jouent un rôle évident et fondamental dans le développement des transports intelligents. Ces équipements peuvent être décomposés en deux familles présentées ci-après [MK07].

#### 2. 3. 5. 1 Les systèmes non-intrusifs

Les systèmes non-intrusifs sont souvent volumineux et leur efficacité dépend des conditions environnementales. Ils sont généralement installés sur le côté de la chaussée. Les équipements les plus fréquents entrant dans cette catégorie sont les radars Doppler, lasers, à ondes ultrasoniques, micro-ondes ou encore acoustiques [MK07]. Des solutions tout aussi courantes sont les dispositifs de traitement d'images, filant directement la circulation. De nos jours, ce traitement peut être réalisé très facilement [JMMSAS13]. Sur la figure 2.12, nous utilisons par exemple le logiciel open-source CAUTOVE<sup>3</sup>. Selon [MK07], le prix de départ d'équipements fonctionnant par traitement d'images serait fixé à 5 000 \$, et pourrait s'envoler jusqu'à 26 000 \$ l'unité.

---

1. <http://claire-siti.inrets.fr/>

2. <http://instant-mobility.com>

3. <http://sourceforge.net/projects/cautove/>

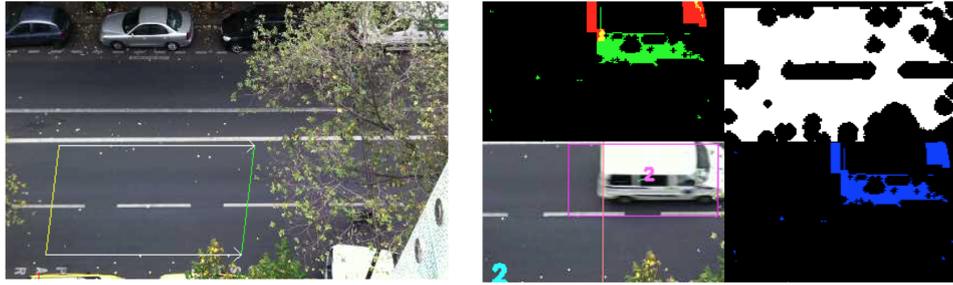


FIGURE 2.12 – Exemple d’utilisation de CAUTOVE. À gauche, nous définissons les voies à prendre en compte. À droite, une détection de véhicule, avec ce que perçoit le logiciel.

### 2. 3. 5. 2 Les systèmes intrusifs

Les systèmes intrusifs offrent généralement une détection ponctuelle, les systèmes précédents en offrant une généralement spatiale. Ils nécessitent une installation plus longue, car ils sont situés directement sur (ou sous) la chaussée. De plus, certains de ces systèmes nécessitent une alimentation indépendante, ce qui complique leur mise en place. En pratique, l'exemple le plus utilisé est la boucle électromagnétique, constituée de plusieurs spires de cuivre enfouies dans la chaussée : elles permettent de mesurer les variations d'inductance lorsqu'un véhicule passe à proximité. La détection de véhicule devient possible, mais également leur classification en fonction de l'importance des variations, ou la mesure de leur vitesse (sous réserve d'avoir deux boucles séparées d'une distance connue). Selon [MK07], l'installation de ces boucles coûterait en moyenne de 500 à 800 \$ l'unité.

Les capteurs magnétiques se comportent de manière similaire aux boucles électromagnétiques [CCV04 ; Che+05 ; MK07]. Les variations produites sur le champ magnétique terrestre lors du passage d'un véhicule sur un capteur sont mesurées et enregistrées par un magnétomètre. De la même manière, ces variations permettent de détecter un véhicule, de connaître son type, de mesurer sa vitesse ou encore sa longueur. [Che+05] montre que les magnétomètres peuvent enregistrer une signature d'une véhicule passant par dessus eux pour 99% des cas, et mesurer leur vitesse moyenne dans 90% des cas. Les matériaux ferreux des véhicules produisent en effet un dipôle magnétique particulier, qui vient perturber un champ magnétique terrestre *a priori* stable. Ces perturbations sont mesurées sur trois axes mais seul l'axe de la hauteur est pris en compte, l'axe des abscisses et des ordonnées pouvant être troublés par la présence d'un second véhicule situé sur une autre voie ou dans la même file. Ces principes sont illustrés sur la figure 2.13.

Ces équipements, qu'ils soient intrusifs ou non, sont généralement connectés à une unité de contrôle permettant leur considération par un système de transport intelligent. Il y a une nécessité de communication évidente, souvent mis de côté par la littérature, qu'il nous faut étudier. Dans la partie suivante, nous décrivons les différents réseaux de communication existant ainsi que leurs limites.

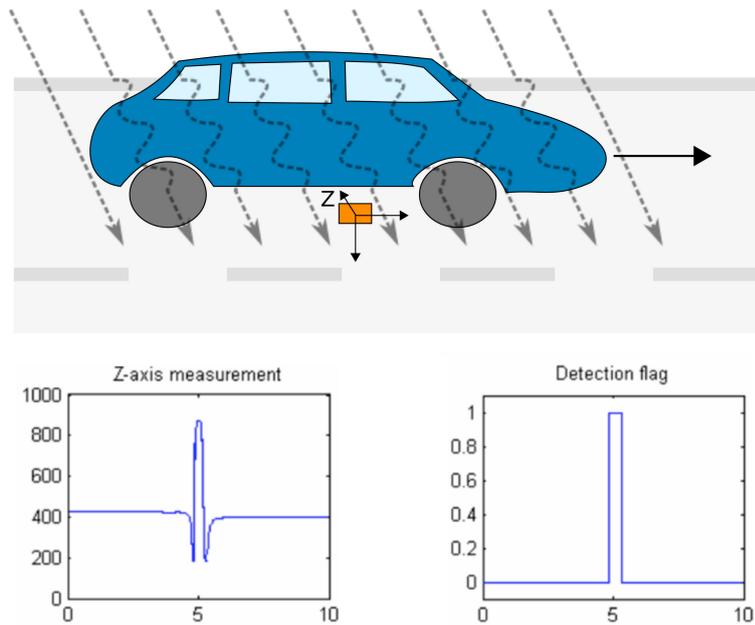


FIGURE 2.13 – Détection d’un véhicule par un capteur. Les mesures sont des exemples issus de [Che+05]. Le premier graphique affiche, en fonction du temps, les variations brutes recueillies par un magnétomètre échantillonné à 128 Hz au passage d’un véhicule. Le deuxième graphique retranscrit, en fonction du temps, l’enregistrement d’un signal de détection : ce dernier apparaît lorsque 10 échantillons successifs sont enregistrés et dépassent un certain seuil.

## 2. 4 Réseaux de communication

### 2. 4. 1 Les réseaux statiques

Les équipements de l’infrastructure routière, tels que les contrôleurs ou les détecteurs de véhicules, doivent échanger de l’information afin de communiquer des mesures ou des directives. Le plus souvent, ils le font par l’intermédiaire d’un réseau filaire (p. ex., fibre optique), généralement relié à un centre de contrôle. Toutefois, l’utilisation de réseaux sans fil est aujourd’hui facilement accessible, que ce soit sur une intersection entre équipements proches (p. ex., réseaux radios) ou entre contrôleurs, en utilisant des réseaux plus longue distance (*WAN*, *réseaux cellulaires*). Nous qualifions ce type de réseau de statiques, car la position de ses éléments n’est pas amenée à évoluer.

Si ces déploiements semblent faciles d’accès de par leur caractère statique, il y a toutefois un nombre important de questions à évaluer. Les réseaux sans fil possèdent des problématiques de couverture, tandis que les réseaux filaires possèdent des problématiques de déploiement (coût, travaux, etc.). Il nous faut également étudier les problématiques de fiabilité ou de performance (latence) : à quel débit le réseau en place est-il capable de délivrer ou de remonter des informations ? Est-ce suffisant ? Que se passe-t-il en cas de faute sur un équipement central ?

## 2. 4. 2 Les réseaux véhiculaires

Un réseau véhiculaire est un réseau où les véhicules sont dotés d'une unité de communication leur permettant de coopérer avec d'autres véhicules dans leur voisinage, de proche en proche (*V2V – vehicle-to-vehicle*).

Ce type de réseau communique généralement sous couvert du standard IEEE 802.11p [802] (WAVE – Wireless Access in Vehicular Environments), exploitant la bande DSRC 5.9 GHz et prévu pour être déployé dans de nombreux véhicules et équipements d'infrastructures. Ce standard, ainsi que d'autres (p. ex., ceux suggérés par l'ETSI – *European Telecommunications Standards Institute*), autorisent notamment les communications de véhicule à véhicule. Les applications utilisant ces communications sont typiquement classées en trois catégories : les applications de sécurité, les applications d'efficacité et de gestion du trafic et les applications de confort et d'infodivertissement [MZ09].

Les réseaux véhiculaires sont plutôt récents, et sont pour le moment essentiellement théoriques, à mesure qu'ils sont limités par leur déploiement. Sur la population de véhicules actuelle, peu de cas possèdent la technologie nécessaire à l'établissement d'un tel réseau. De plus, à supposer que l'ensemble des véhicules possèdent un capteur embarqué, la mobilité aurait pour effet de créer un réseau très volatile et fragmenté, difficilement utilisable en l'état. En effet, [NF13] étudient la topologie d'un réseau véhiculaire sur la ville de Cologne (Allemagne) et montrent que ce type de contrainte est courant et restreint l'utilisation de protocoles véhiculaires. Une solution possible, largement utilisée, considère le réseau de véhicules comme étant un réseau de smartphones, reliés à un oracle commun et échangeant des données sur un service.

Ce type de réseau est en plein développement, mais souffre toutefois d'importants problèmes de déploiement. De plus, même si les constructeurs automobiles commencent aujourd'hui à intégrer des interfaces radio aux véhicules, il est difficile d'imaginer équiper toute une flotte de véhicules, tout du moins pas avant plusieurs années ou de manière sélective (p. ex., taxis, transports en commun). Une solution à ce problème de déploiement serait de faire coopérer un tel réseau avec d'autres réseaux. Nous étudions cette possibilité dans les paragraphes qui suivent.

## 2. 4. 3 Les systèmes coopératifs

Un système coopératif est un réseau hybride où des acteurs mobiles coopèrent avec des acteurs fixes. Ce type de réseau est prometteur, en premier lieu car les équipements de l'infrastructure peuvent profiter des données d'un réseau véhiculaire (*V2I – vehicle-to-infrastructure*), et inversement (*I2V – infrastructure-to-vehicle*). Ainsi, l'intelligence est répartie entre différents éléments du réseau, fixes ou mobiles, ce qui permet de développer des applications qu'il aurait été impossible de mettre en place autrement. Dans certains cas, nous parlons de *systèmes interactifs* lorsque les véhicules peuvent échanger à la fois des données avec l'infrastructure, mais également avec les autres véhicules. L'avantage est considérable, le traitement et la diffusion

des informations pouvant bénéficier de la mobilité des véhicules, ainsi qu'un appui de données. Ceci est représenté par la figure 2.14.

Certaines initiatives, telles que CALM<sup>4</sup> (*Communications Access for Land Mobiles*), ont pour but de standardiser les communications dans ce type de réseaux. Le but de CALM est, notamment, de rendre transparente et continue l'utilisation des protocoles de communication sans fil et des interfaces de communications destinées à une large variété de scénario. Par exemple, utiliser conjointement des réseaux cellulaires et des interfaces 802.11p ou 802.15.4 afin d'obtenir un réseau étendu, réactif et favorisant la coopération entre les acteurs. CALM propose également des orientations technologiques en termes d'architectures (V2V, V2I, etc.) et de protocoles (p. ex., IPv6).

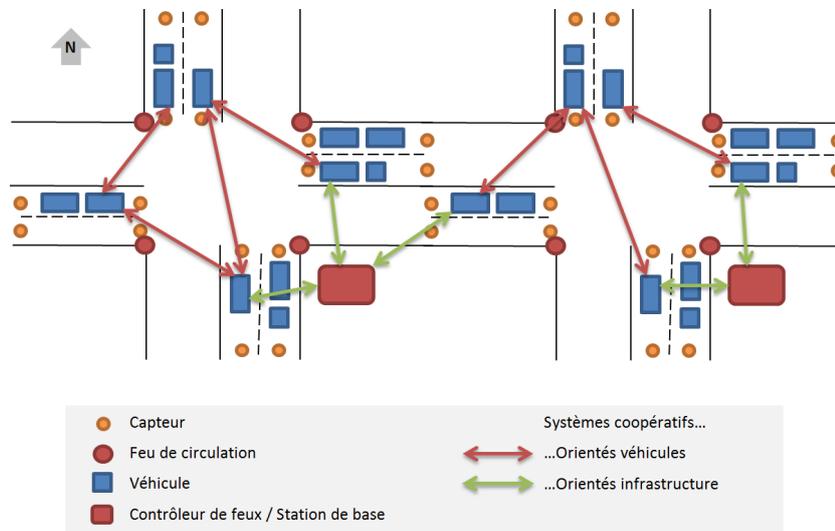


FIGURE 2.14 – Les systèmes interactifs.

Les auteurs de [WLZS05] proposent par exemple un réseau formé de trois types de nœuds : véhicules, magnétomètres, et contrôleurs à une intersection. Les magnétomètres diffusent en permanence des informations contenant notamment leur position. Les véhicules recevant des données de plus de trois capteurs peuvent ainsi calculer leur position par triangulation et envoyer le résultat ainsi que leur vitesse au contrôleur, qui est apte à prendre des décisions quant au changement des feux de circulation sur l'intersection. Dans [Wen+05], les auteurs reprennent un modèle similaire mais le développent sur plusieurs intersections. À partir de ces deux travaux, les auteurs vont même jusqu'à proposer un prototype [CCCT06].

Certains auteurs étudient la possibilité d'un système multi-agents. Wiering [Wie00] propose des méthodes d'apprentissage permettant aux véhicules (agents) de se déplacer en minimisant le temps d'attente aux intersections, et en échangeant des informations avec les feux de circulation. L'auteur introduit également la notion de vote [WVJK04] : chaque véhicule estime ce qu'il a à gagner si le feu auquel il se trouve passe au vert ; un vote est ensuite effectué parmi l'ensemble des agents présents à l'intersection. Houli *et al.* [HZY10] reprennent un modèle similaire à celui de

4. <http://calm.its-standards.info/Public/CALMintroduction.html>

Wiering qu'ils trouvent trop restrictif, et l'étendent à un modèle multi-objectif, afin de mieux gérer l'incertitude temps-réel du trafic : gestion des véhicules prioritaires, longueur des files d'attente, etc.

Un réseau véhiculaire étant nécessaire à l'obtention de ce type de réseau, les conclusions sont les mêmes : nous ne pouvons pas forcer l'ensemble des usagers à utiliser une technologie. Ceci nous impose donc soit de nous baser sur un modèle utopique, soit sur un modèle sélectif, où seuls les usagers bénéficiant de ladite technologie peuvent intégrer le système.

## 2. 5 Distribution du réseau de communication

### 2. 5. 1 Les systèmes distribués

Lorsque la charge de véhicules est élevée, un petit événement peut rapidement engendrer un embouteillage important [Hor13a]. Dans ce cas, communiquer avec un point de décision central peut ne pas être la solution la plus efficace [LH08]. Le temps nécessaire à communiquer avec une entité centrale peut empêcher le développement d'applications sur événements telles que les applications de détection d'incident ou d'embouteillage. Face à de tels systèmes centralisés, l'adoption progressive de systèmes distribués peut être un atout. Profitant des résultats de la branche des systèmes distribués, les dispositifs utilisant cette distribution peuvent facilement communiquer entre eux et rapidement adapter les plans de feux de circulation pour résoudre une situation. De plus, une telle architecture permet de rendre certaines intersections ou zones urbaines totalement autonomes et indépendante d'une quelconque entité centrale. D'une manière générale, nous pouvons dire que les systèmes distribués représentent les avantages suivants :

- **Coût** : plusieurs unités à petits prix au lieu d'une unique unité de calcul, si ce n'est davantage.
- **Performances de calcul** : les moyens centralisés sont généralement moins efficaces qu'une approche décentralisée, ou alors à très fort coût. De plus, le calcul parallèle devient naturel.
- **Sûreté de fonctionnement** : une panne ou faute matérielle ou logicielle sur un système centralisé peut être fatale à ce système. Sur un système distribué, les données manquantes peuvent être interpolées et les équipements peuvent se surveiller les uns les autres [FM11a], selon un principe bien connu : *"tout refus de communiquer est une tentative de communication ; tout geste d'indifférence ou d'hostilité est appel déguisé"* (Albert Camus).
- **Autonomie** : les éléments opérant sur un réseau centralisé sont interconnectés à ce réseau, alors qu'un système distribué peut agir de façon autonome.
- **Réactivité** : communiquer avec un point de décision central peut entraîner un certain temps de latence [LH08].

De nombreux projets et initiatives ont atteint une phase expérimentale en commençant des déploiements à moyenne échelle. CitySense [Mur+08] est un réseau sans fil urbain à l'essai déployé sur la ville de Cambridge (MA, USA), formant un réseau maillé. Il est composé de 100 ordinateurs embarquant un système sous Linux qui peuvent être programmés directement par les utilisateurs finaux. Même si l'objectif principal était de favoriser le développement d'applications pour des réseaux maillés, les nœuds ont été dotés de capteurs environnementaux et de pollution.

## 2. 5. 2 Les réseaux de capteurs sans fil

L'ajout d'un émetteur-récepteur sans fil à des capteurs magnétiques (sec. 2. 3. 5) permet de construire un réseau de capteurs sans fil classique. Les réseaux de capteurs sans fil s'appuient généralement sur des communications de courte portée et de faible coût, vitesse et puissance. Parmi les multiples normes de communication appropriées, IEEE 802.15.4 [IEE11], la technologie Zigbee, offre une couverture d'environ 50-100 mètres à 2,4 GHz pour un débit de données maximal de 250 Kbps et une faible consommation d'énergie [CPSS12]. IEEE 802.11p [802] est également un candidat tout à fait viable, comme il est prévu pour être déployé dans des véhicules et équipements d'infrastructures, il permettrait de la création d'un système coopératif. Les technologies 433 MHz sont également intéressantes, car elles fournissent des interfaces de communication à faible débit et à fort rayon. Toutes ces technologies partagent certaines caractéristiques communes en termes de performance et donnent des modélisations de réseaux similaires : le réseau de communication formé a toutes les caractéristiques d'un réseau local sans fil et peut être utilisé comme un réseau multi-sauts, ou interconnecté par des interfaces WAN. La technologie est donc prête à soutenir une application distribuée.

Les *réseaux de capteurs sans fil* sont un cas particulier des réseaux ad-hoc et se composent d'un large ensemble de capteurs à capacité et énergie généralement limitées, les rendant massivement accessibles et à faible coût. Dans de nombreux cas, les capteurs sont constitués des unités suivantes [Kna00] :

- Une *unité d'acquisition*, qui collecte des données environnementales et est en charge de la conversion analogique vers numérique. Dans notre cas, cette unité est un magnétomètre.
- Une *unité de calcul*, permettant notamment le lancement de procédures et de protocoles.
- Une *unité de communication*, rendant possible la connexion à un réseau sans fil (liens radio, émission et réception). Ce type de communication se fait de proche en proche et permet de s'affranchir des contraintes filaires (temps d'installation et facilité d'accès).
- Une *unité d'énergie*, qui permet la répartition de l'alimentation entre les différents composants. Dans de nombreux cas, les capteurs sont dispersés dans des zones pauvres en énergie, et sont dotés d'une batterie non-rechargeable et non-renouvelable [ASSC02]. Dans le cas du trafic routier, les batteries peuvent

tenir plusieurs années et il est possible de s'abstenir de contraintes énergétiques (p. ex., en raccordant les capteurs au réseau électrique urbain).

Le faible coût, la taille et la facilité d'installation de ces appareils, par rapport à des boucles électromagnétiques, permettent la création d'un réseau dense et plus réactif [CGML08]. Knaian [Kna00] évoque un coût de fabrication inférieur à 30 \$ par unité avec un microcontrôleur de 16 bits et une taille comparable à une pièce. Une fois déployé, ces dispositifs peuvent échanger des informations avec toutes les intersections pertinentes et résoudre rapidement une situation donnée sans intervention d'un serveur central. L'ensemble des intersections pertinentes peut être adapté dynamiquement à la situation et la communication multi-sauts peut soutenir la diffusion des données, même si aucune WAN n'est disponible, ou lorsque l'opérateur ne peut pas se permettre de gérer les données sans fil de tous ses appareils.

Les caméras représentent aussi une solution à faible coût, car leur installation nécessite peu de travaux et car elles peuvent accomplir de bonnes détections avec des techniques de traitement d'images, à moins que leur angle ou vision soient limités. [CPSS12] propose de combiner des capteurs magnétiques à des caméras.

La mise en place d'un système de transport passe par des technologies de communication qui existent déjà, ou qui se développeront dans un futur proche (p. ex., les réseaux coopératifs). De plus, les modèles d'intersections étudiés nous permettent de comprendre le comportement d'un système de régulation de trafic. Il est ainsi possible de contrôler les feux un à un, et d'appliquer un algorithme distribué qui s'adapte, pour chaque cas, aux files d'attente. Toutefois, cette politique de gestion distribuée reste à étudier en détail, car elle dépend *a priori* du réseau sous-jacent. Dans le chapitre suivant, nous caractérisons les graphes issus de tels réseaux, par l'intermédiaire de capteurs sans fil, qui peuvent être déployés de manière dense afin de rendre indépendant tout élément de l'infrastructure routière.



## CHAPITRE 3

---

### Déploiement et caractérisation de la topologie d'un réseau distribué pour la gestion routière

---

#### Vers une approche décentralisée

Les approches centralisées ne peuvent pas faire face à la complexité croissante des réseaux de circulation urbains [TSS07]. Comme vu en introduction, l'utilisation d'un réseau décentralisé permettrait – *a priori* – d'améliorer la gestion de la circulation localisée sur une ou plusieurs intersections. Si un centre de contrôle global semble nécessaire aujourd'hui, l'application de technologies distribuées permettrait néanmoins d'acquérir plus d'autonomie. En effet, chaque élément du réseau distribué peut collaborer avec ses voisins proches et peut prendre des décisions localement. Cette approche a l'avantage de pouvoir répondre rapidement à des événements sur le terrain (p. ex., accidents), en s'affranchissant d'une connexion avec un centre de contrôle pouvant entraîner des temps de latence significatifs.

Le succès des systèmes embarqués nous permet aujourd'hui de déployer un réseau dense de capteurs communiquant en sans fil. De plus, l'utilisation de liens sans fil permet de s'affranchir de connexions filaires, ce qui facilite l'installation des capteurs et réduit les coûts. Le déploiement d'un tel réseau à grande échelle permettrait, en outre, de fournir une infrastructure fixe favorisant le développement des applications véhiculaires, qui ont besoin d'utilisateurs pour former une infrastructure propre.

Toutefois, actuellement, peu de réseaux de ce type sont réellement déployés à l'échelle d'une ville. De tels déploiements se feront à un rythme très lent jusqu'à ce que la technologie elle-même prouve son efficacité. Dans ce chapitre, nous étudions les graphes issus de tels réseaux, et cherchons notamment à caractériser leur topologie, les graphes de connectivité qui en découlent, ainsi que leur rapprochement à des modèles de graphes aléatoires génériques. Cette étude peut servir de base pour les protocoles et l'évaluation de la performance des applications. Sur la base de quelques stratégies de déploiement que nous expliquons dans la section 3.3, nous créons des graphes de communication qui résultent de la mise en place de capteurs sur plus de

52 cartes de villes extraites de *OpenStreetMap*<sup>1</sup>, comme expliqué à la section 3.4. Nous analysons ensuite les propriétés structurelles des graphes qui en résultent en section 3.5 et discutons les aspects réseaux en section 3.6.

## 3.1 Topologie du réseau

Le déploiement de systèmes distribués sur une large zone pose généralement des problèmes de couverture. Leurs éléments ne sont en effet pas systématiquement reliés les uns aux autres, ce qui peut entraîner un partitionnement important. Les recherches dans les réseaux ad-hoc, maillés et de capteurs ont en effet montré que la topologie du réseau a un effet important sur ses performances [IA04; VS09; Puc+11; DHMP14]. La densité du réseau a un effet sur la congestion locale et sur la consommation d'énergie des nœuds. Elle influe notamment sur le protocole MAC. La densité des dispositifs est également un atout qui augmente la tolérance aux pannes : les mesures et les chemins de communication impactés par une panne peuvent être relayés à un dispositif voisin. Le diamètre du réseau (c-à-d. l'excentricité maximale de ses nœuds) a un effet sur le retard de bout en bout. Enfin, le partitionnement du réseau définit si le réseau distribué peut travailler en autonomie ou doit être interconnecté à un réseau cellulaire ou filaire.

Dans la littérature, plusieurs études confirment le fait que les propriétés structurelles du réseau ont un fort impact sur les performances des algorithmes, quels que soient les aspects du processus de communication. Ishizuka et Aida [IA04] examinent l'effet d'une topologie de capteurs sur la tolérance aux fautes et sur la probabilité de détection d'événement. Leurs simulations montrent en particulier que le placement initial des capteurs a un effet significatif sur la fiabilité du réseau. Vassiliou et Sergiou [VS09] étudient la performance de plusieurs algorithmes de contrôle de congestion pour des réseaux de capteurs sans fil sur de simples topologies, qui sont les mêmes que Ishizuka et Aida [IA04]. Ils montrent en particulier que les performances des algorithmes sont affectées par la topologie des réseaux considérés, ceci à cause des délais de transmission ou du taux de livraison. Puccinelli *et al.* [Puc+11] évaluent l'impact de la topologie sur le processus de collecte de données sur un réseau de capteurs, en utilisant des résultats expérimentaux. Leur conclusion est que la topologie doit être considérée afin d'évaluer complètement un protocole. Ducrocq *et al.* [DHMP14] évaluent l'impact de la topologie réseau sur le routage géographique. Ils montrent notamment que différentes topologies peuvent conduire à une différence allant jusqu'à 25% sur le taux de livraison et la longueur moyenne d'une route, et jusqu'à plus de 100% sur les coûts de transmission globaux.

---

1. <http://www.openstreetmap.org/>

## 3. 2 Stratégies de déploiement

### 3. 2. 1 Sur une intersection

Les travaux de la littérature déployant un réseau de capteurs (magnétomètres) sur une intersection s'accordent sur le fait d'utiliser au moins un capteur par voie, permettant de mesurer le débit moyen de véhicules. Certaines variantes existent, comme la possibilité de déployer un capteur par route, mais la vision du trafic est limitée par le rayon de détection des capteurs [ZYC09]. Tubaishat *et al.* [TQSS08] prouvent par simulation qu'utiliser un seul capteur par voie est un déploiement minimum. Ce type de déploiement peut convenir à la surveillance et à la détection du trafic sur l'ensemble d'une intersection. Toutefois, mettre en place deux capteurs au lieu d'un seul permettrait de mesurer plus finement la taille des files d'attente.

Le modèle d'intersection adaptatif classique, utilisé dans plusieurs travaux de la littérature, est ainsi représenté sur la figure 3.1. Le trafic est surveillé par deux capteurs situés sur chaque voie. Le premier est localisé au niveau du feu de circulation. Il comptabilise les départs de véhicules de l'intersection. Le second est placé à une distance appropriée avant le feu. Il comptabilise les arrivées de véhicules. La distance entre ces deux capteurs a toutefois une influence sur les performances du système. [YAKS10] propose de fixer cette distance à 8 véhicules. [ZCZW] propose de baser cette distance sur le feu vert maximum autorisé. Le paramétrage de cette distance étant fonction de métriques propres aux feux de circulation, nous l'abordons plus en détail dans le chapitre suivant, en section 4. 2.

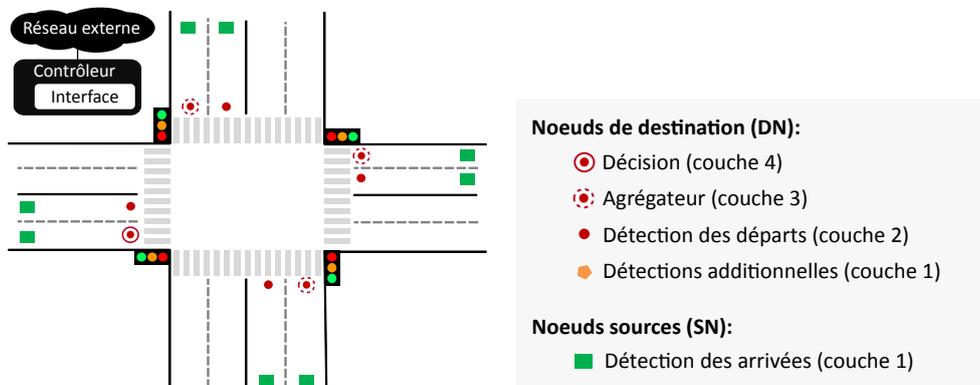


FIGURE 3.1 – Intersection à quatre directions où toutes les voies sont équipées de capteurs.

Le modèle d'intersection classique trouve ses limites dans le cas d'une coopération entre plusieurs intersections. La figure 3.2 illustre deux déploiements de capteurs supplémentaires sur une intersection. Ces scénarios diffèrent essentiellement du modèle classique par le nombre et la position des nœuds *destination*, qui surveillent et comptabilisent les départs de l'intersection. Les rôles affectés à un nœud de destination et les différentes couches hiérarchiques, qui sont propres à nos algorithmes, sont expliqués plus loin dans la section 4. 2. Sur l'architecture classique, axée sur les voies (figure 3.1), les nœuds de destination sont localisés sur la même voie que les nœuds d'arrivée. Un tel déploiement n'est pas capable de distinguer les directions

vers lesquelles se dirigent les véhicules lorsque plusieurs mouvements sont possibles sur une voie. Il n'est également pas capable de détecter si un véhicule est bloqué ou non au milieu d'une intersection. Sur l'architecture orientée vers les directions (figure 3.2(a)), les nœuds de destination sont localisés à l'entrée de chaque voie sortante. Un tel déploiement différencie mieux les véhicules qui quittent l'intersection et permet d'avertir les intersections voisines sur un flot arrivant. Toutefois, caractériser entièrement un flot de véhicules nécessite d'être capable d'enregistrer les signatures électromagnétiques de chaque véhicule, ou à minima de les identifier. L'architecture entièrement adaptative (figure 3.2(b)) combine les deux cas précédents et permet la meilleure précision de détection. Elle nécessite toutefois un déploiement plus dense. Le tableau 3.1 compare ces trois scénarios.

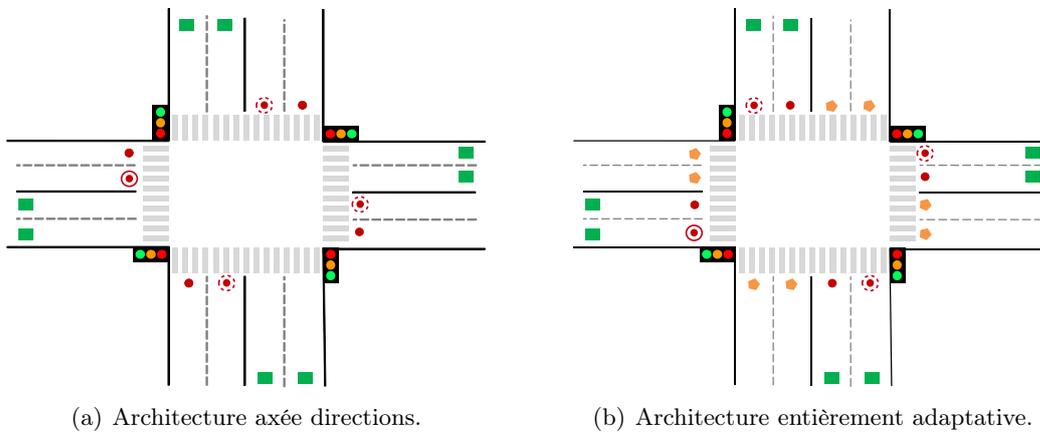


FIGURE 3.2 – Deux autres exemples d'architecture.

Architecture	Orienté voie	Orienté direction	Entièrement adaptatif
Nombre minimal de nœuds	$Voies\ entrantes \cdot 2$	$Voies\ entrantes + Voies\ sortantes$	$Voies\ entrantes + Voies\ sortantes$
Nombre maximal de nœuds	$Voies\ entrantes \cdot 2$	$Voies\ entrantes + Voies\ sortantes$	$Voies\ entrantes \cdot 2 + Voies\ sortantes$
Nombre de communications pour comptabiliser les véhicules d'une voie	2	1 + $Destinations\ de\ la\ voie$	2
Méthode de calcul de la taille d'une file	Naturellement sur les voies	Avec la signature des véhicules	Naturellement sur les voies
Temps additionnel lorsque de nouveaux véhicules arrivent	Oui	Approximativement	Oui
Détection du passage au feu rouge	Oui	Approximativement	Oui
Détection des véhicules bloqués sur l'intersection	Non	Approximativement	Oui
Identification des mouvements multiples sur une seule voie	Non	Oui	Oui
Finalité	Intersections isolées	Plusieurs petites intersections	Plusieurs intersections

TABLE 3.1 – Comparaison qualitative des architectures.

### 3. 2. 2 Sur une large zone

Corredor *et al.* [CGML08] étudient le déploiement de magnétomètres pour gérer le trafic routier sur des autoroutes intelligentes. Ils proposent de déployer de tels capteurs sur chaque voie afin de maximiser la probabilité de détection de véhicules. Ils proposent également de coupler ces capteurs avec des unités en bordure de route (RSU, *Road Side Units*) pour résoudre les problèmes de connectivité.

Hu *et al.* [HSZK08] proposent de déployer des capteurs sur la seconde rocade de Pékin (Chine) afin de gérer le trafic routier. Le déploiement est influencé de manière à ce que la topologie créée soit conforme à un graphe de type "petit monde", afin d'obtenir les avantages de ce type de structure. Ce type de réseau prend pour hypothèse que chacun de ses éléments puisse être relié à n'importe quel autre élément par une courte chaîne de relations. L'article propose d'optimiser le rayon de transmission des nœuds et d'affiner la localisation des nœuds de couverture élevée en utilisant un algorithme évolutionnaire.

CitySee [Mao+12] est un projet qui a pour but de déployer un réseau de capteurs dans la ville de Wuxi (Chine) afin de mesurer le niveau de  $CO_2$  en temps réel. Le papier modélise la question du déploiement comme un problème de placement de nœuds relais et évalue le nombre de nœuds additionnels déployés afin d'améliorer la connectivité.

Tous ces travaux proposent différentes stratégies de déploiement. Les graphes de connectivité en résultant, bien que rarement étudiés par les auteurs, devraient en conséquence différer légèrement. Dans la littérature, on considère généralement que les villes sont des réseaux sans échelle. En outre, les méthodes d'analyse de réseaux complexes, qui sont largement utilisés dans l'analyse des réseaux sociaux, sont également appliqués au cas des réseaux urbains [PCL06b; CLP06; PCL06a]. Toutefois, la topologie du réseau déployé sur une infrastructure urbaine dépend de la méthode de déploiement, et cette topologie a un fort effet sur les performances des protocoles réseaux.

Jusque ici, très peu de contributions proposent des modèles réalistes de déploiement de réseaux de capteurs urbains à large échelle. Aucune contribution à notre connaissance n'a caractérisé la topologie d'un réseau fixe distribué de capteurs qui pourrait être déployé et géré par une ville, même si les applications de ces réseaux pour les feux de circulation ou pour la gestion des limites de vitesse variables sont évidentes.

## 3.3 Modélisation d'un déploiement de capteurs à large échelle

### 3.3.1 Stratégie de base

Commençons par détailler la méthode de déploiement de capteurs supposée dans ce chapitre. Nous acquérons, tout d'abord, la carte d'une ville que nous supposons assez précise pour identifier les intersections avec et sans feux de circulation, le sens de circulation, les voies et les distances entre deux intersections. De telles données peuvent être obtenues depuis des systèmes d'informations géographiques publics tels que OpenStreetMap.

L'un des objectifs du réseau que nous construisons est de pouvoir détecter les véhicules afin d'alimenter un système de transport intelligent, sans l'appui d'un algorithme intelligent de corrélation des données. En nous basant sur les architectures présentées en section 3.2, nous supposons trois stratégies de déploiement possibles.

- La première consiste à déployer un nœud capteur à la fin de chaque voie. En d'autres mots, nous déployons, sur chaque intersection, un nombre de capteurs égal au nombre de voies entrantes, tel que représenté par les ronds jaunes sur la figure 3.3(a). Nous pensons que cette stratégie est représentative des architectures décrites sur le tableau 3.1. Il s'agit d'une stratégie de base, représentative d'un déploiement où les capteurs sont des magnétomètres.
- La deuxième stratégie consiste à déployer un capteur par route, tel qu'illustré sur la figure 3.3(b). Cette stratégie correspond au cas où les capteurs sont des caméras en hauteur, soutenues par un logiciel d'analyse vidéo, qui est capable de capturer toutes les voies simultanément. Cette stratégie peut également correspondre au cas où des capteurs sont des magnétomètres, tel que décrit par [ZYC09].
- Enfin, la troisième stratégie consiste à déployer un capteur par intersection, comme représenté par les ronds bleus sur la figure 3.3(c) (p. ex., avec une caméra fisheye).

Les trois stratégies sont possibles, mais diffèrent sur leur degré de précision. Dans la suite de cet article, les résultats présentés partent du principe que les capteurs sont déployés sur chaque voie. Nous retenons les voies, plutôt que les routes ou les intersections, car nous avons en tête d'utiliser un réseau de capteurs équipés de magnétomètres, tel que décrits dans la section 2.3.5. Toutefois, sur l'ensemble de nos résultats disponibles en ligne (sec. 3.9), il est possible de choisir la stratégie de déploiement voulue.

### 3.3.2 Réduction du nombre de capteurs

Chaque stratégie implique de déployer un nombre important de capteurs, ce qui occasionnerait un coût d'infrastructure qui serait jugé trop élevé par les urbanistes et

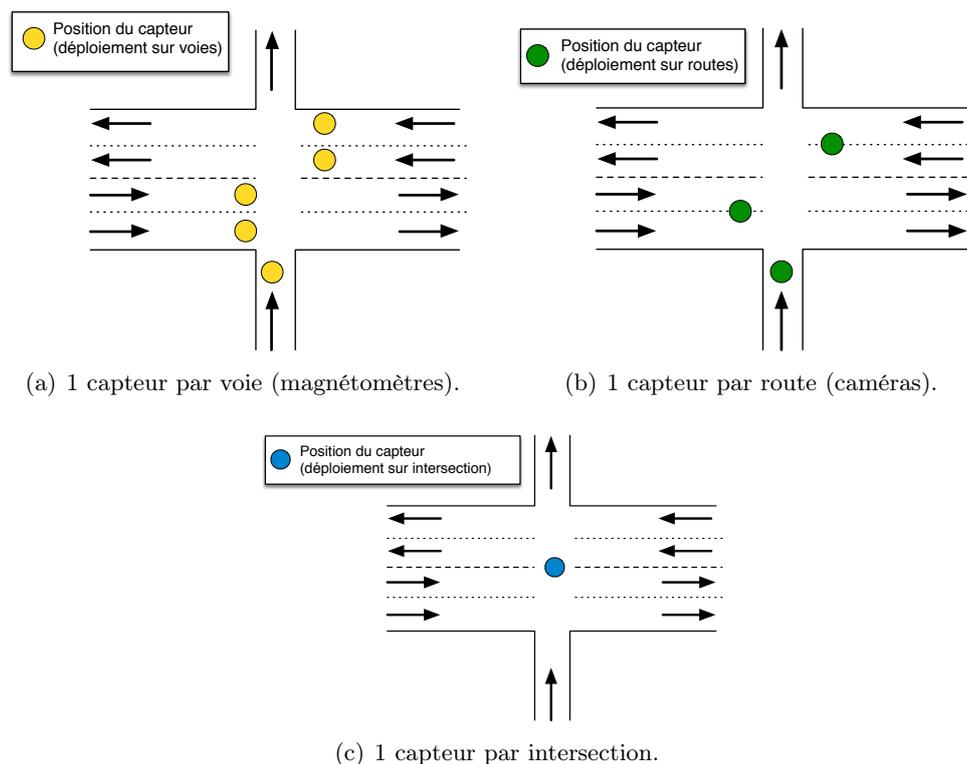


FIGURE 3.3 – Stratégies de déploiement de capteurs illustrés sur une intersection entre une route majeure (4 voies), une rue mineure (1 voie, 1 direction) et l'intersection elle-même.

ingénieurs urbains. Afin de réduire le nombre de capteurs sans impacter les capacités de gestion, nous décidons de ne pas déployer de capteurs entre deux intersections si elles sont trop proches.

D'une part, déployer des capteurs sur des rues trop petites n'est pas une nécessité, car des informations peuvent être collectées sur les rues et intersections adjacentes. En effet, une faible distance peut facilement être couverte par la portée de communication des capteurs situés sur les intersections adjacentes, ce qui permet d'interpoler facilement les données.

D'autre part, certaines structures complexes d'intersections (par exemple, des ronds points à feu de circulations) sont souvent représentées sur certaines cartes par une suite de plusieurs intersections adjacentes très rapprochées les unes des autres. Notons également que certains simulateurs, tels que SUMO, génèrent de fausses intersections lorsque, par exemple, la largeur d'une route est augmentée ou réduite d'une voie. Afin de ne pas faire de confusion et de considérer uniquement des éléments légitimes du réseau routier, nous considérons qu'il n'est pas nécessaire, dans tous ces cas, de déployer des capteurs sur les intersections concernées, comme représenté par la figure 3.4.

Nous choisissons par la suite de déployer des capteurs sur des rues séparant deux intersections d'au moins 50 mètres. En faisant varier cette distance de 10 à 100 mètres, comme représenté sur la figure 3.5 avec l'exemple de Paris, nous trouvons

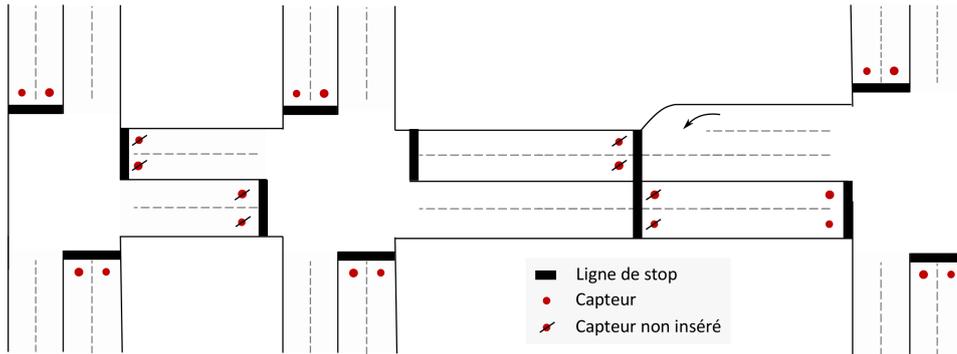


FIGURE 3.4 – Cas de figures où les capteurs ne sont pas intégrés sur les cartes. Le premier cas résulte d’une longueur de voie trop courte. Le deuxième cas de l’absence de plus de deux directions.

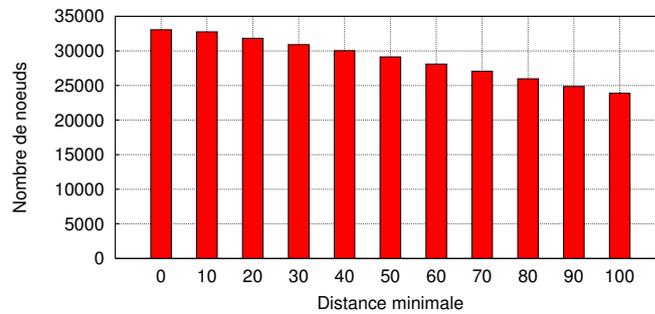


FIGURE 3.5 – Évolution du nombre de capteurs en fonction de la distance minimale séparant deux intersections.

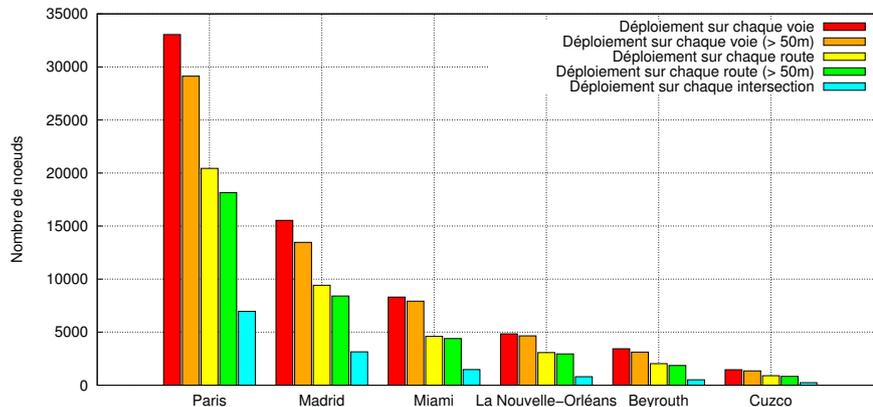


FIGURE 3.6 – Comparaison du nombre de capteurs déployés en suivant différentes stratégies, sur six scénarios représentatifs de notre jeu de données.

que le nombre global de capteurs décroît linéairement lorsque le seuil augmente. En d’autres mots, cette valeur n’a qu’un faible impact sur nos résultats, 50 mètres étant un juste milieu correspondant à des distances de communication réalistes pour le type de capteur que nous souhaitons utiliser (sec. 3.3.3).

La figure 3.6 représente le nombre de nœuds dans six villes représentatives de notre jeu de données (dont la sélection est justifiée à la section 3.4.2). Le cas *chaque*

*voie*, indiqué en rouge, correspond au placement d'un capteur par voie sur chaque intersection. Les secondes barres, en orange, représentent le nombre de capteurs restant quand nous supprimons les capteurs trop proches. Les troisièmes barres, en jaune, représentent les scénarios où un capteur est déployé par route au lieu de l'être sur chaque voie. Les quatrièmes barres, en vert, combinent ces optimisations. Les cinquièmes et dernières barres, en bleu, représentent les scénarios où un capteur est déployé sur chaque intersection.

### 3. 3. 3 Création du graphe de connectivité

Chaque méthode de positionnement de capteur produit un ensemble de nœuds,  $N$ , avec des coordonnées géographiques. Nous créons un graphe pondéré non orienté  $G = (N, E)$ , dont l'ensemble des arêtes ( $E$ ) est créé en comparant les distances euclidiennes entre les capteurs au rayon de transmission des nœuds.  $E = \{(i, j, \delta)\}$  est un ensemble de paires de nœuds pondérées et non triées,  $i$  et  $j$ , dont les poids  $\delta \in ]0 : 1]$  représentent la force de la connexion (p. ex. la qualité du signal sans fil). La normalisation de ces poids dans l'intervalle  $]0 : 1]$  peut être réalisée à partir des valeurs minimales et maximales de la métrique représentant la force de la connexion.

Le poids des arêtes est calculé en se basant sur le modèle fourni par les capteurs *Sensys Networks VSN240*<sup>2</sup>, qui sont utilisés sur les routes du monde entier et qui peuvent être déployés de manière dense [HKV08]. Ces nœuds utilisent une puissance de sortie nominale de 0 dBm et ont une sensibilité de réception de -95 dBm dans la bande de fréquence 2.4 GHz. Nous confrontons ces valeurs à un modèle de propagation simplifié qui correspond à une interface réseau IEEE 802.15.4 de 2.4 GHz ([Lee ; Mar99]). Le modèle définit l'affaiblissement de propagation (*path loss*, en décibels) sur une distance de  $d$  mètres telle que :

$$PL(d) = \begin{cases} 40.2 + 20 \log_{10}(d), & 0.5m \leq d \leq 8m \\ 58.5 + 33 \log_{10}(d/8), & d > 8m \end{cases} \quad (3.1)$$

La figure 3.7 montre la relation entre la distance séparant deux capteurs et le poids calculé. Notons que ce modèle, qui définit simplement un rayon de transmission à ce niveau de l'analyse, peut s'adapter à la plupart des technologies qui fonctionnent dans la bande S (2 GHz à 4 GHz), de même qu'à d'autres bandes de fréquence étroites telles que la bande 5.9 GHz utilisée par IEEE 802.11p (WAVE).

Les graphes que nous générons sont donc différents de ceux issus des graphes suivant les rues et intersections, comme illustré sur la figure 3.8. D'une part, les liens sans fil ne suivent pas strictement les routes et plusieurs nœuds existent généralement sur une intersection. De plus, les graphes suivant les rues peuvent être orientés (rues à sens unique), ce qui n'est pas le cas de nos graphes.

2. <http://www.sensysnetworks.com/products/sensor/>

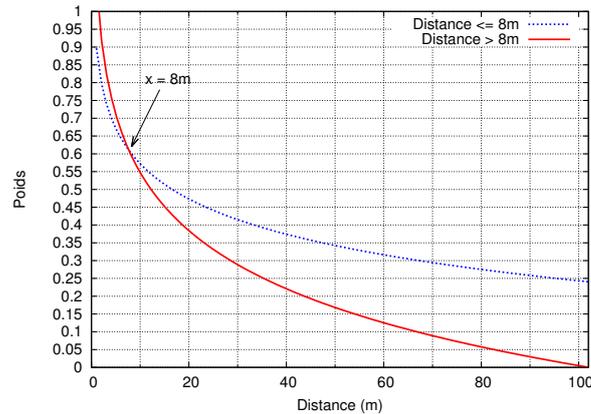
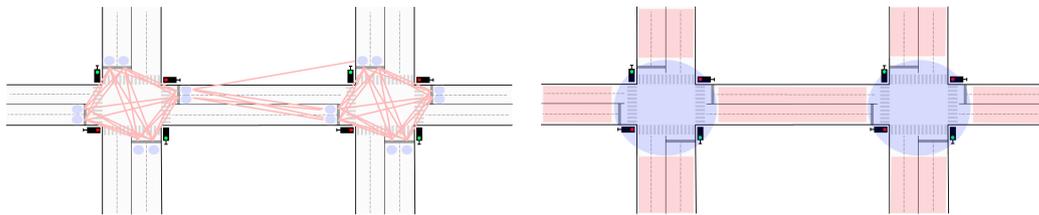


FIGURE 3.7 – Pondération des arêtes en fonction de la distance entre les nœuds.



(a) Les graphes que nous générons se basent sur un déploiement de capteurs.

(b) Les graphes intersections-rues se basent sur l'infrastructure routière.

FIGURE 3.8 – Différence entre les graphes que nous générons et les graphes intersections-rues.

## 3. 4 Création du jeu de données

### 3. 4. 1 Méthode et outils

Nous appliquons la méthode de création de graphe décrite dans la section 3. 3 sur un ensemble de 52 cartes de villes extraites depuis *BBBike.org*<sup>3</sup>, un service qui permet de récupérer les cartes OpenStreetMap de plus de 200 villes et régions mondiales. Notons que certaines cartes choisies ne sont pas strictement délimitées aux villes indiquées, il s'agit parfois d'une région autour de ces villes, incluant la périphérie (exemple de Paris et de ses nombreuses banlieues).

Dans le but de filtrer les informations contenues dans ces cartographies complètes, nous supprimons les éléments qui ne sont pas utiles à notre étude (p. ex. pistes cyclables, aires piétons) à l'aide de *NETCONVERT*, issu de SUMO 0.19 [BBEK11 ; KEBB12]. Les cartes en résultant sont plus faciles d'utilisation que leurs équivalentes OpenStreetMap. Afin de ne pas surcharger le réseau, nous gardons uniquement les routes principales et secondaires, tel que défini sur le wiki OpenStreetMap<sup>4</sup>. Nous supprimons ainsi les routes qui sont inaccessibles pour des véhicules motorisés, ainsi que certaines routes mineures (p. ex., quartiers résidentiels).

3. <http://download.bbbike.org/osm/>

4. <http://wiki.openstreetmap.org/wiki/Key:highway>

L’algorithme 3.1 résume les principales opérations nécessaires à la génération de nos graphes. Il est décomposé en deux principales étapes. La première, entre la ligne 1 et la ligne 15, décrit la manière dont les nœuds sont placés. La carte est tout d’abord convertie au format approprié. L’algorithme parcourt ensuite chacune des intersections référencées (ligne 3) et complète sa liste de nœuds en fonction de la méthode de déploiement paramétrée. La deuxième étape, décrite entre la ligne 16 et la ligne 21, énumère tous les couples de nœuds existants et regarde pour chacun si une arête peut être ajoutée, en fonction de la portée de communication – ou du modèle de propagation – définie.

### 3. 4. 2 Sélection de six scénarios représentatifs

Nous créons les graphes pour 52 villes que nous avons extraites de la base de données publique. Parmi ces 52 scénarios, nous sélectionnons 6 villes représentatives pour illustrer cette étude. Tous les jeux de données et les résultats sont disponibles en ligne à l’adresse <http://g.sfaye.com/> (sec. 3. 9).

La première propriété qui a influencé notre choix est l’aire couverte par la ville. Elle est indépendante de la méthode de déploiement. Nous cherchons à inclure aussi bien des villes étendues que des petites, afin de prendre en compte la diversité dans les règles d’urbanisme. Les surfaces couvertes par chacune des 52 villes sont représentées sur la figure 3.9(a). Nous sélectionnons la plus grande et la plus petite ville : respectivement *La Nouvelle-Orléans* et *Beyrouth*.

Le second critère de sélection est la densité de nœuds qui est produit par la méthode de base. La densité a un effet direct sur la performance du réseau : elle influence la probabilité de collisions au niveau de l’accès au support de communication, ainsi que la probabilité de congestion sur le réseau de communication. La figure 3.9(b) représente le nombre moyen de nœuds déployés par kilomètre carré sur chacune des 52 villes. La barre verte représente le nombre de nœuds appartenant à des voies qui mènent à un feu de circulation, lorsque cette information est disponible. Le cumul avec la barre rouge représente la densité de tous les types de nœuds (c-à-d., le nombre de voies totales). Nous incluons à notre sélection la plus dense et la moins dense des villes : respectivement *Miami* et *Cuzco*.

Finalement, nous ajoutons au jeu de données deux réseaux supplémentaires dont la taille et la densité sont intermédiaires à l’ensemble de nos scénarios. Afin de faire notre choix, nous étudions les profils de distribution des interdistances entre tous les couples d’intersections existants sur les cartes. Nous choisissons de comparer ces distributions car elles sont représentatives de la topologie des infrastructures routières. La figure 3.10 représente la diversité des distributions que nous trouvons dans notre jeu de données sur quelques scénarios. En analysant tous les graphes, nous trouvons deux principaux profils de distribution :

1. La plupart des villes (37 dans notre jeu de données) présentent une distribution des distances unimodale et asymétrique, biaisée vers la gauche. Miami (Fig.3.10(a)), Beyrouth (Fig. 3.10(b)) et Paris (Fig. 3.10(c)) suivent cette tendance. Ce type de distribution indique que ces villes ont une densité et une

---

**Algorithme 3.1:** algorithme simplifié de création d'un graphe

---

**Entrées :**  $C$ , carte OpenStreetMaps.

**Sorties :**  $G$ , graphe.

**Paramètres :**  $m$ , méthode de déploiement ;  $p$ , portée de communication ;  $t$ , type d'intersection.

```

// Étape 1 : placement des nœuds
1 carte ← netconvert(C);
2 noeuds ← ();
// Parcours de chaque intersection de la carte
3 pour chaque  $i \in \text{carte}$  (intersections de type t) faire
    // Création des nœuds en fonction de la méthode de déploiement
4     si  $m == \text{intersection}$  alors
5         |  $\text{noeuds} \leftarrow \text{noeuds} \cup \text{position}(i)$ ;
6     sinon si  $m == \text{route}$  alors
7         | pour chaque  $r \in \text{routes menant vers } i$  faire
8             |  $\text{noeuds} \leftarrow \text{noeuds} \cup \text{position}(r)$ ;
9         | finprch
10    sinon si  $m == \text{voie}$  alors
11        | pour chaque  $v \in \text{voies menant vers } i$  faire
12            |  $\text{noeuds} \leftarrow \text{noeuds} \cup \text{position}(v)$ ;
13        | finprch
14    finsi
15 finprch

// Création des liens
16 liens ← ();
17 pour chaque couple  $a, b \in \text{noeuds}$  faire
18     | si  $\text{lien}(a, b, m) > 0$  alors
19         |  $\text{liens} \leftarrow \text{liens} \cup (a, b, \text{lien}(a, b, m))$ ;
20     | finsi
21 finprch

// Création du graphe
22  $G \leftarrow (\text{noeuds}, \text{liens})$ ;

```

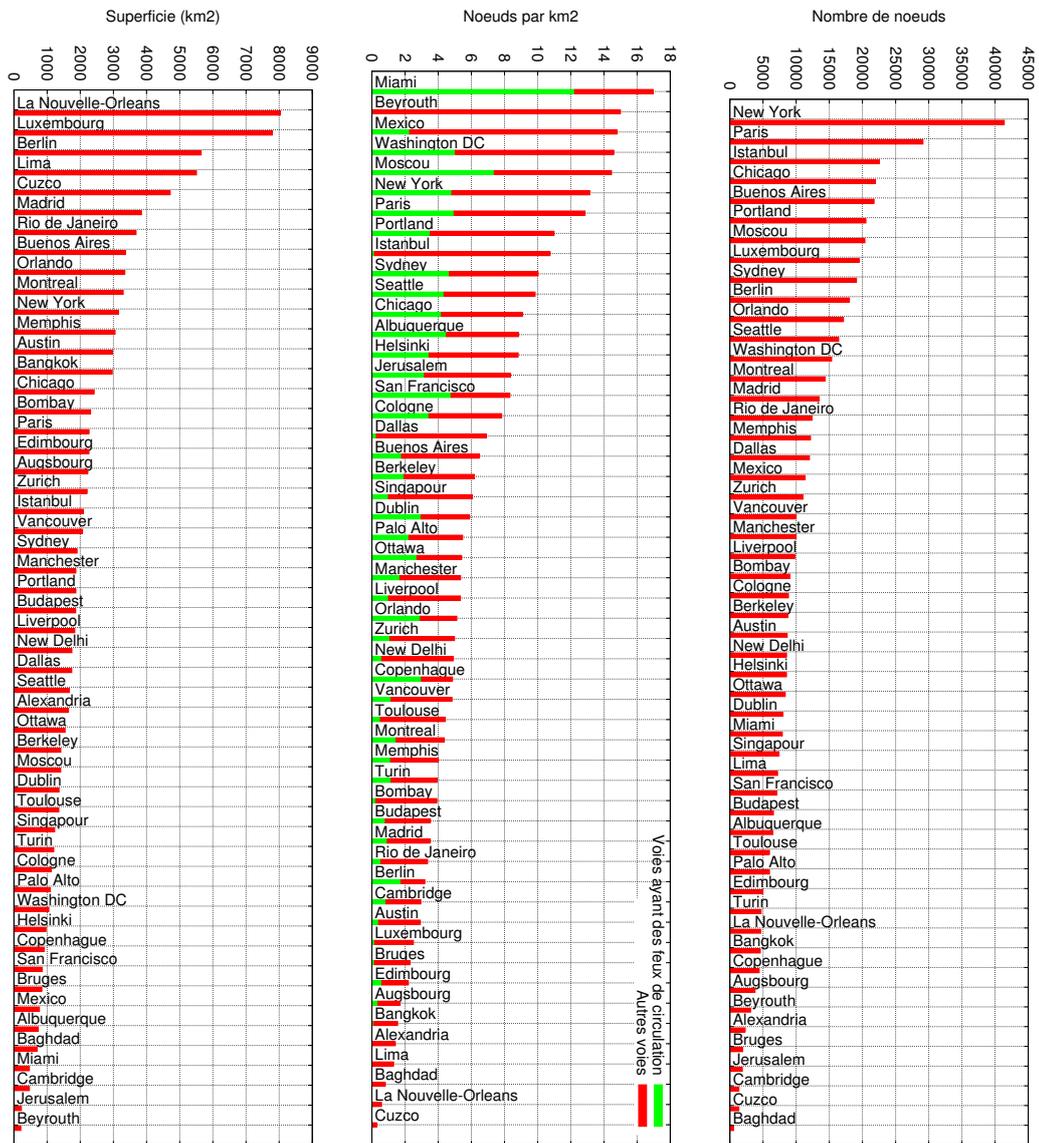
---

**netconvert(C)** : convertit  $C$  en format SUMO. Ne conserve que les routes principales et secondaires (sec. 3.4.1) et supprime les intersections et routes ne respectant pas nos critères (sec. 3.3.2).

**position(i)** : retourne la position du capteur sur l'élément  $i$ .

**lien(a,b,m)** : retourne la pondération représentant le lien entre le nœud  $a$  et  $b$ , en fonction de la portée de communication ou du modèle d'affaiblissement de propagation  $m$  (sec. 3.3.3). Si cette valeur est inférieure à 0, les nœuds ne sont pas connectés.

---



(a) Taille des villes ( $km^2$ ). (b) Densité de nœuds par  $km^2$  - (c) Nombre de nœuds - Toute Intersections avec et sans feux. type d'intersection.

FIGURE 3.9 – Caractérisation du jeu de données.

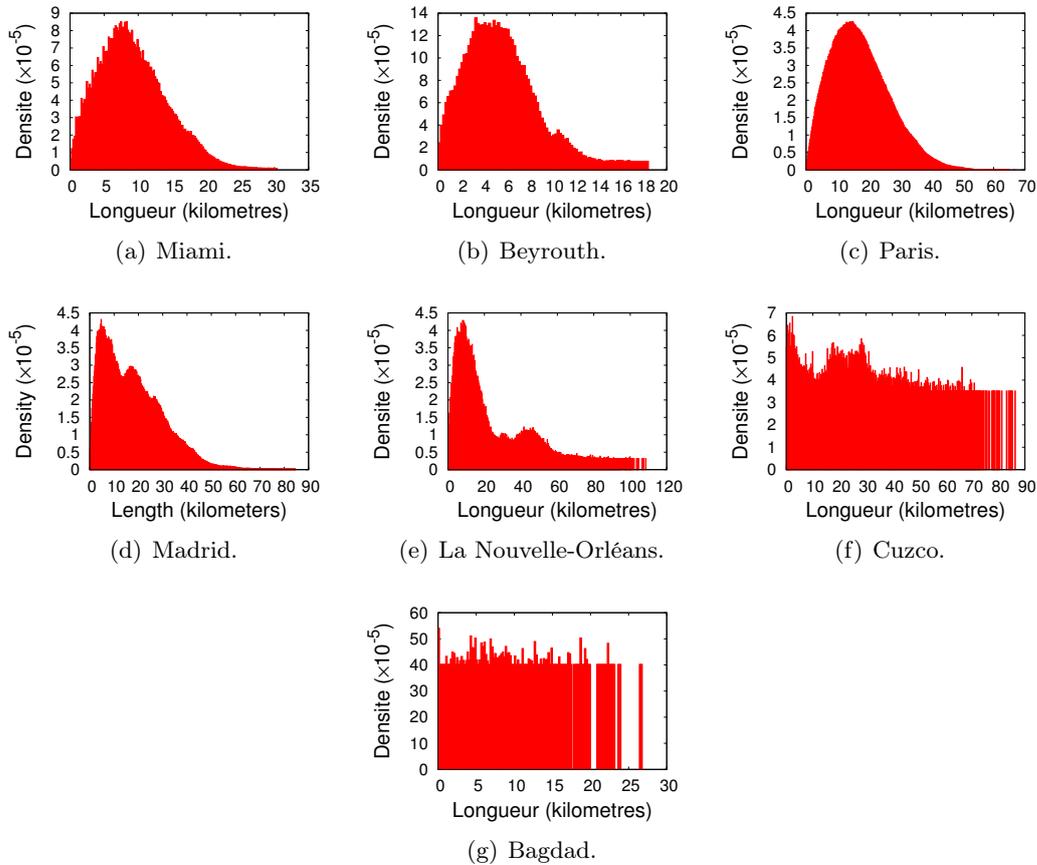


FIGURE 3.10 – Distribution des distances entre les intersections sur des villes représentatives.

répartition des intersections relativement régulier. C'est typiquement le cas pour les villes géométriques (p. ex., Miami), les villes éparées (Cuzco) ou les villes respectant des schémas particuliers (p. ex., Paris, dont l'unique centre est entouré de banlieues). La longueur du pic donne une indication sur la régularité de la structure de la ville. Son évolution vers des valeurs plus petites est plus prononcée sur des réseaux routiers denses.

2. Certaines villes (11 dans notre jeu de données) montrent une distribution des distances bimodale, telles que Madrid (Fig. 3.10(d)) ou La Nouvelle-Orléans (Fig. 3.10(e)). Cela signifie généralement que les règles urbaines sont différentes pour le centre ville et pour les zones périphériques, et que la frontière entre les deux zones est brutale.

Notons que certaines distributions (4 dans notre jeu de données) sont quasiment uniformes et présentent des profils déviant des deux principaux décrits. C'est par exemple le cas de Bagdad (figure 3.10(g)), qui suit ce type de profil, et dans une certaine mesure Cuzco (3.10(f)), même si quelques pics apparaissent sur les distances les plus faibles. Notons que si ce type de distribution est indépendant de la taille de la ville ou de sa densité, il est plus fréquent sur les réseaux peu denses.

En nous basant sur les deux principaux profils existants, nous ajoutons Madrid et Paris à nos cartes représentatives. La figure 3.9(c) représente le nombre de nœuds déployés dans chaque scénario. En termes d'implémentation réseau, la taille du réseau a une influence directe sur le schéma d'adressage et sur la mémoire requise pour les tables de routage, également sur le coût de déploiement.

Les 6 scénarios que nous avons sélectionné reflètent bien la diversité du jeu de données, car nous avons des réseaux larges (Paris), des petits (Cuzco) et des intermédiaires. La figure 3.11 représente la localisation des capteurs sur ces 6 cartes lorsqu'un capteur est mis sur chaque voie.

## 3. 5 Analyse du graphe de connectivité

### 3. 5. 1 Degré des nœuds

La figure 3.12 représente le degré moyen des nœuds pour notre jeu de données complet, c.-à-d. le nombre moyen de nœuds qui sont dans le rayon de transmission d'un nœud. La figure 3.13 est un agrandissement du degré moyen des nœuds pour les six cartes sélectionnées.

En termes de réseaux de communication, le degré d'un nœud représente le nombre de concurrents avec lesquels il doit rivaliser pour accéder au canal sans fil. Comme un nœud doit partager la bande passante du canal avec tous ses voisins, la planification du réseau doit viser un degré relativement faible. Pourtant, une valeur trop faible n'est pas souhaitable, car un certain degré offre une plus grande diversité de parcours et de redondance.

La figure 3.12 montre que tous les graphes ont un degré moyen similaire qui oscille entre 4 et 9 voisins, et entre 5 et 8 voisins pour les six cartes sélectionnées. Ces chiffres correspondent assez bien au cas d'utilisation classique des normes sans fil d'aujourd'hui. Nous pouvons remarquer que les degrés moyens de tous les réseaux sont assez proches les uns des autres. Des villes comme Beyrouth, dont le réseau routier est relativement uniforme, ont un degré moyen plus élevé que d'autres villes comme Paris, par exemple, qui ont une grande zone suburbaine.

La distribution des degrés est une mesure classique pour caractériser les grands graphes de terrain. Nous pouvons remarquer sur les figures 3.12 et 3.13 que le degré maximal est relativement élevé (près de 35 pour Cuzco et Madrid), mais pas trop.

La moyenne empirique et l'écart type sont très différents dans toutes les distributions de degrés. Par conséquent, ces distributions ne peuvent pas être représentées par des distributions de Poisson. Ils ne correspondent pas non plus à une distribution en loi de puissance, comme le montre la représentation log-log de la distribution de degrés, qui est loin d'être linéaire, sur la figure 3.14 pour Paris.

Nous avons finalement approché la distribution empirique des degrés par une distribution gamma dont : (1) le paramètre *d'échelle* ( $\theta$ ) est calculé, pour chaque graphe, comme étant le ratio entre la variance empirique ( $\sigma^2$ ) et la moyenne empirique

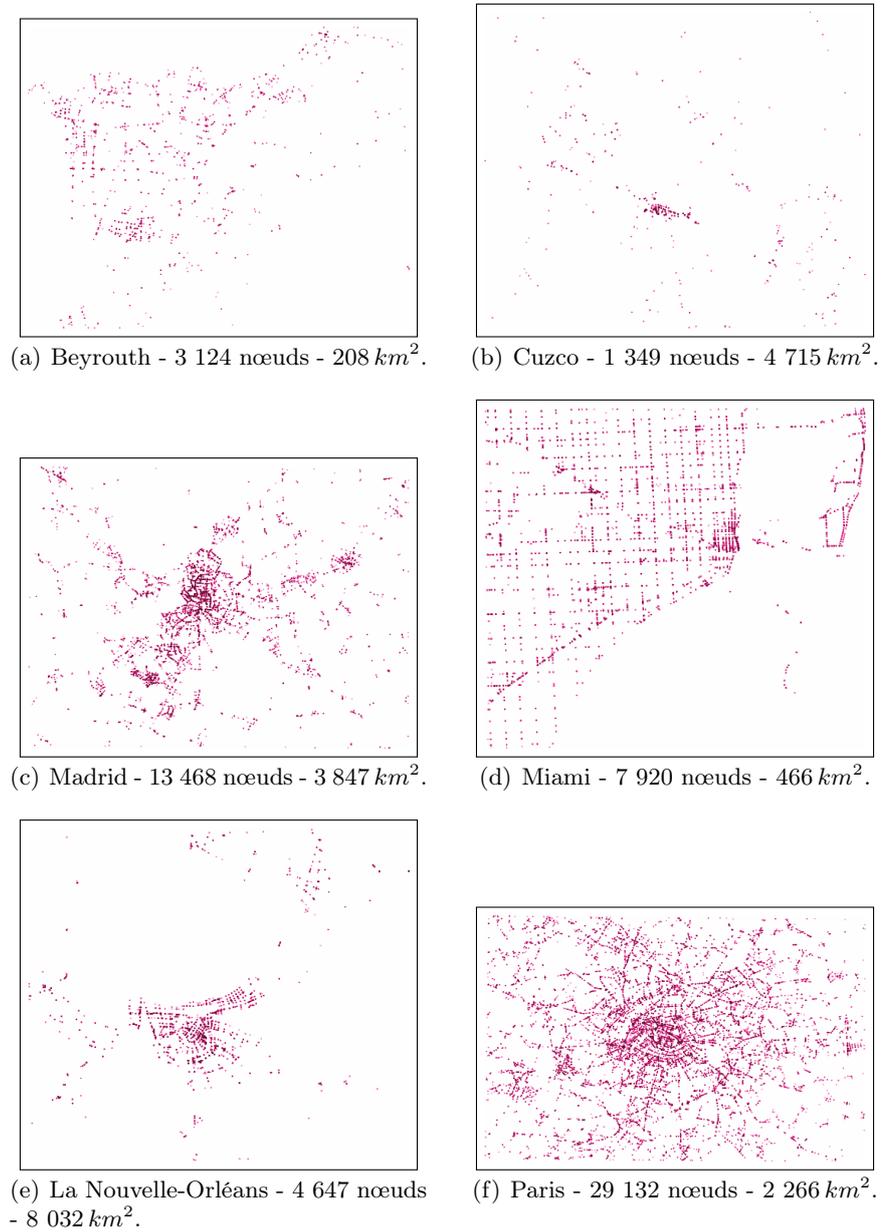


FIGURE 3.11 – Les six cartes représentatives de notre jeu de données, détaillées dans ce chapitre.

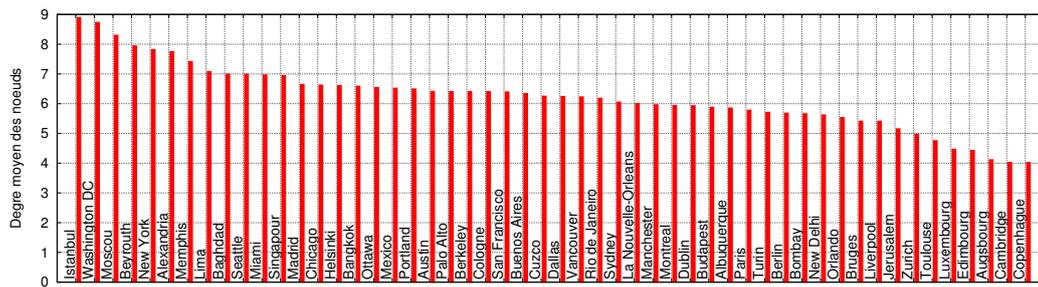


FIGURE 3.12 – Degré moyen sur notre jeu de données complet.

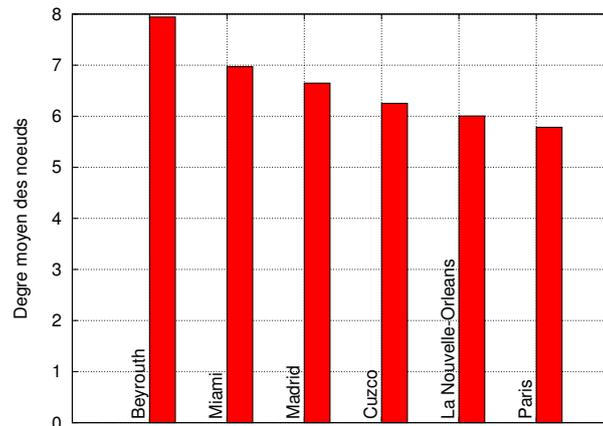


FIGURE 3.13 – Degré moyen sur les six cartes.

$\bar{x}$ , et (2) le paramètre de *forme* ( $k$ ) est calculé comme étant le ratio entre la moyenne empirique et le paramètre d'échelle :  $\theta = \sigma^2/\bar{x}$  et  $k = \bar{x}/\theta$ .

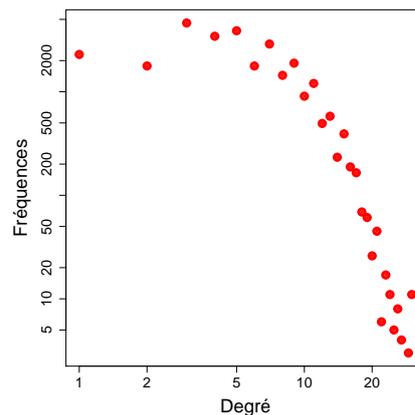


FIGURE 3.14 – Distribution des degrés pour le scénario de Paris (échelle log-log).

Les distributions ajustées sont représentées par les courbes noires sur la figure 3.15 et les valeurs des paramètres sont reportées dans le tableau 3.2. Excepté pour les valeurs élevées (p. ex. autour d'un degré de 23 pour le scénario de Beyrouth), l'ajustement est bon. La figure 3.16 représente le diagramme quantile-quantile qui compare les distributions empiriques et ajustées. Plus proche les points sont de la ligne diago-

nale, meilleure est la correspondance. Ce graphe nous dit qu'un ajustement utilisant une distribution gamma est relativement précis et s'écarte légèrement des degrés élevés. Si les valeurs les plus basses et la queue des distributions diffèrent, leurs modes sont tous situés autour d'une valeur de 5 nœuds, ce qui nous donne une bonne idée du niveau de congestion attendu dans le réseau.

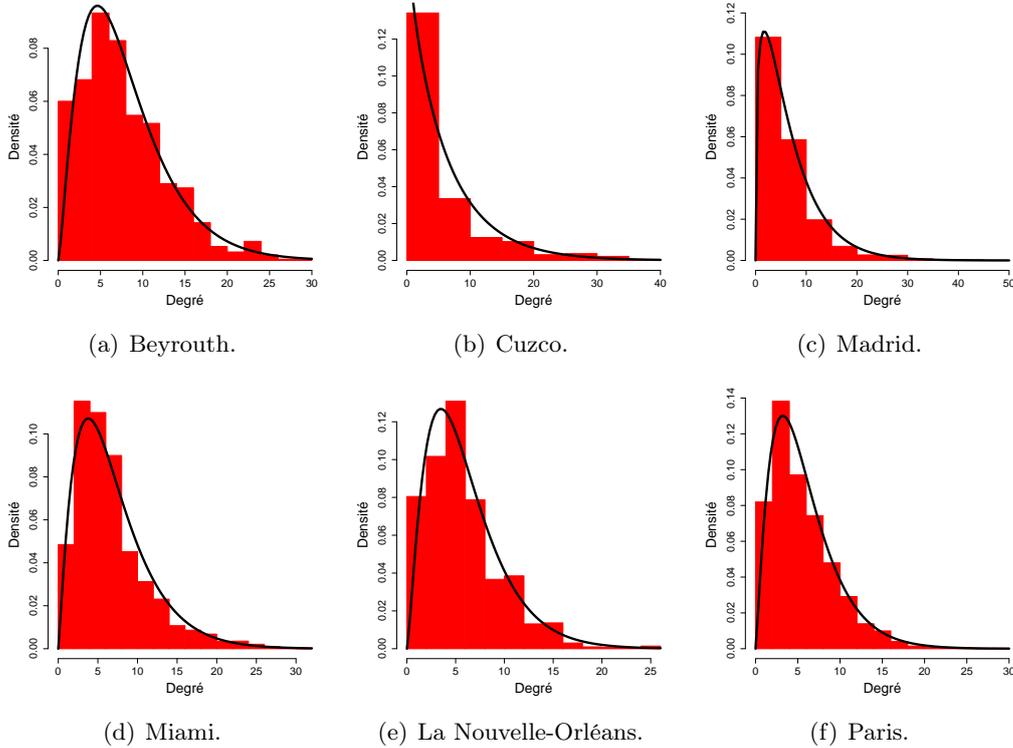


FIGURE 3.15 – Distributions des degrés empiriques et ajustés.

Ville	Forme ( $k$ )	Échelle ( $\theta$ )
Beyrouth	2.397899	3.312492
Cuzco	0.9380117	6.666784
Madrid	1.347767	4.933085
Miami	2.183021	3.192106
La Nouvelle-Orléans	2.380301	2.52304
Paris	2.26065	2.558382

TABLE 3.2 – Paramètre des distributions Gamma.

### 3. 5. 2 Coefficient de clustering

Le *coefficient de clustering* (ou *coefficient d'agglomération*) d'un nœud dans un graphe est la probabilité que deux voisins de ce nœud soient eux-mêmes des voisins. Il explique la présence de communautés dans le graphe et il s'agit d'une mesure classique pour caractériser les graphes. En termes réseaux, un coefficient de clustering élevé

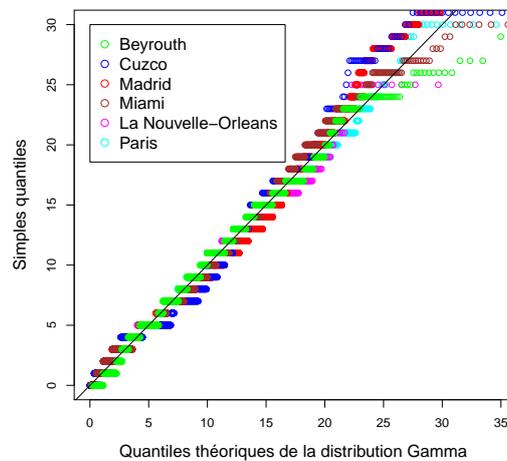


FIGURE 3.16 – Diagramme quantile-quantile de la distribution des degrés contre les distributions Gamma.

implique une présence plus importante de cliques, permettant la création de chemins de communication plus courts.

Les réseaux de type "petit monde", qui modélisent plusieurs graphes d'interaction, présentent généralement un coefficient de clustering élevé. Nous savons que les propriétés de ces réseaux ne sont pas vérifiées pour des réseaux ayant de fortes contraintes géographiques [CS04b], tels que les réseaux rues-intersections, formés uniquement par les éléments de l'infrastructure routière. La figure 3.17(a) représente la moyenne du coefficient de clustering pour chacun des six scénarios, tandis que la figure 3.17(b) représente les courbes de fréquences cumulées (CDF) associées. Bien que les graphes soient très partitionnés, nous pouvons observer que la plupart des composantes connexes semblent avoir un fort coefficient de clustering. Le type de graphe que nous générons repose sur des contraintes géographiques qui sont biaisées par la forte densité de capteurs déployés sur chaque intersection. En comparaison avec les graphes rues-intersections, cela semble nous indiquer que les graphes que nous générons possèdent la propriété des petits mondes, qui imposent que le coefficient de clustering soit indépendant du nombre de nœuds globaux. De plus, cette propriété impose généralement une faible distance moyenne entre deux nœuds du graphe concerné. Sur chacun de nos réseaux, le diamètre des composantes connexes (figures 3.23 plus bas) est, en moyenne, inférieur au logarithme du nombre total de nœuds des graphes. Cette comparaison est généralement utilisée pour attester d'une faiblesse "suffisante". Le diamètre moyen étant nécessairement supérieur à la distance moyenne entre deux sommets, nous avons finalement la confirmation que les graphes générés possèdent la propriété des petits mondes. Ceci traduit notamment un nombre de chemins de communication plus nombreux et plus courts.

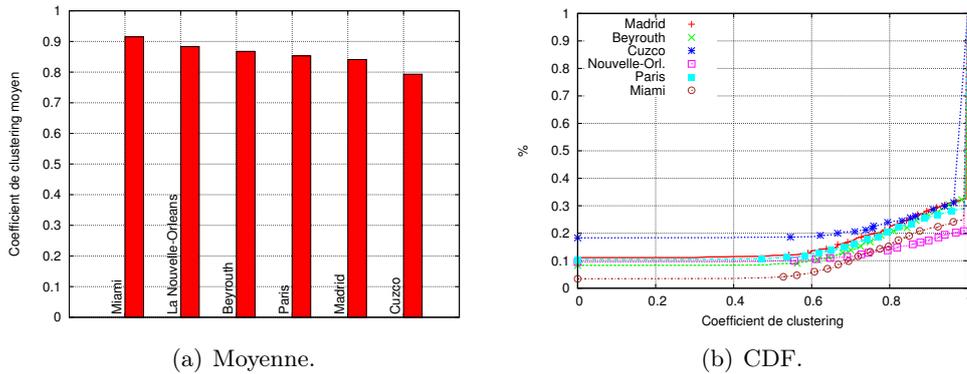


FIGURE 3.17 – Coefficient de clustering.

### 3. 5. 3 Rapport avec la création de graphes aléatoires

Les degrés tendent à suivre une distribution gamma : aucun modèle de graphe aléatoire de l'état de l'art ne s'inscrit vraiment dans ce type de réseaux. Les modèles de Gilbert [Gil59] produisent une distribution des degrés qui correspond à une distribution binomiale. Le modèle de Erdős and Rényi [ER59] génère des graphes dont la distribution des degrés suit une distribution de Poisson, tout comme le modèle de graphes géométriques aléatoires [Pen03], qui est classiquement utilisé pour générer des réseaux sans fil aléatoires. La méthode d'attachement préférentielle proposée par Barabasi et Albert [BA99], tout comme le modèle de Watts et Strogatz [WS98], produisent tout deux des réseaux sans échelle dont la distribution des degrés suivent une loi de puissance. Le tableau 3.3 récapitule les propriétés de ces principaux modèles de graphes aléatoires.

Référence	Distribution des degrés	Coefficient de clustering
Gilbert [Gil59]	Loi binomiale	Élevé
Erdős and Rényi [ER59]	Loi de Poisson	Faible
Watts et Strogatz [WS98]	Loi de puissance	Élevé
Barabasi et Albert [BA99]	Loi de puissance	Dépend du nombre de nœuds ( $N^{-.75}$ )
Graphe géométrique aléatoire (p. ex., Penrose [Pen03])	Loi de Poisson	Dépend du seuil de distance entre deux nœuds

TABLE 3.3 – Comparaison de quelques modèles de graphes aléatoires.

Générer des graphes qui correspondent à nos déploiements est possible. L'expérimentateur doit d'abord décider du type de la ville qu'il souhaite générer et décider des paramètres de forme et d'échelle de la distribution gamma. Les valeurs de forme plus petites permettent d'obtenir une distribution avec de faibles degrés, et donc de modéliser des villes dans lesquelles les intersections sont éloignées les unes des autres. La paramètre d'échelle définit la hauteur du pic et donc modélise l'uniformité des degrés. Il représente dans une certaine mesure la régularité de la distance entre les intersections. Utiliser des modèles tels que la méthode de Molloy et Reed [MR95] permet d'utiliser une distribution des degrés arbitraire : il est possible de définir une séquence des degrés correspondant, par exemple, à une distribution gamma voulue, proche des paramètres observés d'une ville. Le problème est que les graphes générés ne sont pas géométriques (c-à-d. sans dimensions) : chaque nœud a la possibilité

de communiquer avec n'importe quel nœud du réseau, ce qui tend à créer moins de composantes connexes que ce qui est observé par la suite (section 3. 6).

Une solution serait d'adapter la méthode de Molloy et Reed au contexte des graphes géométriques, en opposition aux graphes traditionnels. En plus de se baser sur une séquence de degrés respectant une loi gamma, chaque nœud serait défini avec un emplacement géographique. Ceci impliquerait d'imposer une composante de dimension, comme la distribution des interdistances, afin que le placement des nœuds respecte les schémas que nous pouvons décrire dans notre jeu de données.

## 3. 6 Analyse du réseau

### 3. 6. 1 Partitionnement du réseau

Un graphe peut être partitionné en une ou plusieurs *composantes connexes*. Une composante connexe est un sous-graphe dans lequel tous les couples de nœuds sont connectés par un chemin, et tel que aucun chemin n'existe entre les nœuds d'une composante et les nœuds à l'extérieur de cette composante.

Dans notre déploiement, une composante connexe modélise un groupe de nœuds qui sont connectés ensemble mais déconnectés du reste du réseau. Au sein de chaque composante connexe, des liens existent entre les nœuds et ont un poids dépendant de la qualité du lien. La figure 3.18 représente quatre intersections qui sont gérées à l'aide d'un capteur par voie et le graphe en résultant. Ce dernier est partitionné à cause d'une trop longue distance entre les groupes gauches et droits de capteurs. La capacité des composantes connexes à être liées peut également dépendre de la qualité de la connexion (p. ex., bâtiment les séparant).

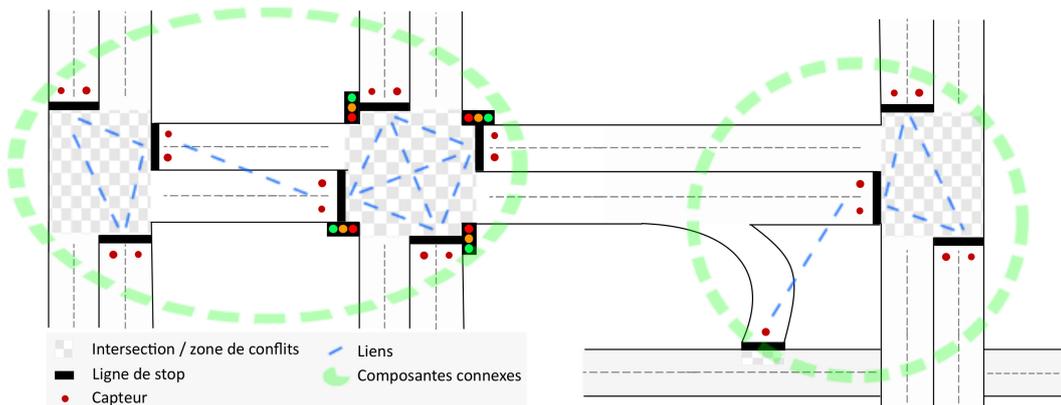


FIGURE 3.18 – Un réseau de quatre intersections gérées par un capteur par voie.

La fragmentation du réseau n'est pas un problème en soit, car les composantes peuvent être interconnectées entre elles à l'aide d'un réseau cellulaire ou à l'aide d'un réseau câblé métropolitain. Toutefois, le nombre de composantes réseau indépendantes doit rester raisonnable afin de limiter la complexité de mise en place. C'est

pourquoi nous évaluons également une stratégie afin d’interconnecter les composantes connexes proches, en section 3.7, après avoir analysé les graphes bruts.

### 3.6.2 Composantes connexes

La distribution des degrés représente la connectivité locale. Pour évaluer la connectivité globale des réseaux, nous analysons le nombre de composantes connexes qui forment ces réseaux. Cela signifie que chacune de ces aires est totalement autonomes et déconnectées du centre de contrôle, ou qu’il est nécessaire de déployer des nœuds ou des liens uniquement à des fins de connectivité. Le gestionnaire du réseau peut ainsi choisir de placer, dans chaque sous-réseau, une passerelle à travers un réseau cellulaire ou via un réseau filaire métropolitain, et/ou de déployer des capteurs supplémentaires. Nous étudions l’effet d’une telle extension dans la section 3.7.

La figure 3.19 montre le nombre de composantes connexes dans les différents réseaux. Ce nombre dépend directement de la dimension de ces réseaux ainsi que du nombre de nœuds. Nous pouvons voir que Paris a plus de 5 500 composantes connexes pour un total d’environ 29 000 nœuds, ce qui reflète un réseau très fragmenté mais montre également une réelle difficulté dans la mise en œuvre de protocoles qui nécessiteraient l’ensemble du réseau.

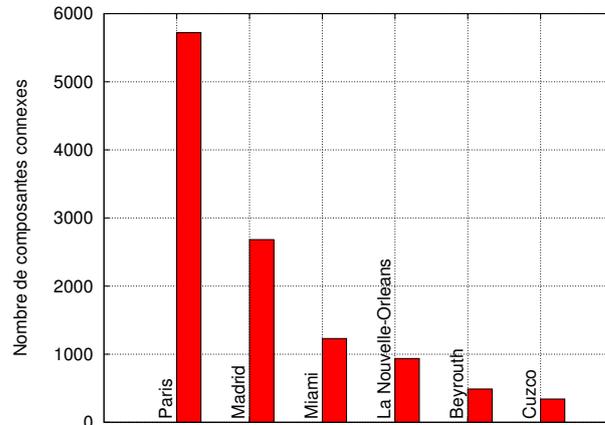


FIGURE 3.19 – Nombre de composantes connexes.

La figure 3.20 montre le pourcentage de composantes connexes qui sont uniquement composées d’un seul nœud, c’est à dire le nombre de capteurs qui sont trop éloignées pour être directement connectés au reste du réseau par l’intermédiaire de la même technologie sans fil. Cette proportion augmente à mesure que la densité de nœuds diminue (figure 3.9(b)). Cuzco, par exemple, a environ 25% de composantes connexes composées d’un seul nœud, tandis que Paris en a environ 12,5%, ce qui représente 750 nœuds.

La figure 3.21 montre la CDF de la distance entre une composante et ses composantes voisines. Pour identifier les composantes les plus proches, nous calculons d’abord les coordonnées du centre de gravité de chaque composante connexe. Cela produit un ensemble de points dans le plan. Nous construisons ensuite le diagramme

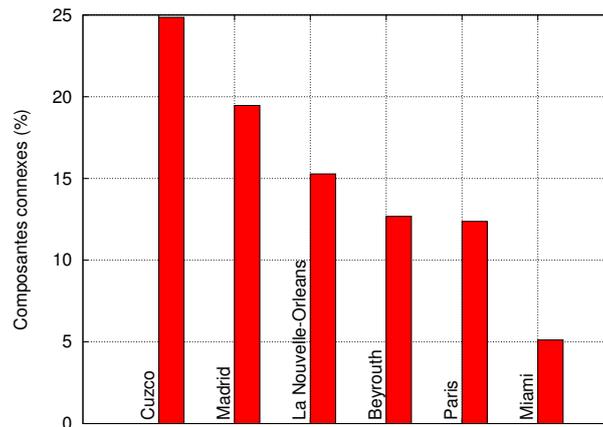


FIGURE 3.20 – Pourcentage de nœuds isolés.

de Voronoï de cet ensemble de points. Un diagramme de Voronoï sépare le plan en zones centrées sur chaque nœud. Une zone est composée de tous les points qui sont plus proches du nœud central que n'importe quel autre nœud. Nous considérons donc que deux éléments sont voisins si leurs cellules de Voronoï ont une frontière commune, comme le représente l'exemple donné sur la figure 3.22. La triangulation de Delaunay est le dual géométrique du diagramme de Voronoï, tout aussi représentatif. Il s'agit du graphe associé, sur lequel deux points sont reliés par une arête si leurs cellules sont voisines dans le diagramme de Voronoï. La figure 3.21 montre que quelques éléments sont très loin du reste du réseau, mais que la plupart des composantes sont relativement proches les unes des autres. Cela indique que la réduction du nombre de composantes par l'insertion de relais devrait être efficace.

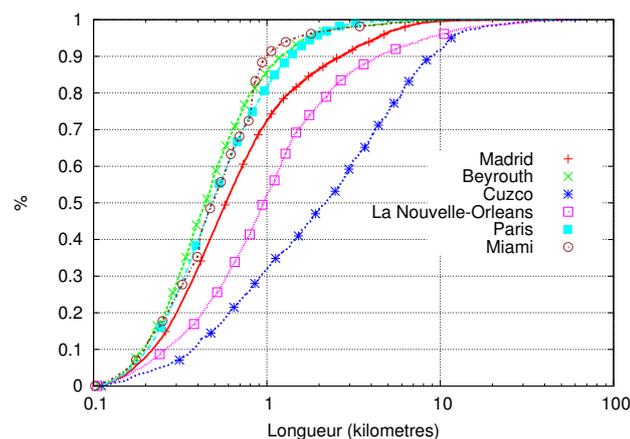


FIGURE 3.21 – CDF de la distance entre les composantes connexes proches.

### 3. 6. 3 A l'intérieur des composantes connexes

La figure 3.23(a) représente le diamètre moyen des composantes connexes de chaque réseau. Le diamètre est la longueur du plus long des chemins les plus courts entre les couples de nœuds qui appartiennent à la même composante. Il est exprimé en

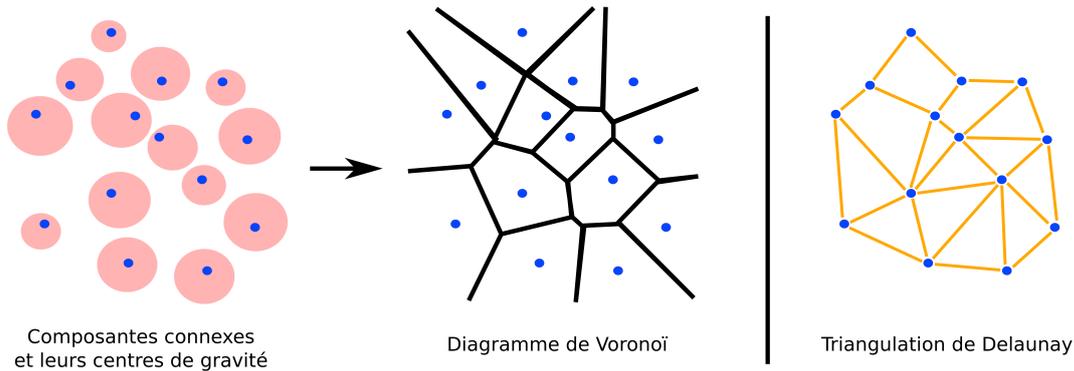


FIGURE 3.22 – Représentation d’un diagramme de Voronoï et d’une triangulation de Delaunay.

nombre de sauts. Cette distance influence le délai de transmission entre des paires de nœuds à l’intérieur d’une composante connexe. Il dépend de la taille des composantes et, dans une moindre mesure, de la densité des nœuds. Nous pouvons voir que ce diamètre reste très faible, essentiellement en raison de la présence de plusieurs composantes de petites tailles. La figure 3.23(b), qui représente la CDF des diamètres, confirme ce résultat. La distribution des distances correspondante est illustrée sur la figure 3.23(c), pour l’exemple de Paris (les distributions qui correspondent aux autres réseaux étant similaires). Les réseaux sont donc composés principalement de composantes connexes de petites tailles et de quelques unes plus grandes.

Regardons à présent la *composante connexe maximale*, qui est la composante connexe qui contient le plus grand nombre de nœuds. La figure 3.24(a) représente le nombre de nœuds qui appartiennent à cette composante maximale et varie de 33 nœuds (La Nouvelle-Orléans) à plus de 130 nœuds (Miami). Les nœuds qui appartiennent à la même composante connexe peuvent être considérés comme appartenant au même domaine de diffusion, ce qui donne une indication sur le coût des émissions et sur le nombre de nœuds qui peut être atteint par des paquets de contrôle (ARP, protocoles de routage, etc.).

La figure 3.24(b) montre la CDF des distances (en nombre de sauts) qui séparent les couples de nœuds dans cette composante maximale. Elle donne une indication sur les délais. Nous pouvons voir ici que les distributions vont des composantes de faible diamètre (environ 4 sauts) aux composantes plus grandes (10 sauts), et que ces distributions ne suivent pas toujours la tendance définie par la taille de la composante, ou de la densité moyenne. Madrid, par exemple, est moins dense que Paris (fig. 3.9(b)) mais les nœuds de sa composante maximale sont séparés par des chemins plus courts, pour des quantités comparables. La réponse réside dans la distribution des inter-distances (fig. 3.10) : le mode de la distribution de Paris est situé à une distance supérieure à celle du premier mode de la distribution de Madrid. Cela indique que la densité des intersections est plus élevée dans le centre de Madrid qu’à Paris.

La figure 3.24(c) montre la CDF des longueurs d’arêtes. Ce paramètre est important pour l’évaluation de l’atténuation sur les liaisons sans fil et donc de la qualité des liens ou du nombre attendu de transmissions. La distribution est globalement

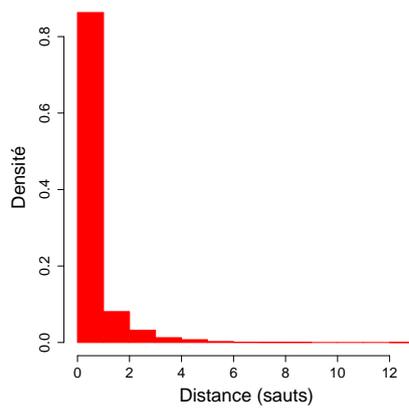
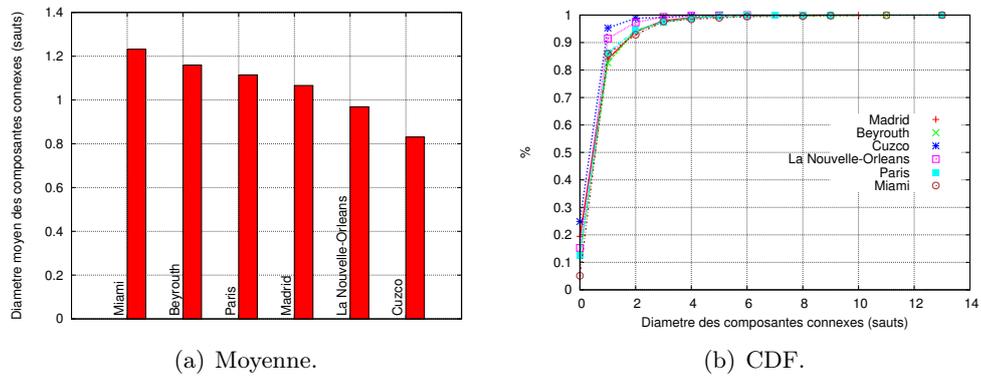


FIGURE 3.23 – Diamètre des composantes connexes.

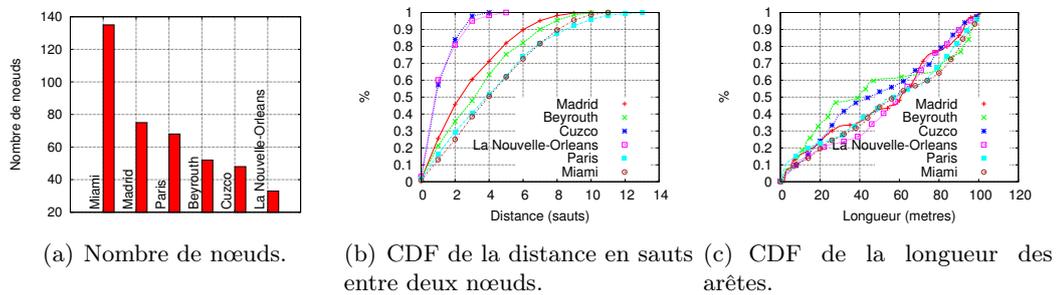


FIGURE 3.24 – Analyse de la composante maximale.

uniforme sur chacun des réseaux, la CDF étant presque linéaire pour chaque cas (les différences constatées entre 20 et 60 mètres provenant des spécificités architectures des villes).

### 3. 6. 4 Analyse de fiabilité

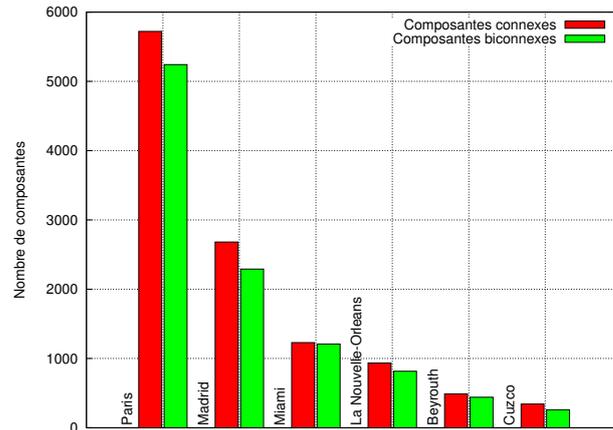


FIGURE 3.25 – Nombre de composantes connexes et biconnexes.

La figure 3.25 compare le nombre de composantes connexes (barres rouges) avec le nombre de *composantes biconnexes* (barres vertes) sur chacun des réseaux. Une composante biconnexe est une composante connexe dans laquelle il existe au moins deux chemins entre chaque couple de nœuds. Elle reflète la proportion de sous-réseaux qui ne peuvent tolérer aucune défaillance de nœud. Les valeurs montrent qu'il y a seulement une petite proportion des sous-réseaux qui présentent une telle faiblesse structurelle et que peu de déploiements supplémentaires seront nécessaires pour se conformer au critère de fiabilité classique N-1 (c.-à-d., le réseau doit rester exploitable même lorsqu'un élément disparaît inopinément).

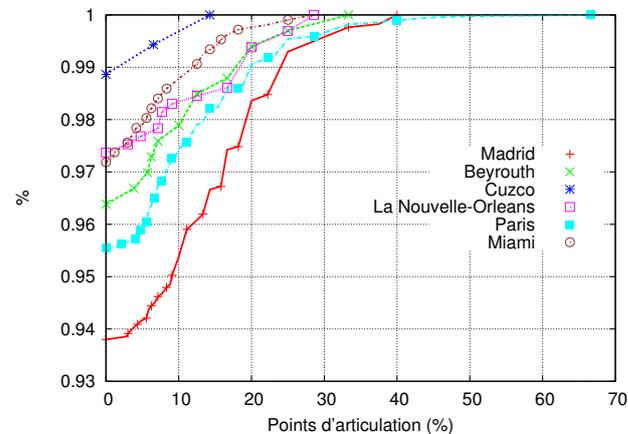


FIGURE 3.26 – CDF du nombre de points d'articulation (pour chaque composante constituée d'au moins trois nœuds).

La figure 3.26 représente la CDF du nombre de composantes qui présentent un pourcentage donné des nœuds qui sont des *points d'articulation* (représenté en abscisse). Un point d'articulation est un nœud dont la suppression déconnecte la composante à laquelle il appartient, ce qui augmente le nombre de composantes connexes. Cette définition implique qu'un point d'articulation ne peut exister que dans des composantes connexes formées par au moins trois nœuds. Le graphique ne considère donc que ces composantes connexes.

Tout d'abord, les courbes sont assez redondantes. Dans le pire des cas (Madrid), près de 94 % des composantes n'ont pas de point d'articulation. Dans le pire des cas, une composante connexe a 50 % de ses nœuds qui sont essentiels pour la connectivité, ce qui correspond à une chaîne de nœuds. Un réseau comme Paris, par exemple, tend à avoir un grand nombre de points d'articulation, car les zones peu denses en périphérie (c.-à-d., banlieues) sont nombreuses. Madrid a les mêmes caractéristiques que la ville de Paris, sans les banlieues dispersées, mais avec plusieurs zones de forte densité dans le centre-ville. Dans ce cas, la hausse du taux de points d'articulation n'est pas aussi soudaine.

### 3. 7 Amélioration du déploiement

L'analyse présentée dans la section précédente a été menée sur des graphes "bruts", créés en positionnant seulement les capteurs qui ont un rôle de surveillance du trafic. Comme aucun effort n'a été fait pour améliorer la connectivité, ces graphiques sont composés de plusieurs composantes connexes. Un opérateur désireux d'acquérir des données ou de diffuser des politiques sur l'ensemble de son réseau doit interconnecter ces composantes.

Dans cette section, nous examinons l'effet d'une telle stratégie d'interconnexion qui repose sur l'insertion de nœuds de relais que nous supposons posséder les mêmes caractéristiques radio que les nœuds capteurs. Ces nœuds relais sont disposés de manière à fusionner deux composantes connexes. Nous définissons la distance qui sépare deux composantes connexes arbitraires comme étant le minimum de la distance entre les couples de nœuds qui appartiennent à chaque composante. Selon cette distance, nous avons besoin d'un ou de plusieurs relais intermédiaires afin de fusionner les deux sous-graphes. En ayant connaissance du rayon de transmission d'un nœud, nous plaçons, entre deux composantes connexes voisines, une chaîne de nœuds suffisamment grande pour couvrir l'espace à remplir et connecter ces deux composantes.

Nous supposons que l'opérateur impose une limite sur le nombre maximum de nœuds intermédiaires qui pourraient être déployés à des fins d'interconnexion entre deux composantes. Nous étudions les cas allant de 1 à 10 relais. Par exemple, si une chaîne de nœuds suffit à connecter deux composantes connexes, nous l'ajoutons. Nous déterminons si cette chaîne est suffisante en nous basant sur le modèle de propagation défini en section 3. 3. 3, et séparons chaque nœud de la chaîne à ajouter d'une distance égale à la moitié de leur portée maximale, afin que les signaux transmis ne soient pas complétement atténués par la distance. Une valeur de 10 est très peu probable, car il

en résulterait des chaînes de 10 nœuds pour interconnecter les composantes, sachant que la défaillance de l'un de ces nœuds se traduirait par un partitionnement.

La figure 3.27 montre la distribution des degrés pour le réseau de Paris avec cette amélioration. Tous les scénarios passent par la même évolution. Nous pouvons voir que le rapprochement avec la distribution gamma est toujours valide. Le tableau 3.4 montre les nouvelles valeurs des paramètres des distributions gamma pour les six scénarios représentatifs quand 4 relais au plus sont ajoutés. En comparant ces valeurs avec le tableau 3.2, nous pouvons remarquer globalement une diminution du paramètre d'échelle et une augmentation du paramètre de forme. Ceci indique que les réseaux résultants sont moins uniformes en raison de la présence de chaînes de nœuds. Sans surprises, le nombre de nœuds connectés étant plus important, la distribution se déplace vers la droite : le degré moyen augmente.

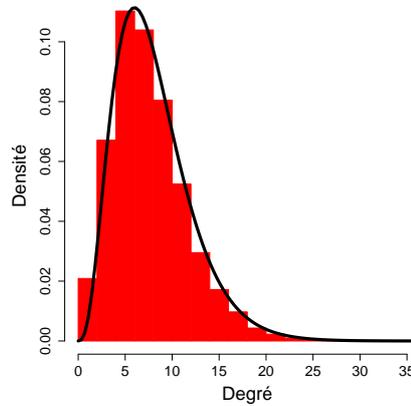
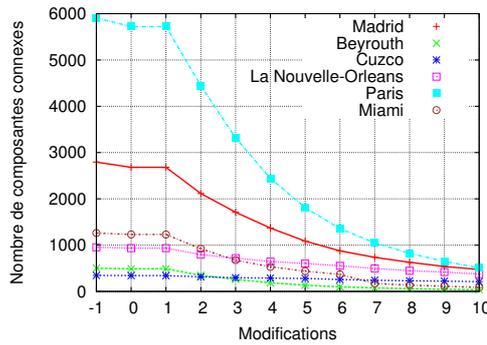


FIGURE 3.27 – Paris. Distributions empiriques et ajustées lors de l'ajout de 4 relais (au plus) entre les composantes connexes.

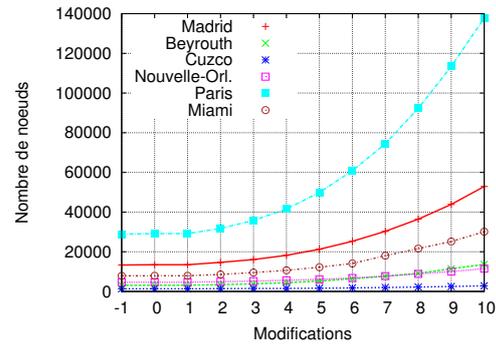
Ville	Forme ( $k$ )	Échelle ( $\theta$ )
Beyrouth	3.463977	3.030837
Cuzco	1.222276	5.911105
Madrid	2.321423	3.584129
Miami	3.682748	2.584144
La Nouvelle-Orléans	2.929175	2.5145
Paris	3.955912	2.024664

TABLE 3.4 – Paramètres des distributions gamma ajustées dans un déploiement avec amélioration de la connectivité.

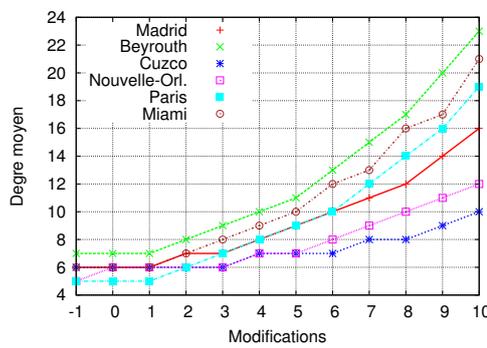
La figure 3.28(a) représente l'évolution du nombre de composantes connexes. La valeur  $-1$  présent sur l'axe des abscisses correspond au cas où les points d'articulations sont supprimés du graphe (voir section 3.6.4). Nous pouvons voir qu'insérer un seul relai n'a qu'un faible impact, tandis qu'augmenter ce seuil à 2 ou 3 nous permet d'obtenir un nombre de composantes connexes significativement plus bas. Le tout semble converger à des valeurs comparables, aux alentours de 200 composantes connexes.



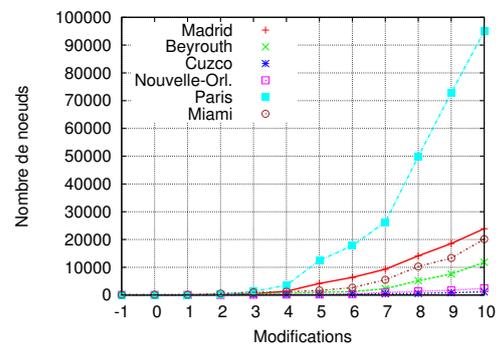
(a) Nombre de composantes connexes.



(b) Nombre de nœuds.



(c) Degré moyen.



(d) Pourcentage de nœuds dans la composante maximale.

FIGURE 3.28 – Amélioration de la connectivité des réseaux.

La figure 3.28(b) représente l'évolution du nombre de nœuds déployés en fonction de la valeur du seuil. Nous pouvons remarquer que la valeur tend à augmenter plus vite dans les réseaux dispersés, alors que la réduction du nombre de composantes connexes ralentit. Dans le cas de Paris – le réseau avec le plus de composantes connexes – nous avons besoin d'ajouter près de 60 000 nœuds pour obtenir moins de 1 000 composantes connexes.

La figure 3.28(c) représente l'évolution du degré moyen dans le graphe. Elle montre que l'effet d'ajout de nouveaux nœuds sur le degré varie selon la densité initiale du réseau (voir figure 3.9(b)). Les réseaux avec une densité initiale plus élevée (Beyrouth, Miami, Paris) voient une plus forte augmentation dans le degré des nœuds que d'autres.

Finalement, la figure 3.28(d) montre l'évolution du nombre de nœuds qui appartiennent à la composante maximale. Ce graphique montre que, même si l'amélioration n'est pas la même pour toutes les villes, cette composante est capable de recueillir jusqu'à 90 % des nœuds.

Évidemment, la stratégie d'amélioration détaillée ici n'est qu'une possibilité, la plus simple si nous souhaitons garder un réseau homogène de capteurs. Le fait d'ajouter une chaîne de nœuds pour lier deux composantes connexes rend la structure fragile, mais cela montre que cette opération est faisable facilement et sans beau-

coup de contraintes. Utiliser des relais longue portée serait également une solution, leur nombre devant être de  $N - 1$  relais au minimum pour lier les  $N$  composantes connexes de notre réseau (égal au nombre minimum d'arêtes dans un graphe connexe à  $N$  nœuds). Ce type de réseau serait capable d'être consolidé avec un réseau véhiculaire, comportant également ses propres contraintes [NF13].

## 3.8 Analyse des autres stratégies de déploiement

Dans les sections précédentes, nous avons fait le choix d'étudier une stratégie de déploiement où les nœuds sont positionnés sur les voies du réseau routier. Dans cette section, nous nous intéressons aux deux autres stratégies décrites en 3.3.1, afin de montrer les différences existantes avec notre étude principale.

### 3.8.1 Degré des nœuds

La figure 3.29 représente le degré moyen des nœuds lorsqu'ils sont déployés sur chaque route. En comparaison au premier déploiement étudié, les degrés sont toujours relativement proches (entre 2 et 6 pour la majorité des cas), mais logiquement moins importants. En effet, le fait de réduire le nombre de nœuds déployés sur une intersection réduit le nombre de voisins que chacun peut avoir. Ceci se ressent directement sur nos résultats. En termes de performances réseaux, les degrés moyens indiqués restent suffisamment élevés pour offrir un niveau de redondance correct.

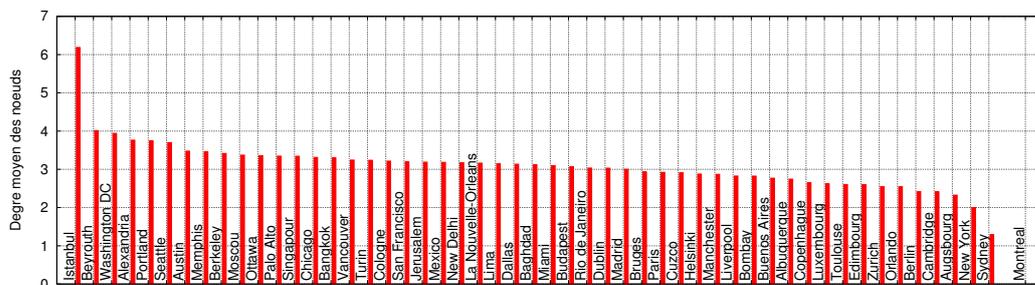


FIGURE 3.29 – Degré moyen sur notre jeu de données complet, pour un déploiement à un nœud par route.

La figure 3.30 représente le degré moyen des nœuds lorsqu'ils sont déployés seuls sur chaque intersection. Les remarques sont les mêmes que pour la figure 3.29 : la chute du degré moyen, situé entre 0 et 2 pour la majorité, est liée à la baisse du nombre de nœuds déployés sur chaque intersection. Dans le cas présent, un seul nœud est déployé au centre de chaque intersection, le degré moyen est donc particulièrement bas, voire nul pour la plupart des cartes. Cela veut dire que pour de nombreux scénarios, un bon nombre de nœuds sont isolés et les intersections ne peuvent pas directement communiquer entre elles. Dans les cas précédents, cette communication était plus facilement accessible, car les nœuds étaient présents en plus grand nombre

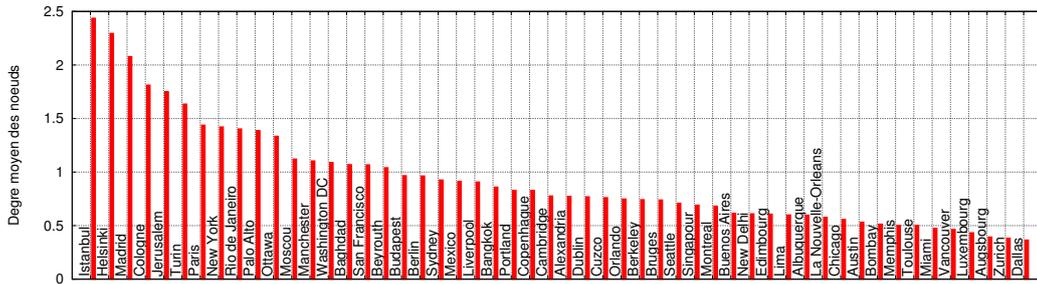
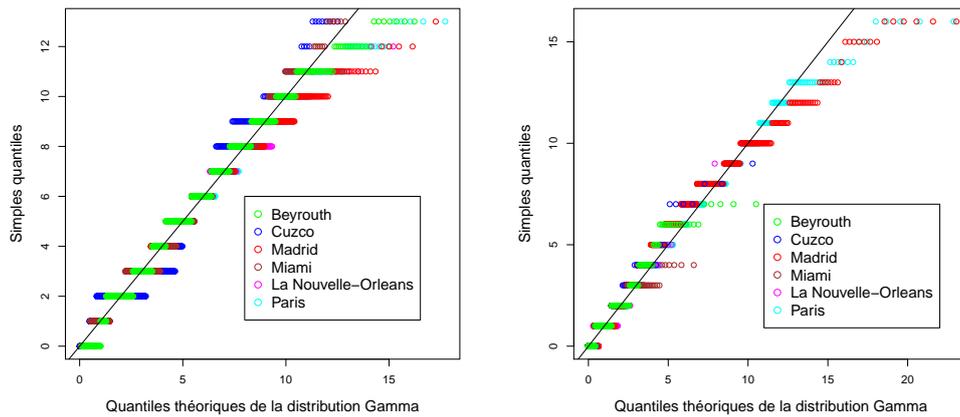


FIGURE 3.30 – Degré moyen sur notre jeu de données complet, pour un déploiement à un nœud par intersection.

et répartis autour de l’intersection, et non en son centre (ce qui augmente notamment de plusieurs mètres la portée de communication).

Les figures en 3.31 représentent deux diagrammes quantile-quantile, qui comparent les distributions de degrés empiriques et ajustées sur une distribution gamma. La figure 3.31(a) représente le cas d’un déploiement de nœuds sur routes. La figure 3.31(b) représente le cas d’un déploiement de nœuds sur intersections. Sur ces graphiques, plus proches les points sont de la ligne diagonale et meilleure est la correspondance. Nous remarquons qu’une distribution gamma est toujours pertinente pour décrire les distributions de degrés obtenues. Toutefois, il semblerait que la différence soit plus marquée pour le scénario avec un déploiement utilisant un nœud par route, et encore plus marquée pour le scénario avec un déploiement utilisant un nœud par intersection. En d’autres termes, la correspondance avec une distribution gamma semble être meilleure à mesure que le nombre de nœuds déployé sur une intersection est important.



(a) Déploiement avec un nœud par route. (b) Déploiement avec un nœud par intersection.

FIGURE 3.31 – Diagrammes quantile-quantile de la distribution des degrés contre les distributions Gamma.

### 3. 8. 2 Partitionnement

Les figures en 3.32 représentent le nombre de composantes connexes pour chacun des deux déploiements. Qu'il s'agisse d'un déploiement sur routes (figure 3.32(a)) ou d'un déploiement avec un nœud par intersection (figure 3.32(b)), les réseaux décrits restent fortement partitionnés. Visiblement, nous remarquons que le nombre de composantes connexes est directement fonction du nombre de nœuds déployés par intersection.

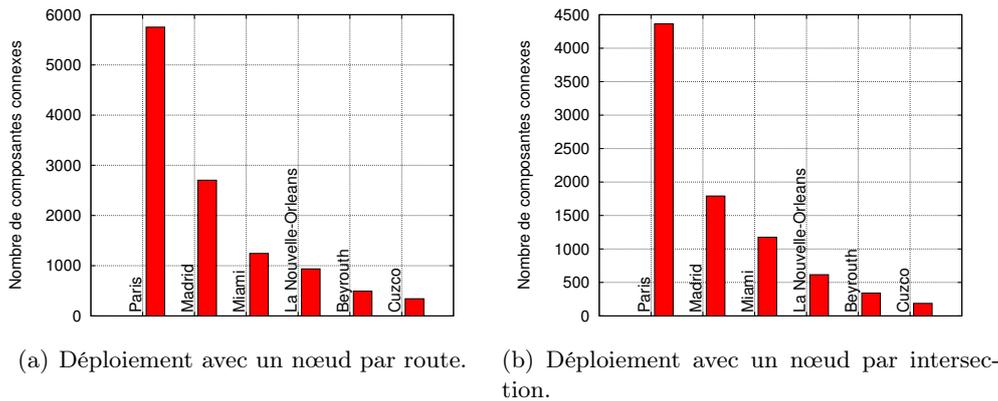


FIGURE 3.32 – Nombre de composantes connexes.

Les figures en 3.33 représentent le coefficient de clustering moyen pour chacun des six scénarios étudiés. En comparant ces valeurs à celles obtenues pour le premier déploiement étudié (section 3. 5. 2), nous remarquons que ce coefficient a tendance à s'affaiblir à mesure que le nombre de nœuds déployés sur une intersection décroît. Pour le cas de Paris, par exemple, notre premier déploiement nous donne un coefficient de clustering de 0,85. En déployant uniquement des capteurs sur routes, ce coefficient passe à 0,7. En déployant uniquement un capteur par intersection, il descend à 0,25. Ce constat nous permet de confirmer les remarques formulées par [CS04b], qui indique que les réseaux soumis à fortes contraintes géographiques ont généralement un coefficient de clustering faible. Dans les cas étudiés, plus le nombre de capteurs est faible, plus le graphe obtenu correspond à un graphe de type rues-intersections, où la diversité des chemins de communication et la largeur des composantes connexes sont moins importantes. Notons que les nœuds générés par la stratégie de déploiement avec un capteur par intersection sont les mêmes que ceux des graphes de type rues-intersection.

### 3. 8. 3 Largeur des composantes connexes

En se fiant à ces analyses, nous pouvons affirmer que les conclusions faites pour le premier déploiement étudié sont similaires pour ces deux nouveaux cas. D'une part, les réseaux obtenus sont au moins autant partitionnés, si ce n'est davantage à cause du nombre de nœuds moins important. D'autre part, la mise en place d'un système de transport intelligent se reposant sur ces déploiements reste intéressant sur des zones urbaines importantes (p. ex., centre ville).

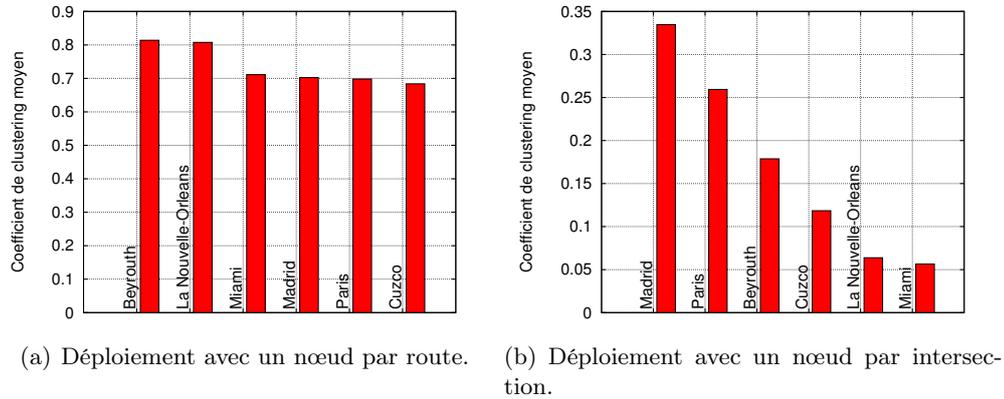


FIGURE 3.33 – Coefficient de clustering moyen.

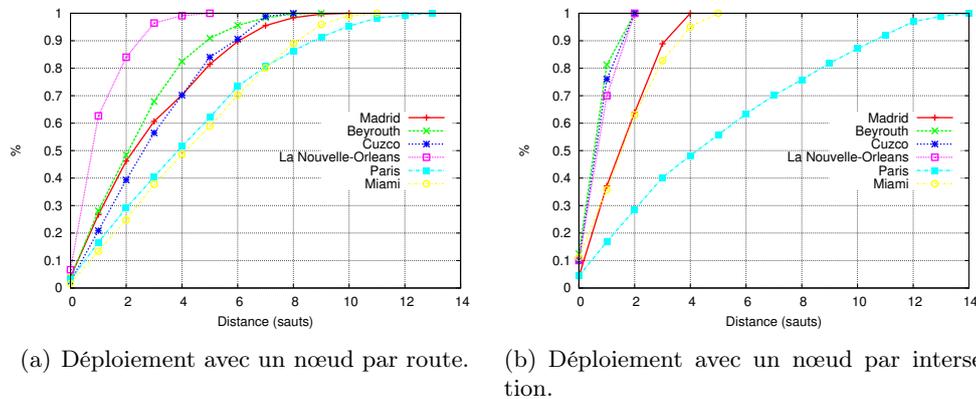


FIGURE 3.34 – CDF de la distance séparant tous les couples de nœuds de la composante connexe maximale.

Les graphiques représentés en 3.34 décrivent la CDF du nombre de sauts séparant les nœuds de la composante connexe maximale, pour chacun des deux déploiements. Pour le cas d'un déploiement sur route, la figure 3.34(a) nous montre que la largeur de la composante connexe maximale est d'au moins 13 sauts pour le cas de Paris. Pour le cas d'un déploiement d'un nœud sur chaque intersection, cette largeur est d'au moins 14 sauts pour la même ville. Autrement dit, dans ce dernier cas, la composante connexe maximale peut couvrir au moins 14 intersections. Cela représente une zone urbaine plutôt intéressante pour appliquer – par exemple – un algorithme de contrôle des feux de circulation. Nous remarquons également que cette distance est plus importante à mesure que le nombre de nœuds diminue. Ceci a un lien direct avec le degré moyen des nœuds : positionner un nœud sur chaque intersection offre moins de diversité dans les chemins de communication existants, ce qui augmente naturellement le nombre de sauts entre deux nœuds.

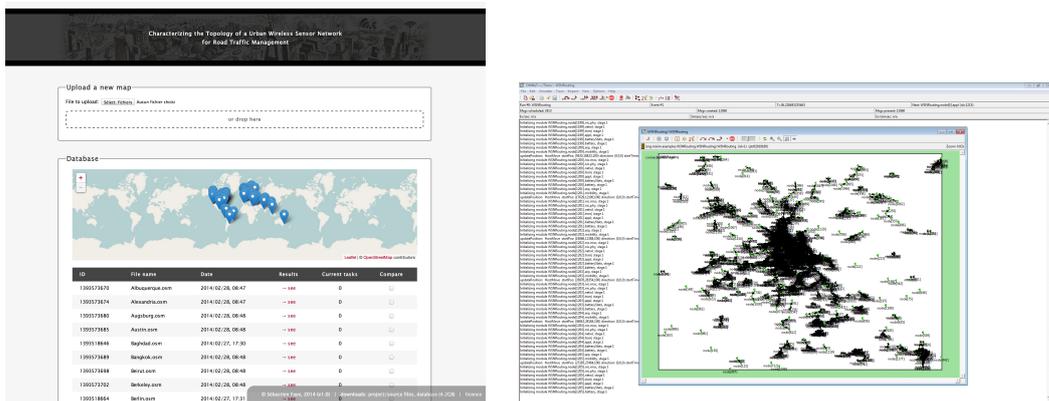
### 3.9 Outils développés

L'ensemble des résultats présentés dans ce chapitre ont été générés par un outil que nous avons développé sous licence LGPL. Cet outil, tout comme l'ensemble de nos résultats sur plus de 52 scénarios – représentant des villes et zones géographiques mondiales – sont disponibles en ligne, à l'adresse <http://g.sfaye.com/> [Fay14a].

Les scripts de génération des graphes, appelant les différents outils en séquence avec différents paramètres configurables, sont également disponibles au travers d'une interface web et pour le téléchargement à la même adresse. Il est ainsi possible de mettre en ligne sa propre carte et de paramétrer, pour chaque scénario, une méthode de déploiement (dont des cas non abordés dans ce chapitre), un modèle de propagation radio ou une distance fixe, ou encore de définir les intersections destinées à accueillir des capteurs (à feux de circulation ou toutes).

Notons qu'il est également possible de générer les modèles OMNeT++ [VH08] (côté réseau) et SUMO (côté transport) correspondant à chaque scénario. Ces modèles peuvent servir de base à des simulations, qui permettraient aux ingénieurs d'évaluer et de comparer des protocoles et des algorithmes réseaux, par exemple en utilisant le framework MiXiM.

Enfin, soulignons que ce logiciel a été entièrement développé en *PHP* et en *Python* et se base notamment sur la librairie NetworkX<sup>5</sup> [HSS08] afin de traiter les aspects relatifs aux graphes. Les figures 3.35 sont deux captures d'écran : la première est l'interface principale de notre plateforme (figure 3.35(a)), la deuxième est un exemple de génération de modèle pour OMNeT++ (exemple pris pour la ville de Turin, figure 3.35(b)).



(a) Interface de la plateforme

(b) Génération de modèles OMNeT++ (exemple de Turin)

FIGURE 3.35 – Aperçu de notre outil, développé sous licence LGPL.

5. <https://networkx.github.io/>

## 3. 10 Conclusion

Dans ce chapitre, nous avons examiné une stratégie afin de déployer un réseau de capteurs au niveau des intersections de plusieurs villes. Nous avons présenté une méthode de génération de graphes qui est basée sur des contraintes opérationnelles. Nous montrons notamment que les modèles classiques de graphes aléatoires ne modélisent pas ces réseaux, dont la distribution des degrés correspond à une distribution gamma. Nous montrons également que le graphe résultant est fortement divisé et comprend jusqu'à 25 % de nœuds isolés. Cependant, le réseau résultant présente tout de même un bon niveau de redondance. Ensuite, le diamètre moyen des composantes connexes est faible, mais peut s'élever une valeur permettant une bonne couverture. Enfin, en regardant les distances entre les composantes connexes proches, nous décrivons une méthode pour augmenter la connectivité du réseau.

Dans certains cas abordés, il est simplement inconcevable d'interconnecter l'ensemble des nœuds présents sur le terrain à une entité centrale. Créer plusieurs couches hiérarchiques permettrait de résoudre en partie le problème, mais ce serait au détriment de la réactivité et en échange de moyens coûteux. Fonctionner de manière distribuée et s'affranchir des contraintes filaires est idéal, car mis à part des réseaux relais, les contraintes sont moins nombreuses. Toutefois, il est difficile d'imaginer une application s'exécutant sur des réseaux aussi partitionnés que ceux abordés dans ce chapitre.

L'effet sur les différents protocoles réseau et algorithmes reste à évaluer, par exemple via des simulations. Cependant, les conclusions que nous tirons devraient aider à la sélection des protocoles les plus appropriés pour cette catégorie de scénarios.

Dans les chapitres suivants, nous nous intéresserons à des problématiques plus locales, au travers de la gestion des feux de circulation. Nous avons vu au cours de ce chapitre que le déploiement d'un réseau distribué, tel que nous le concevons, est possible et permet de couvrir *a priori* une zone urbaine suffisamment grande pour couvrir plusieurs intersections. Notre but est à présent de développer des algorithmes par dessus ce type de zone, et découvrir le fonctionnement de ce type de système de transport et ses bénéfices.



## CHAPITRE 4

---

### Contrôle des feux de circulation distribué sur plusieurs intersections

---

#### Gestion d'une zone urbaine ciblée

Au chapitre précédent, nous avons proposé une méthode qui permet de déployer des capteurs sur une large zone urbaine. Dans ce chapitre, nous nous intéressons aux applications qu'il est possible d'exécuter par dessus ce type de réseau. Si les systèmes de gestion sont nombreux, à commencer par ceux s'occupant du stationnement, nous nous tournons plutôt du côté de la gestion des feux de circulation, dont les enjeux – introduits notamment au chapitre 2 – sont cruciaux.

Outre les temps de déplacement, la perte de temps dans la circulation augmente la pollution et le bruit, ce qui en fait un élément clé en zone urbaine. De plus, les embouteillages ont tendance à s'étendre depuis leur point de formation. Une gestion adaptative des feux de circulation, à l'aide d'un système distribué tel que nous venons de le décrire, pourrait empêcher cette expansion et aider à résorber ou à alléger une telle situation plus rapidement.

Ce chapitre présente et évalue TAPIOCA (*distribuTed and AdaPtive Intersec-tiOns Control Algorithm*). Cet algorithme utilise un réseau distribué (p. ex., un réseau de capteurs sans fil) afin d'obtenir les données de répartition des véhicules et de calculer et appliquer une politique de gestion des feux de circulation.

La section 4.1 présente les algorithmes de gestion des feux s'appliquant sur les déploiements décrits au chapitre précédent. Nous présentons une architecture hiérarchique possible au niveau d'une intersection en section 4.2. Nous détaillons ensuite, en section 4.3, un premier algorithme de gestion des feux de circulation fonctionnant sur une intersection isolée. Cet algorithme sélectionne dynamiquement les mouvements et les temps de feux vert afin de réduire le temps moyen d'attente des automobilistes sans engendrer de famine. Dans la section 4.4, nous présentons une généralisation de cet algorithme qui permet aux intersections proches de collaborer et de se synchroniser afin de créer des vagues vertes. L'algorithme général, TAPIOCA, est présenté dans la section 4.5. Les résultats de simulation sont obtenus avec le

simulateur SUMO et sont présentés en section 4.6. Ils comparent le temps moyen d'attente et les tailles des files d'attente générées par TAPIOCA, en comparaison aux algorithmes et méthodes similaires de l'état de l'art.

## 4.1 Algorithmes de contrôle des feux de circulation

L'utilisation de la théorie des files d'attente est une approche venant en tête naturellement lorsqu'il s'agit de modéliser les files d'attente présentes aux intersections (section 2.1.2). Les auteurs de [ANR11] abordent par exemple son utilisation, qui permet d'obtenir – à partir des données recueillies des capteurs – la taille moyenne d'une file sur une voie. Le temps de feu vert est calculé par rapport à la taille de ces files, de manière à obtenir le temps nécessaire à les traiter.

En se basant sur un déploiement de deux capteurs par voie (section 3.2), Yousef *et al.* [YAKS10] définissent un mécanisme de gestion des feux à une simple intersection en modélisant chaque mouvement comme une file d'attente M/M/1. En utilisant une matrice qui définit les mouvements en conflits, ils proposent un algorithme qui sélectionne les combinaisons de mouvements compatibles qui possèdent le plus grand nombre de véhicules sur leurs files. Ils calculent ensuite le temps de feu vert proportionnellement au nombre total de véhicules. Leur algorithme élabore ainsi un plan de feu dynamique cycle après cycle avec un nombre de phases qu'ils fixent à quatre au maximum. Ils étendent ensuite leur travail à un réseau maillé d'intersections en sélectionnant tout d'abord, lorsqu'une phase débute, les mouvements qui reçoivent le plus de véhicules en provenance d'intersections proches. Leur méthode prend en compte le temps nécessaire pour aller d'une intersection à une autre, en incluant les arrêts et les ralentissements.

Après avoir abordé un modèle distribué où les capteurs communiquent avec un contrôleur dans [TSS07], Tubaishat *et al.* proposent dans [TQSS08] une gestion des feux plus aboutie. Leur solution consiste à se baser sur un système de gain, qui correspond pour une voie au nombre de voitures entre les deux capteurs. Ainsi, chaque phase possible est évaluée par la somme des gains de ses mouvements. La phase qui est la plus chargée est donc sélectionnée en premier, jusqu'à ce que son nombre de véhicules passe en dessous d'une autre ou que le temps de feu maximum soit dépassé.

Zou *et al.* [ZYC09] se reposent sur un déploiement incorporant uniquement un capteur par direction. Ils supposent que ces capteurs sont capables de détecter les véhicules sur cinq mètres et utilisent des routes de deux voies dans leur scénario (soit 6,5 mètres de largeur). Ils sont ainsi capables de comptabiliser le nombre de voitures empruntant chacune des voies. Cela permet d'établir la durée des feux selon le nombre de véhicules par minute. Si le nombre de passages est inférieur, par exemple, à cinq véhicules par minute sur une direction, alors le feu correspondant est configuré à dix secondes. Ceci est une application de la logique floue. L'inconvénient majeur de cette solution est que le nombre de voies possibles pour une intersection est limité par la

portée de détection des capteurs. De plus, les mesures prises en compte sont basées minutes après minutes et non en fonction des phases ou des cycles.

Aucune de ces contributions n'autorise vraiment les mouvements entrant en conflit à avoir le feu vert en même temps. De plus, ils agissent généralement en créant des cycles. Il serait probablement plus intéressant d'agir phase après phase, afin de gagner en réactivité. Enfin, ils prennent uniquement en compte la taille des files d'attente, ce qui peut produire un problème d'ordonnancement bien connu : la famine. Dans ce dernier cas, le feu vert n'est jamais accordé à une file d'attente car elle n'est pas considérée prioritaire (c.-à-d., elle possède moins de véhicules que les autres). La figure 4.1, importée de l'un de nos premiers travaux [FCD12a], nous montre que jouer avec des critères de famine et des critères de file d'attente peut mener à des résultats plus intéressants qu'avec un seul objectif. L'algorithme présenté, dénoté "Actuated", est la première version de TAPIOCA. Nous pouvons voir qu'une sélection de phase basée à 75% sur la taille des files d'attente et à 25% sur un critère de famine procure le meilleur temps d'attente, en comparaison à d'autres configurations ou à l'algorithme de Yousef *et al.*. TMAX représente sur ce schéma le temps de feu vert maximum (dénoté  $\tau^{g^{max}}$  dans cette thèse), qui dépend du scénario considéré mais qui – dans un cas général – peut être fixé aux environs de 30 secondes.

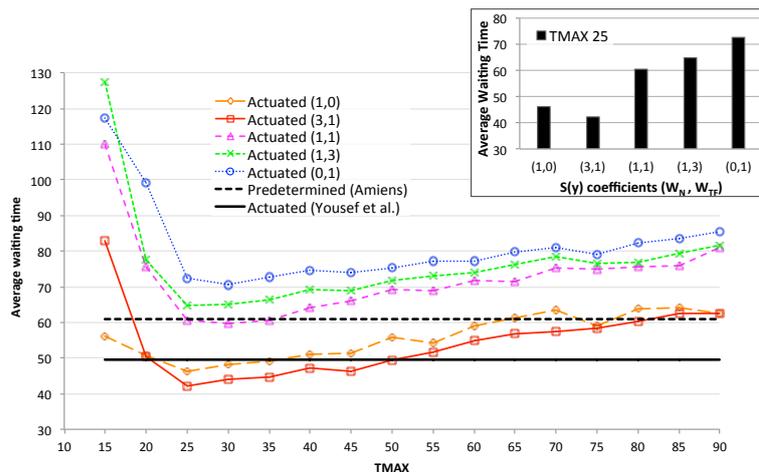


FIGURE 4.1 – Simulations extraites de [FCD12a] montrant l'évolution du temps d'attente en fonction de la pondération de deux critères : (file d'attente, famine).  $S(y)$  représente une fonction de score équivalente à celle définie dans les sections suivantes.

Zhou *et al.* [ZCZW] proposent un algorithme qui sélectionne les combinaisons de mouvements qu'il est possible d'effectuer simultanément et sans conflits. Cela laisse le choix entre douze phases, comme l'illustre la figure 4.2 sur une intersection à quatre directions. Leur choix se base sur de multiples critères : la présence de véhicules prioritaires, la durée des périodes sans véhicule, le degré de famine, le temps d'attente total et la taille des files d'attente. Chaque critère est évalué et la sélection d'une phase se fait par priorité (p. ex., si il y a un véhicule d'urgence, donner le feu vert, sinon, passer au critère suivant). Toutefois, cet algorithme nécessite que tous les véhicules roulent à la même vitesse et soient du même type. [ZCW11] étend

ce travail au cas de plusieurs intersections en considérant davantage de paramètres. Un temps minimum de feu vert est calculé en se basant sur le nombre de véhicules qu'il y a à traiter localement. Il peut être étendu en fonction des flots qui arrivent des intersections voisines. Outre des hypothèses trop lourdes, le problème majeur de cet algorithme – à l'inverse des précédents – est sa complexité. En effet, les critères considérés nécessitent de la mémoire, des calculs et une fréquence de transmission élevée afin de maintenir des valeurs à jour.

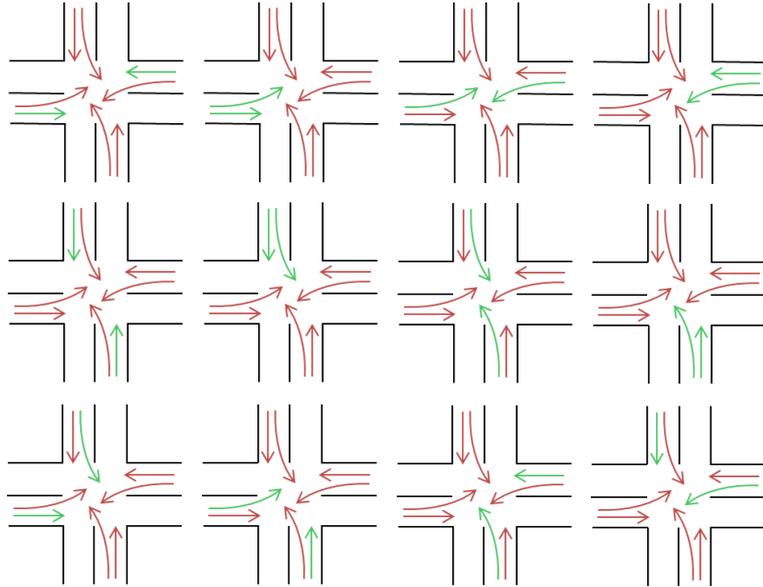


FIGURE 4.2 – Combinaisons sans conflits possibles pour un carrefour à 4 directions.

Dans tous ces papiers, les capteurs sont utilisés comme de simples détecteurs qui reportent leurs mesures à une entité centrale. Toutefois, une telle organisation ne passe pas à l'échelle et a une faible tolérance aux pannes. De plus, cette organisation possède une certaine latence dans la communication et la décision, car les données sont traitées et les décisions sont prises sur un point de collecte proche du contrôleur de feux.

## 4.2 TAPIOCA : organisation du réseau

L'algorithme proposé dans ce chapitre, TAPIOCA, peut être exécuté sur toutes les architectures décrites au chapitre précédent (section 3.2, figures 3.1 et 3.2). Par défaut, TAPIOCA distribue les tâches aux différents capteurs présents sur l'intersection.

Les nœuds de *destination* ( $DN$ ) surveillent et comptabilisent les départs lorsque le feu est vert. Ces nœuds sont typiquement situés au niveau des feux de circulation, sur chaque voie d'entrée ou au début de chaque voie de sortie (cas de la figure 3.2(a)).

Les nœuds *source* ( $SN$ ) mesurent les arrivées en permanence. Ils sont classiquement situés à une distance fixe des feux de circulation, sur chaque voie. Il est

important de bien régler cette distance. Si elle est trop petite, elle n'englobera pas suffisamment de véhicules. Si elle est trop grande, cela augmentera les erreurs liées, par exemple, aux changements de voies. Toutefois, les nœuds *SN* peuvent également être mobiles (p. ex., montés sur rail) ou peuvent être sélectionnés dynamiquement parmi un ensemble de nœuds fixes. Pour de grandes files d'attente, ils peuvent également être remplacés par des caméras.

### 4. 2. 1 Architecture

Parmi les nœuds *DN*, nous choisissons d'élire un *agrégateur de direction* pour chaque direction afin de collecter, traiter et agréger le trafic de tous les nœuds *DN* situés sur les voies émanant de la même direction. Ces agrégateurs servent également à la communication avec les intersections voisines, accessibles via leur route. Enfin, ils servent de niveau hiérarchique intermédiaire, afin d'éviter à une seule entité de tout récolter.

Ces agrégateurs communiquent avec un nœud responsable du calcul du plan de feux et de la communication avec le contrôleur de feux. Ce nœud *décision* peut être élu parmi les capteurs et réélu périodiquement pour remplir certains objectifs (p. ex., distribution des coûts énergétiques).

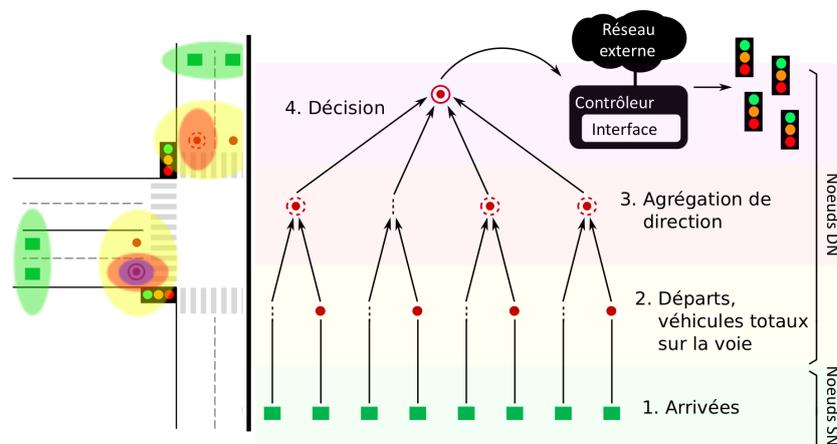


FIGURE 4.3 – Une intersection à quatre directions et son architecture.

Cette organisation, décrite sur la figure 4.3, n'est pas obligatoire pour TAPIOCA. Elle présente toutefois certains avantages.

Tout d'abord, les agrégateurs de direction possèdent des informations sur le nombre total de véhicules sortant de l'intersection vers une destination donnée. Ils sont en mesure d'envoyer cette information à toutes les intersections proches et sont de bons candidats afin d'agir en tant que passerelles vers ces intersections, dans le cas où le réseau fonctionne en multi-sauts.

Deuxièmement, tous les rôles d'agrégateurs sont échangeables. Cela signifie que le protocole d'élection distribué peut être en charge de sélectionner les meilleurs candidats en fonction de plusieurs critères (p. ex., niveau de batterie, capacité des nœuds, etc.). Dans la plupart des cas, ces critères sont numériques et il devient possible de

créer un score attaché à chaque nœud. Sur chaque nœud, la fonction génératrice de ce score serait monotone (toujours décroissante si nous prenons le cas du niveau de batterie). Cette propriété permet notamment d’appliquer des algorithmes d’élection ayant un coût raisonnable, tels que celui proposé par [JKZ02] qui opère en un temps de  $\log(n)$ .

Enfin, ces rôles peuvent être redéfinis entre les capteurs en cas de défaillances. Cette architecture procure un certain niveau de tolérance aux pannes.

## 4. 2. 2 Distance entre les nœuds $SN$ et $DN$

Nous rejoignons bon nombre d’auteurs qui estiment que la distance séparant les nœuds  $SN$  et  $DN$  doit être fonction des limites temporelles fixées sur l’intersection [ZCZW]. Nous estimons ainsi qu’elle doit correspondre au nombre de véhicules qui peuvent traverser l’intersection lorsque la durée du feu vert est à son maximum autorisé, noté  $\tau^{g^{max}}$ . Cette durée est généralement bornée pour limiter le temps d’attente qui serait mal perçu par les usagers.

Cette distance peut être réglée à  $n \cdot l_{veh}$ , où  $n$  dénote le nombre de véhicules qui sont capables de passer en  $\tau^{g^{max}}$  secondes et  $l_{veh}$  la taille moyenne d’un véhicule. Si  $\tau^s$  est le temps de démarrage d’un véhicule et si  $\tau^h$  représente le temps inter-véhiculaire qui sépare deux véhicules, alors :

$$n = \frac{\tau^{g^{max}} - \tau^s}{\tau^h} \quad (4.1)$$

Pour rappel, selon [Gor+05],  $l_{veh} = 6$  mètres,  $\tau^s = 4$  s et  $\tau^h = 2$  s sont des valeurs réalistes qui peuvent être ajustées empiriquement. Pour plus de clarté, ces notations sont illustrées en annexe A.

En se basant sur des études précédentes [FCD12a; FCD12b] et sur [Gor+05], une valeur typique pour cette distance serait 75 mètres. Selon l’équation 4.1, cela correspond à une valeur de  $\tau^{g^{max}}$  de 30 secondes, ce qui est cohérent par rapport à nos simulations (p. ex., figure 4.1).

## 4. 3 TAPIOCA : le cas d’une intersection isolée

Dans [FCD12a], nous considérons que chaque intersection est seule, isolée. Dans ce cas de figure, il n’y a pas besoin de communication à large échelle, le but étant juste de considérer un flux de véhicules entrant. Les capteurs déployés sur chaque intersection échangent des informations en utilisant des communications sans fil et se mettent en accord sur un plan de feux.

Au lieu de définir des cycles, l’algorithme fonctionne uniquement avec une granularité d’une phase. De plus, la sélection dynamique des mouvements s’effectue sur la base de deux objectifs : réduire le temps d’attente moyen et éviter les situations de famine. Enfin, cet algorithme permet également de donner le feu vert aux mouvements qui sont légèrement en conflit. Ces mouvements sont des mouvements qui

n'ont pas d'impact sur la sécurité des utilisateurs, c.-à-d. lorsque des règles de priorités évidentes existent entre les mouvements. Ces trois propriétés permettent à cet algorithme de réduire le temps moyen d'attente des usagers.

L'algorithme 4.1 décrit en pseudo-code la décision qui est prise par le nœud responsable du calcul du plan de feux, afin de gérer une intersection isolée. Il est détaillé et expliqué au fil de cette section. Les mouvements sont d'abord évalués à l'aide de plusieurs objectifs (lignes 1 à 8, section 4.3.1). Ensuite, une phase est sélectionnée ou créée en associant les mouvements existant, en tenant compte d'éventuels conflits avec d'autres voies ou avec des piétons (lignes 9 à 17, sections 4.3.2, 4.3.3 et 4.3.4). Enfin, la durée des feux verts est calculée en fonction de l'importance du trafic et la phase est lancée (lignes 18 à 21, section 4.3.5).

### 4. 3. 1 Classement de mouvements sur la base de deux objectifs

Considérons un mouvement possible  $(s, d)$  allant de la direction  $s$  à la direction  $d \in D$ ,  $D$  dénotant l'ensemble des directions possibles. Le nœud de décision (c.-à-d., couche 4 sur la figure 4.3) connaît la répartition des véhicules sur l'intersection. Il est donc capable d'associer un score local  $S(s, d)$  au mouvement  $(s, d)$ . Ce score dépend du nombre de véhicules présents sur les voies entrantes qui composent le mouvement  $(\eta^{(s,d)})$  et du temps passé depuis la dernière sélection du mouvement  $(F^{(s,d)})$ , c.-à-d. depuis qu'il a eu le feu vert.

Nous devons combiner les métriques qui reflètent chaque objectif (charge et intervalle entre deux sélections successives) dans une seule expression afin de définir formellement ce score. À ces fins, nous normalisons chaque objectif en utilisant une fonction générique,  $\gamma(\cdot)$ . Le but de cette fonction est de mettre les objectifs  $F$  et  $\eta$  dans l'intervalle  $[0; 1]$  et ainsi de les dépourvoir de dimensions, à des fins de comparaisons.  $\gamma(\{F, \eta\}^{(s,d)})$  est défini naturellement comme le ratio entre la valeur de l'objectif du mouvement  $(s, d)$  et la somme de toutes les valeurs d'objectif des différents mouvements :

$$\gamma(F^{(s,d)}) = \frac{F^{(s,d)}}{\sum F} \qquad \gamma(\eta^{(s,d)}) = \frac{\eta^{(s,d)}}{\sum \eta}.$$

Notons que si la somme toutes les valeurs de  $\{F, \eta\}$  est nulle, alors ce score est nul également. Lorsque cette étape est réalisée, une approche classique serait de définir le score comme étant une combinaison linéaire de  $\gamma(F^{(s,d)})$  et de  $\gamma(\eta^{(s,d)})$ . Toutefois, sachant que nous cherchons un classement plutôt qu'une évaluation, nous aimerions donner plus de poids aux mouvements qui ont un nombre de véhicules nettement plus élevés que les autres, ou qui n'ont pas été sélectionnés depuis longtemps. C'est pourquoi nous définissons le score comme étant une somme pondérée des carrés des mesures normalisées :

$$S(s, d) = W_\eta \cdot \left( \gamma(\eta^{(s,d)}) \right)^2 + W_F \cdot \left( \gamma(F^{(s,d)}) \right)^2,$$

---

**Algorithme 4.1:** algorithme simplifié de TAPIOCA sur une intersection isolée

---

**Paramètres :** *mouvements*, liste des mouvements de l'intersection ; *phases*, liste des phases possibles sur l'intersection ;  $\tau^{g^{max}}$ , temps de feu vert maximal.

**Variables :**  $\eta^m$ , nombre de véhicules sur le mouvement  $m$  ;  $\eta^{voie}$ , nombre de véhicules sur la voie *voie* ;  $F^m$ , dernière fois que le mouvement  $m$  a eu le feu vert ;  $p_{suivant}$ , phase suivante.

```

// Évaluation de chaque mouvement
1 majF();
2 pour chaque  $m \in mouvements$  faire
3    $\eta^m \leftarrow 0$ ;
4   pour chaque  $voie \in m$  faire
5      $\eta^m \leftarrow \eta^m + (\eta^{voie} / nbMouvements(voie))$ ;
6   finprch
7    $S(m) = (\eta^m / \sum \eta)^2 + (F^m / \sum F)^2$ ;
8 finprch
// Sélection d'une phase parmi celles connues (p. ex., SUMO - Sec. 4. 3. 2. 2)
9  $S_{max} = -1$ ;  $p_{suivant} = -1$ ;  $\eta^{max} = -1$ ;
10 pour chaque  $p \in phases$  faire
11   // Le score d'une phase est la somme du score de ses mouvements
12    $score = \sum_{m \in p} S(m)$ ;
13   // Une phase est sélectionnée si elle a le plus haut score et si elle
14   possède des véhicules
15   si  $S_{max} < score$  et  $fileMax(p) > 0$  alors
16      $S_{max} \leftarrow score$ ;
17      $p_{suivant} = p$ ;
18      $\eta^{max} = fileMax(p)$ ;
19   finsi
20 finprch
// Calcul de la durée du feu vert et lancement de la phase
21 si  $S_{max} < score$  et  $fileMax(p) > 0$  alors
22    $\tau^g = \min(\eta^{max} \cdot \tau^h, \tau^{g^{max}})$ ;
23   lancerPhase( $p_{suivant}, \tau^g$ );
24 finsi

```

**majF()** : Remet à jour les valeurs de  $F^m$ , pour chaque mouvement  $m$ . Au moment de l'appel de cette procédure, tous les mouvements ayant le feu vert sont paramétrés avec une valeur nulle ( $F^m = 0$ ). Tous les autres sont mis à jour en fonction du dernier temps depuis lequel ils ont eu le feu au vert ( $F^m = temps\ actuel - dernier\ temps\ connu$ ).

**nbMouvements(voie)** : retourne le nombre de mouvements possibles depuis une voie *voie*.

**fileMax(phase)** : retourne la taille de la plus grande file appartenant à la phase *phase*.

**lancerPhase(phase, temps)** : Provoque l'arrêt de la phase en cours et lance la phase suivante *phase* pendant une durée de *temps* secondes.

---

où  $W_\eta$  et  $W_F$  sont des poids définis par le concepteur, qui peuvent être changés empiriquement par les opérateurs afin de favoriser la performance (temps moyen d'attente) ou l'expérience de l'utilisateur (famine). Dans l'article [FCD12a] (dont l'un des résultats est représenté sur la figure 4.1), nous montrons par simulation l'intérêt de confronter ces deux paramètres. Il est également possible de mettre en place un algorithme d'apprentissage local ou un réglage dépendant des poids des intersections voisines. Par défaut, ces poids restent identiques. Mettre au carré les objectifs garde les valeurs dans le même intervalle mais accentue leur importance quand celles-ci sont élevées. Cette dernière propriété a l'effet désiré : si l'un des objectifs a une valeur élevée pour un mouvement donné par rapport aux autres, sa contribution au score doit être suffisamment importante pour favoriser sa sélection. Si aucun véhicule n'est présent sur les voies originaires de la direction  $s$ , le score est défini comme étant nul pour tous les mouvements concernés, même si le mouvement n'a pas été sélectionné depuis longtemps :  $\forall d \in D, S(s, d) = 0$ .

## 4. 3. 2 Sélection d'une phase

### 4. 3. 2. 1 Basée sur une matrice des conflits

Une fois que les scores ont été calculés pour chaque mouvement, le nœud de décision examine quels mouvements peuvent être combinés afin de créer une nouvelle phase. À ces fins, il utilise une matrice des conflits, qui indique les mouvements qui peuvent être effectués en toute sécurité simultanément (sec. 2. 2. 2. 1).

Cette matrice est utilisée pour combiner les mouvements individuels qui peuvent composer une phase valide, en additionnant leurs scores et en explorant toutes les combinaisons possibles (afin de trouver les plus optimales). La phase sélectionnée est celle qui possède le plus haut score.

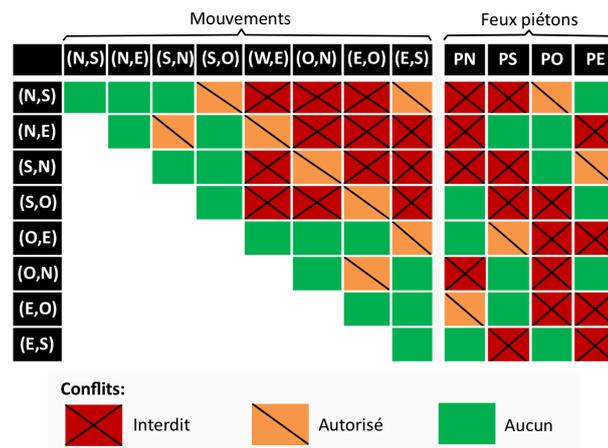


FIGURE 4.4 – Exemple d'une matrice des conflits.

La figure 4.4 représente une telle matrice des conflits avec trois niveaux. (1) Les éléments verts indiquent les mouvements non conflictuels. (2) Les éléments rouges indiquent les mouvements qui ne peuvent pas se dérouler en même temps, p. ex. pour des raisons de sécurité. (3) Entre les deux, les éléments oranges indiquent les mouve-

ments qui sont en conflit, mais avec un niveau de sécurité acceptable (p. ex. priorité face aux mouvements allant à gauche). Ce cas est représenté sur la figure 4.5, où deux mouvements de virage à gauche sont en conflit. Notons que ce cas peut cependant limiter le nombre de véhicules qui peuvent traverser l'intersection et potentiellement introduire des situations de blocage. La section 4.3.3 décrit comment TAPIOCA permet de gérer de telles situations.

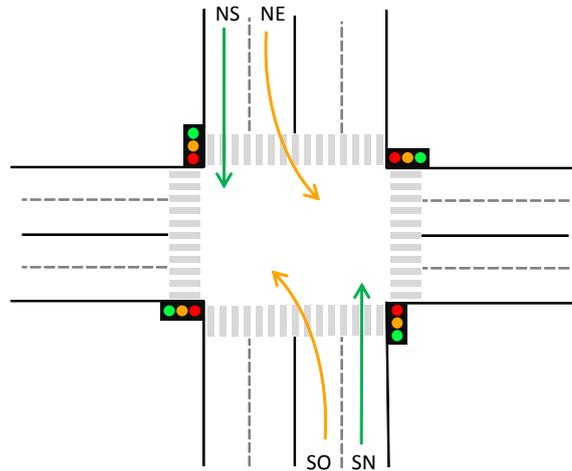


FIGURE 4.5 – Exemple de phase avec deux mouvements en conflit léger.

Pour finir, remarquons que cette matrice comprend également des colonnes pour les passages piétons, abordés en section 4.3.4.

#### 4.3.2.2 Basée sur une phase statique

Trouver la meilleure combinaison de mouvements lorsque la matrice des conflits autorise de multiples mouvements simultanés requiert des calculs. Il nous faut en effet maintenir une liste de toutes les combinaisons de mouvements possibles sur chaque intersection. Une combinaison pouvant avoir plus de deux mouvements. Cela nécessite des calculs initiaux de complexité variable (dépendant de la taille de la matrice des conflits) et des coûts de mémoire supplémentaires. Pour plus de simplicité, dans nos simulations (voir 4.3.6, 4.6.1 et 4.6), nous supposons que la liste des phases possibles est déjà connue. Pour cela, nous nous basons sur les plans de feux statiques préconfigurés sur chaque intersection, dont les phases sont réalistes. Afin de composer des phases en conflit léger (section précédente), nous pouvons également combiner les phases des plans de feux statiques.

Pour chaque phase possible, il nous est facile de calculer le score  $S(s, d)$  de chaque mouvement ayant le feu vert. La phase sélectionnée est la phase qui possède la plus haute somme de scores.

### 4. 3. 3 Politique avec des mouvements faiblement en conflit

Les véhicules qui tournent à gauche ont généralement la priorité la plus basse, ils sont bloqués lorsqu'un véhicule vient du mouvement situé dans la direction opposée. La possibilité de laisser les deux mouvements se produire en même temps dépend donc de la charge des deux mouvements.

Les stratégies classiques consistent à laisser passer chacun de ces mouvements dans une phase dédiée, ou de séparer la phase en deux sous-phases : la première sous-phase permet aux mouvements concurrents de passer et la seconde permet aux mouvements à faible priorité de passer. Une telle phase est légèrement différente du cas où deux phases s'enchaînent, car la deuxième sous-phase s'enchaîne immédiatement après la première. TAPIOCA adopte cette deuxième solution.

### 4. 3. 4 Passages piétons

Les passages piétons peuvent facilement être inclus dans le modèle de TAPIOCA, en ajoutant des entrées à la matrice des conflits, comme le montre la figure 4.4. Les passages pour piétons peuvent être considérés comme des voies de véhicules classiques. Cependant, ils se comportent différemment, car il est plus difficile de dénombrer les usagers en attente. TAPIOCA adopte une autre approche : lorsque la composition d'une phase est déterminée, tous les passages piétons qui ne sont pas en conflit avec les mouvements sélectionnés sont sélectionnés également. Fonctionner ainsi ne génère pas de famine car les feux piétons suivent le rythme des feux de circulation. Cette règle peut être allégée, nous pourrions par exemple tolérer les conflits modérés avec une règle de priorité (p. ex., lorsque des véhicules tournent à droite et doivent donner la priorité aux piétons). La matrice des conflits doit être configuré pour permettre à tous les passages pour piétons d'avoir le feu vert. Dans ce cas, les passages pour piétons ne sont pas un obstacle à la composition de phase, ils sont complémentaires et n'interfèrent pas dans le processus de décision.

Dans des cas particuliers, il est impossible de configurer une matrice des conflits avec aucun conflit entre les véhicules et les piétons. Dans ce cas, un passage pour piétons  $P_n$  peut avoir les mêmes propriétés que les mouvements de véhicules :  $F^{P_n}$  est la dernière fois que le feu pour piétons est passé au vert et  $\eta^{P_n}$  est un indicateur de présence de piétons. Ce dernier peut être calculé avec plus ou moins de précision par des caméras, des capteurs piézoélectriques ou plus simplement avec des boutons poussoirs.

### 4. 3. 5 Durée de phase

La vie d'une intersection est, dans TAPIOCA, une succession de phases. Contrairement à la plupart des stratégies, il n'y a pas de notion explicite de cycle. Les phases se succèdent et TAPIOCA veille à ce que tous les mouvements soient sélectionnés régulièrement.

Quand une phase se termine, les feux passent à la couleur orange pendant un temps  $\tau^{orange}$ , traditionnellement fixé à 3 secondes. Bien entendu, des méthodes précises d'obtention de ce temps existent, nous le simplifions toutefois à une valeur moyenne pour éviter des temps de calcul trop longs [Gor+05; US 08; Min09]. Puis, tous les feux se mettent au rouge pendant un temps de sécurité  $\tau^{secure}$ . Cela laisse suffisamment de temps au contrôleur afin de déterminer quels mouvements composeront la phase à venir.

Une fois les mouvements sélectionnés, en adéquation avec les critères exposés dans les sections précédentes, le contrôleur calcule le temps de feu vert de la phase.

#### 4.3.5.1 Temps de feu vert

Pour rappel,  $T_s$  est le temps de démarrage et  $\tau^h$  est le temps qui sépare le passage de deux véhicules successifs.  $\tau^{g^{max}}$  est la durée de feu vert maximale, qui est soit défini par l'utilisateur, soit déterminé de façon à équilibrer les performances et l'expérience des utilisateurs (section 4.2).  $\tau^g$ , le temps de feu vert, est calculé en fonction de la charge de la voie ayant le plus grand nombre de véhicules. Ceci laisse la possibilité à l'intersection de pouvoir vider toutes les voies concernées :

$$\tau^g = \min(\tau^s + \eta^{max} \cdot \tau^h, \tau^{g^{max}}),$$

où  $\eta^{max}$  est le nombre de véhicules sur la voie la plus chargée. Ce temps est borné par  $\tau^{g^{max}}$  afin d'éviter de bloquer l'intersection. Dans le cas où  $\tau^g < \tau^{g^{max}}$ , des véhicules additionnels peuvent arriver sur ces voies. Dans ce cas, nous choisissons de laisser le temps de feu vert se rallonger d'une période de  $\tau^h$ , jusqu'à ce qu'il n'y ait plus de nouvelles arrivées, ou jusqu'à ce que  $\tau^g = \tau^{g^{max}}$ .

#### 4.3.5.2 Effets du temps de feu maximum

La valeur de  $\tau^{g^{max}}$  a une influence sur la performance globale du système. Dans [FCD12a] (dont l'un des résultats est présenté sur la figure 4.1), nous trouvons que le temps optimal de  $\tau^{g^{max}}$  pour une intersection de la ville d'Amiens (France) est situé entre 25 s et 35 s. La valeur de  $\tau^{g^{max}}$  est plus petite lorsque les conflits sont interdits, car laisser tous les mouvements survenir dans cette situation nécessite plusieurs phases, ce qui réserve moins de temps à une seule phase. Dans ces premières simulations, nous avons choisi de paramétrer empiriquement  $\tau^{g^{max}}$ , en testant un ensemble de valeurs de 15 s à 70 s et en utilisant le meilleur résultat.

Si une telle approche empirique est toujours possible, nous proposons ici de régler  $\tau^{g^{max}}$  en considérant la durée initiale du cycle ( $\tau^c$ ) défini dans les plans de feux statiques lorsqu'ils sont disponibles.

$\tau^c$  est notamment réglé pour limiter le temps d'attente entre deux sélections de la même phase, et est défini en fonction du trafic et de la configuration de l'intersection. Sur une intersection arbitraire, nous configurons  $\tau^{g^{max}}$  proportionnellement au nombre de véhicules qui composent la phase :

$$\tau^{g^{max}} = \frac{\eta^{max} \cdot \tau^c}{\sum_{\{a,b\} \in D} \eta_{max}^{(a,b)}}.$$

Ainsi, nous sommes assurés d'obtenir une valeur de  $\tau^{g^{max}}$  cohérente par rapport à la diversité des intersections. Cette méthode est appliquée dans nos simulations, et se trouve utile et efficace pour les grands scénarios tels que TAPASCologne (section 4.6).

### 4. 3. 6 Évaluation sur une intersection isolée (Amiens)

Les simulations présentées sur la figure 4.6 sont effectuées à l'aide de SUMO et sont basées sur une intersection de la ville d'Amiens (France), pour laquelle nous avons à notre disposition des données réelles de trafic, la répartition des véhicules et les plans de feux statiques (cycles, phases et temporisations). Chaque simulation dure 7 200 secondes et ne nécessite aucune période d'entraînement. Les résultats présentés sur la figure 4.6 sont les valeurs moyennes calculées sur les véhicules qui traversent l'intersection pendant ce temps de simulation.

Les figures 4.6(a) et 4.6(b) montrent que TAPIOCA réalise le meilleur temps d'attente moyen par rapport à un algorithme adaptatif issu de la littérature (55%), au plan de feu statique de la ville d'Amiens (62%) et au plan de feux généré par SUMO (80%). Notons que SUMO génère ses plans de feux sans prendre en compte le trafic qui est potentiellement présent, mais en se basant uniquement sur la cartographie du réseau et des valeurs représentatives des cas de figures généraux. Les plans de feux générés sont donc souvent inadaptés et représentatifs d'un scénario mal configuré. La figure 4.6(c) montre que TAPIOCA produit également les files d'attente les plus courtes en moyenne. Enfin, la figure 4.6(d), qui décrit la répartition de la meilleure stratégie pour les véhicules intégrés aux simulations, nous montre que TAPIOCA génère un meilleur temps d'attente pour plus de 40% des véhicules du scénario.

Les résultats que nous décrivons sur une intersection semblent encourageants. Toutefois, nous souhaitons nous intéresser à présent aux effets d'un scénario possédant plusieurs intersections. Notre algorithme ayant uniquement une vision locale, aucune collaboration n'existe entre les feux verts d'intersections adjacentes. Ceci peut notamment provoquer des arrêts ou des phases qui enverraient de manière excessive, par exemple, des véhicules sur une intersection voisine encombrée. Cette intuition est notamment levée par la figure 4.7, qui est extraite de simulations présentant un scénario à plusieurs intersections, abordé dans la suite du document. Nous constatons que la version isolée de TAPIOCA se comporte nettement moins bien que l'algorithme de Yousef *et al.* (conçu pour gérer plusieurs intersections) et à hauteur du plan de feux prédéterminé de la ville d'Amiens (conçu pour faire correspondre les feux de circulation).

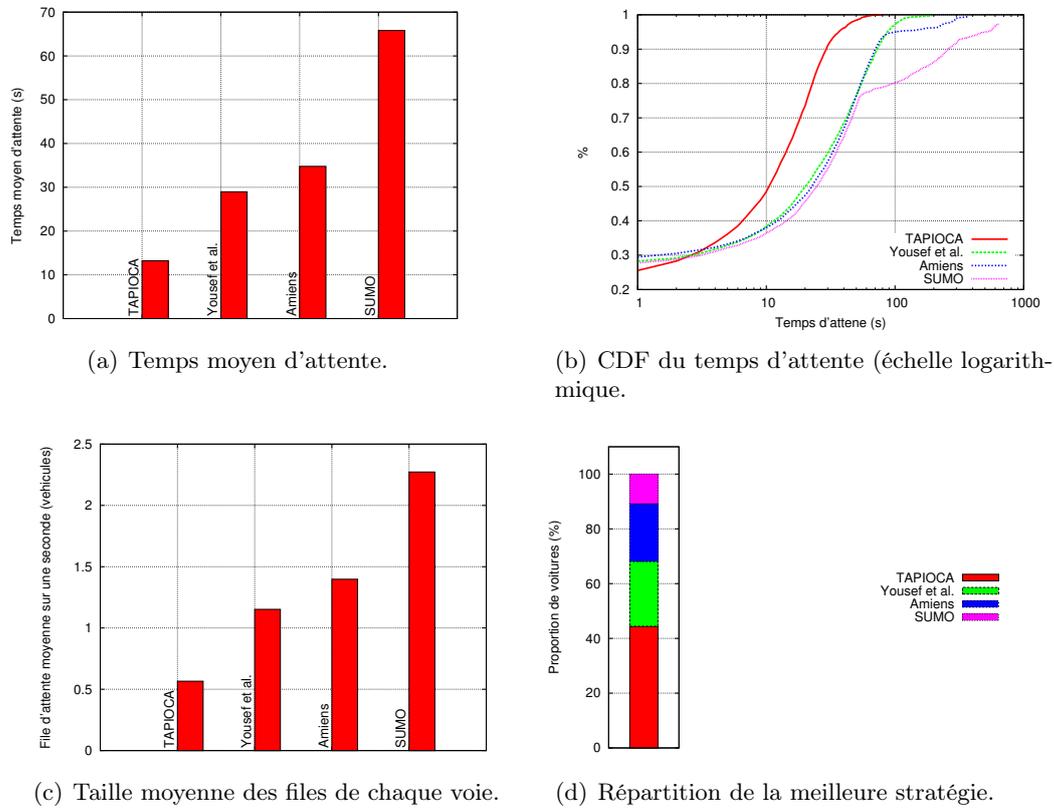


FIGURE 4.6 – Simulations sur une intersection isolée de la ville d'Amiens.

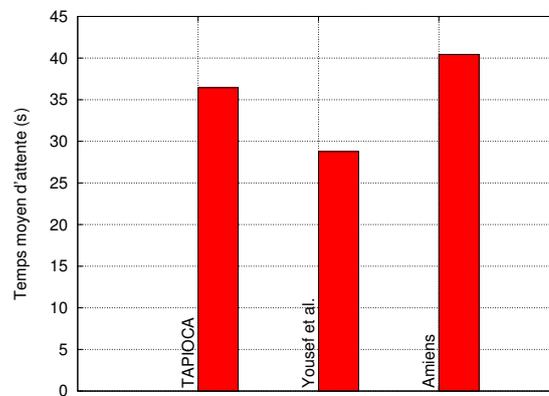


FIGURE 4.7 – Simulations sur trois intersections adjacentes de la ville d'Amiens.

## 4. 4 TAPIOCA : le cas de plusieurs intersections

Dans cette section, nous proposons une généralisation de l’algorithme précédent à plusieurs intersections adjacentes, en définissant une méthode pour synchroniser les intersections proches et créer des vagues vertes. Nous supposons que chaque intersection peut communiquer avec ses intersections adjacentes à travers le réseau de capteurs sans fil. Des relais supplémentaires peuvent être utilisés, formant un réseau multi-sauts, lorsque la distance entre deux intersections est plus grande que la distance de transmission des nœuds, comme étudié au chapitre 3. Une connexion WAN est également possible, si elle est disponible. Nous supposons également que les horloges des nœuds sont synchronisées avec une précision suffisante pour l’application (c.-à-d. approximativement une seconde).

### 4. 4. 1 Cas 1 : amélioration du score

Dans [FCD12b], nous avons proposé une première approche pour gérer le cas des intersections multiples. Cette approche sélectionne les mouvements qui composent la phase à venir sur la base de mesures locales et distantes combinées en un score global, qui mélange trois objectifs. D’abord, un score local est calculé avec la méthode décrite dans la section 4. 3. 1. Ensuite, les résultats sont échangés entre intersections proches de sorte à favoriser les mouvements qui, (1) ont la plus grande capacité (c.-à-d. éviter d’envoyer trop de véhicules dans les zones déjà embouteillées) et qui, (2) tentent de synchroniser les intersections successives pour créer des vagues vertes.

Les vagues vertes sont des chemins de feux verts successifs, synchronisées afin que les véhicules ne ralentissent pas. Il s’agit de l’une des techniques les plus efficaces pour décharger un réseau, car elle rend le trafic plus fluide. La mise en œuvre habituelle de vagues vertes suppose que le véhicule roule à la vitesse limite et la synchronisation doit être adaptée lorsqu’une surcharge apparaît (sec. 2. 2. 3. 5). Cette première version ne tient pas compte de la charge et sa performance n’est donc pas optimale lorsque le niveau de congestion retarde considérablement les véhicules entre deux intersections voisines.

### 4. 4. 2 Cas 2 : synchronisation indépendante

Dans la version de TAPIOCA introduite dans le présent chapitre, nous avons choisi de modifier la philosophie de synchronisation des intersections. Au lieu de prendre directement en compte les informations provenant de voisins dans le calcul d’un score complexe, nous laissons chaque intersection prendre ses propres décisions au niveau local, comme si elle était isolée. Le résultat est un ensemble de mouvements actifs et un nombre prévu de véhicules qui iront, pendant la phase suivante, de l’intersection de départ ( $I^1$ ) jusqu’à certaines intersections voisines.

Notons  $I^2$  l’une des ces voisines qui devrait s’attendre à recevoir un flux entrant.  $I^1$  pourrait envoyer un message afin d’avertir  $I^2$  à propos de ce nouveau flux. Cependant, cela pourrait générer un trafic réseau élevé, qu’il nous faut limiter afin de

réduire la probabilité de pertes et de congestion dans le réseau de communication. De plus, l'utilisation d'un réseau de type IEEE 802.11p, qui a vocation à être partagé, accentuerait ce problème. C'est pourquoi nous avons choisi de limiter la transmission de ces messages dans le seul cas où  $I^2$  est moins chargée (c'est à dire a un nombre total de véhicules plus faible) que  $I^1$ . L'idée derrière ce filtrage est une stratégie gloutonne :  $I^2$  ne devrait envisager l'interruption de son propre cycle que si cela permet de réduire la charge d'une intersection voisine plus chargée. Sinon, elle poursuit son propre programme local.

Ce message de synchronisation contient le nombre de véhicules que  $I^1$  s'attend à envoyer à  $I^2$  et une estimation du temps requis par ces véhicules pour atteindre  $I^2$ . Ce temps peut être évalué en se basant simplement sur la distance entre les deux intersections. Il peut également être évalué par l'échange d'un échantillon de signatures électromagnétiques des véhicules entre les nœuds  $DN$  de  $I^1$  et les nœuds  $SN$  de  $I^2$ , en tenant compte du temps inter-véhiculaire ( $\tau^h$ ), orange ( $\tau^{orange}$ ), de sécurité ( $T^{secure}$ ) et de démarrage ( $T^s$ ) (sec. 2.2.3).

Lorsque  $I^2$  reçoit un tel message, il évalue localement la possibilité de rompre le cycle en cours pour favoriser une vague verte. En effet, le filtrage de message qui a été effectué par  $I^1$  repose nécessairement sur des informations légèrement anciennes. Il compare les deux gains attendus, c'est à dire le nombre de véhicules qui traversent l'intersection si la phase actuelle est maintenue et si la vague verte interrompt la phase actuelle. Si le choix est en faveur de la vague verte,  $I^2$  détermine une nouvelle composition de la phase, en appliquant l'algorithme de TAPIOCA classique avec la contrainte supplémentaire de sélectionner nécessairement les mouvements qui correspondent au flux entrant.

Les messages de synchronisation arrivent de manière asynchrone sur une intersection, qui pourrait avoir à faire face à plusieurs demandes au cours de la même phase. Dans ce cas, la stratégie qui maximise le nombre de véhicules à recevoir est sélectionnée.

La figure 4.8 représente un réseau de neuf intersections. Chacune de ces intersections déroule une phase, composée des mouvements représentés par les flèches vertes. Les messages de synchronisation qui sont transmis sont représentés par les flèches noires. Remarquons qu'une intersection n'envoie pas de message à des voisins qui ne sont pas la destination des mouvements de la phase courante, ni à des voisins qui sont plus chargés en termes de véhicules. Une fois que ces messages sont reçus, les intersections de destination prennent une décision et ignorent les demandes lorsque le gain des objectifs locaux est plus intéressant que le gain des vagues vertes. Ces demandes ignorées sont représentées par des flèches barrées.

## 4.5 TAPIOCA : algorithme détaillé

TAPIOCA est distribué sur les intersections et sélectionne les mouvements qui composent la phase  $P + 1$  à la fin de la phase  $P$ , en se basant sur les principes exposés aux sections 4.3 et 4.4. L'algorithme détaillé ci-dessous est destiné à être mise en place sur l'architecture 3.1 ou 3.2(b), c.-à-d. avec les nœuds  $SN$  positionnés

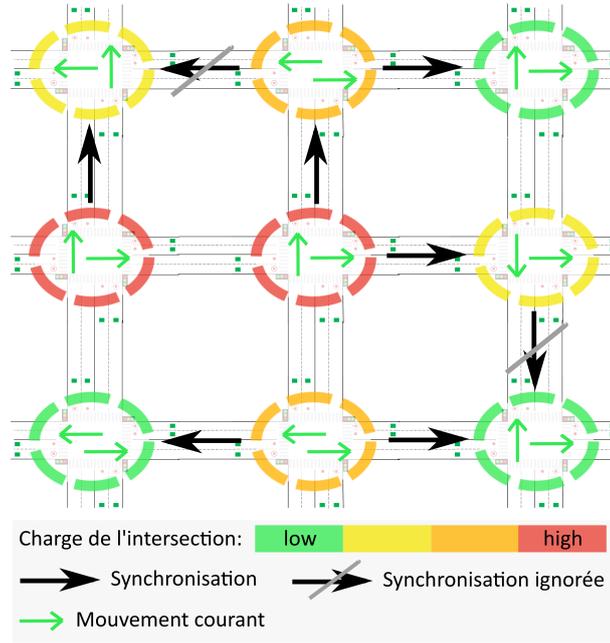


FIGURE 4.8 – Exemple de scénario où des messages de synchronisation envoyés entre les intersections.

sur les mêmes voies que les nœuds  $DN$ . Le cas de l'architecture 3.2(a) est détaillé dans [FCD12b].

TAPIOCA est décomposé en 7 étapes :

1. **Comptabilisation des véhicules sur chaque voie (nœuds SN et DN) :** pour chaque voie  $i$ , à la fin de la phase  $P$ , chaque nœud  $SN$  (couche 1) envoie  $\eta_i^A(P)$ , le nombre d'arrivées enregistrées pendant la phase  $P$ , au nœud  $DN$  qui est sur sa voie (couche 2). En parallèle, chaque nœud  $DN$  a enregistré le nombre de départs survenus durant la phase ( $\eta_i^D(P)$ ). Chaque nœud  $DN$  met ensuite à jour  $\eta_i(P+1)$ , le nombre de véhicules qui sont présents sur la voie au début de la phase  $P+1$  :

$$\eta_i(P+1) = \eta_i(P) + \eta_i^A(P) - \eta_i^D(P).$$

Il transmet ensuite toutes les valeurs de  $\eta_i(P+1)$  à l'agrégateur de direction (couche 3) qui gère la direction entrante à laquelle le nœud  $DN$  appartient.

2. **Agrégation par direction (nœuds agrégateurs de direction) :** pour chaque direction  $s$ , l'agrégateur maintient le temps passé depuis la dernière sélection des mouvements qui commencent en  $s$ ,  $F^s$ , afin de détecter et de prévenir les cas de famine. Pour chaque mouvement commençant en  $s$ , il additionne les valeurs de  $\eta_i^{P+1}$  reçues depuis toutes les voies pertinentes pour obtenir  $\eta^{(s,d)}$ , la longueur totale des files d'attentes pour le mouvement  $(s, d)$ .

Si une voie  $i$  est l'origine de  $M$  mouvements ( $M > 1$ ), nous choisissons d'établir par défaut pour chaque mouvement  $(s, d)$ ,  $\eta^{(s,d)} = \eta_i^{P+1}/M$ , afin d'éviter de

compter  $M$  fois les véhicules de la voie  $i$ . Si des capteurs additionnels sont installés sur les voies de sortie, ils peuvent coopérer avec le nœud  $DN$  de la voie qui a plusieurs mouvements afin de déterminer un coefficient pour chaque mouvement. Par exemple, si une moyenne de 60% de véhicules de la voie  $i$  suivent le mouvement  $(s, d)$ , alors nous pouvons configurer  $\eta^{(s,d)} = \eta_i^{P+1} \cdot 0.6$ . Finalement, le nœud responsable de la direction transmet ces deux valeurs au nœud responsable de l'intersection (couche 4). Ces types de nœuds gèrent également les messages de synchronisation reçus des intersections voisines. Si ils en ont, ils les transmettent au même moment.

3. **Composition de la prochaine phase (nœud décision)** : une fois qu'il a reçu les données de tous les agrégateurs, le nœud de décision calcule les scores locaux  $S(s, d)$  des différents mouvements (section 4.3.1). Il combine ensuite les mouvements en utilisant la matrice des conflits ou un ensemble prédéterminé de phases du plan de feux statique et sélectionne la combinaison qui possède le meilleur score. À ce stade, des critères supplémentaires peuvent être envisagés (p. ex., la détection de véhicules d'urgence, éviter une combinaison en cas de détection d'accident). Un exemple de cette opération, en pseudo-code, est donné sur l'algorithme 4.1 (lignes 1 à 17).
4. **Durée de la prochaine phase (nœud décision)** : une fois que les mouvements sont sélectionnés, le temps de feu vert est configuré en accord avec la section 4.3.5.
5. **Transmission aux intersections voisines (nœud décision, agrégateurs)** : le nœud décision transmet la composition de la prochaine phase et sa temporisation aux nœuds agrégateurs, avec un éventuel message de synchronisation et le nombre de véhicules sur l'intersection. Chaque nœud agrégateur transmet ensuite ces informations aux intersections voisines.
6. **Application de la phase (nœud décision)** : le nœud décision ordonne au contrôleur de changer les feux spécifiés pendant le temps spécifié. Cela marque le début de la phase  $P + 1$ .
7. **Pendant la phase  $P + 1$  (nœud décision)**. Si un message de synchronisation arrive et est considéré intéressant (section 4.4), le nœud décision relance le processus de sélection de phase, en forçant la sélection des mouvements évoqués par la vague verte. Lorsque la phase est finie, le nœud décision applique si nécessaire le temps supplémentaire pour les voies tournant à gauche (section 4.3.3).
- **Surveillance des véhicules entre les intersections (nœud DN)** : durant la phase  $P + 1$ , les nœuds  $DN$  des directions sélectionnées peuvent envoyer les signatures temporisées des véhicules aux nœuds  $DN$  correspondant sur les intersections qui sont susceptibles de les accueillir. Cela permet d'estimer le temps qui est requis pour aller d'une intersection à une autre. Ce délai peut être utilisé afin d'améliorer le processus de création de vagues vertes, sa variation permettant de détecter assez tôt une formation d'embouteillage.

La figure 4.9 représente le diagramme de collaboration de TAPIOCA, c.-à-d. les relations qui existent entre les nœuds et les principales actions de l’algorithme. Notons que l’algorithme se base sur des notions définies dans ce chapitre. Toutefois, il est facile d’imaginer des améliorations. Nous pourrions par exemple considérer la défaillance d’un capteur ou des informations provenant d’un réseau véhiculaire, de téléphones portables ou de GPS connectés.

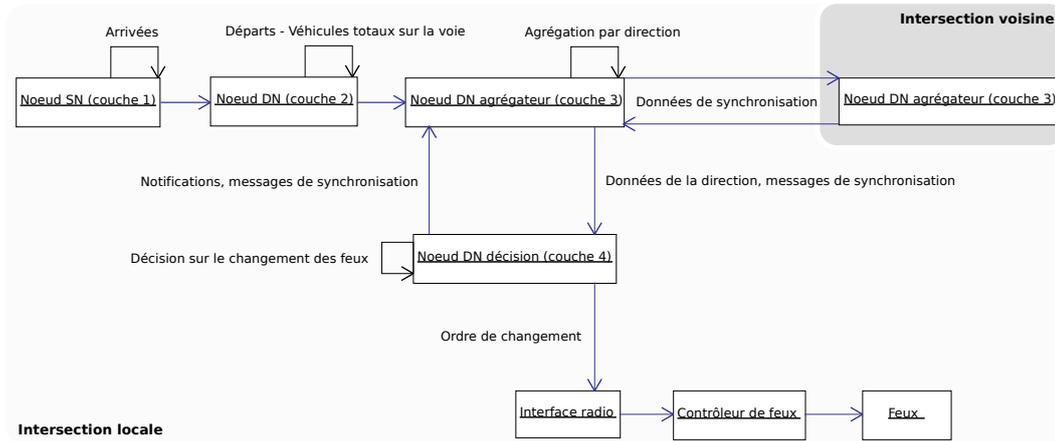


FIGURE 4.9 – Diagramme de collaboration de TAPIOCA.

## 4. 6 Simulations

Nous évaluons TAPIOCA en utilisant le simulateur SUMO 0.17. Nos simulations sont faites sur plus de 1 425 137 voyages distribués parmi plus de 32 scénarios distincts : les résultats complets et plus de figures, sont disponibles en ligne<sup>1</sup>. Nous détaillons dans cette section des métriques et scénarios représentatifs.

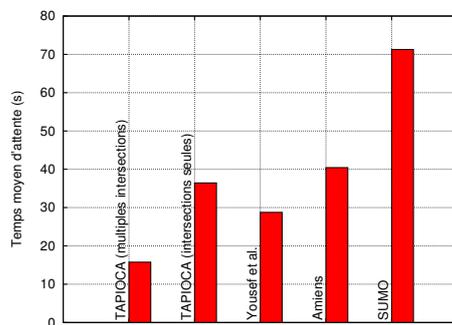
### 4. 6. 1 Scénario 1 : ville d’Amiens

Les simulations présentées sur les figures 4.10, de la même manière que celles présentées en section 4. 3. 6, sont basées sur trois intersections de la ville d’Amiens (France), pour laquelle nous avons des données de trafic, dont la répartition des véhicules et les plans de feux (cycles, phases, synchronisations entre intersections et temps de feux). Chaque simulation dure 7 200s et est traversée par plus de 5 500 véhicules.

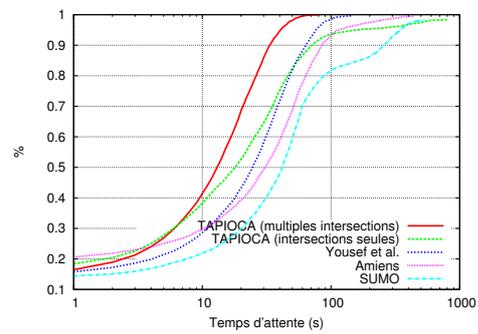
Les figures 4.10(a) et 4.10(b) montrent la comparaison entre TAPIOCA, TAPIOCA sur une intersection isolée ([FCD12a], sec. 4. 3), l’algorithme de Yousef *et al.* ([YAKS10]), le plan de feux utilisé sur Amiens et le plan de feux généré par SUMO.

Les résultats de simulations nous montrent que TAPIOCA, en créant des vagues vertes, accompli le meilleur temps moyen d’attente. Il est de 56% meilleur que TA-

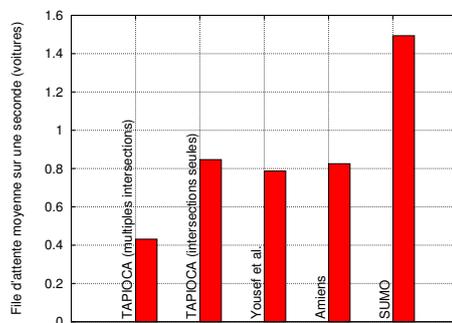
1. <http://tapioca.sfaye.com/>



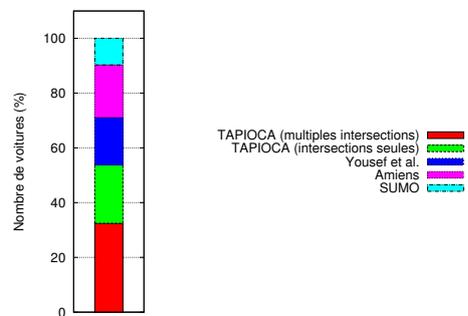
(a) Temps moyen d'attente.



(b) CDF du temps d'attente (échelle logarithmique).



(c) Longueur de file moyenne.



(d) Répartition de la meilleure stratégie.

FIGURE 4.10 – Comparaison de performances : plusieurs intersections de la ville d'Amiens.

PIOCA sur simple intersection, 45% meilleur que l'algorithme de Yousef *et al.*, 61% meilleur que le plan de feux de Amiens et 78% que le plan de feux de SUMO.

La figure 4.10(c) montre que TAPIOCA génère également les plus petites files, tandis que la figure 4.10(d) montre la répartition des stratégies en fonction des usagers. Nous pouvons constater que TAPIOCA génère des temps d'attente plus courts pour 32% des usagers, et la version isolée pour 17% des usagers. Au total, la moitié des utilisateurs attendent moins avec notre algorithme, avec ou sans vagues vertes.

## 4. 6. 2 Scénario 2 : TAPASCologne

TAPASCologne<sup>2</sup> est l'un des plus grands – si ce n'est le plus grand – jeu de données à destination de SUMO. Il modélise la ville de Cologne (Köln, Allemagne) en se basant sur sa carte OpenStreetMap. Il intègre deux heures de trafic, c.-à-d. plus de 70 000 véhicules. Il est représenté sur la figure 4.12.

Les résultats sont présentés sur la figure 4.11 et montrent le temps moyen d'attente des usagers, ainsi que la file d'attente moyenne. Nous pouvons voir, sur les figures 4.11(a), 4.11(b) et 4.11(c), que TAPIOCA accomplit toujours le meilleur temps d'attente, en comparaison à l'algorithme de Yousef *et al.* (réduction de 36%) et au plan de feux prédéterminé de TAPASCologne / SUMO (réduction de 48%). La figure 4.11(d) montre que TAPIOCA donne également la taille de file d'attente la plus petite, en moyenne. Finalement, les figures 4.11(e) et 4.11(f) montrent le temps moyen entre deux sélections successives du même mouvement : nous pouvons voir que TAPIOCA accomplit en moyenne le temps le plus petit. Ces résultats nous prouvent notamment que la stratégie adoptée se comporte bien sur des scénarios à grande échelle et réalistes.

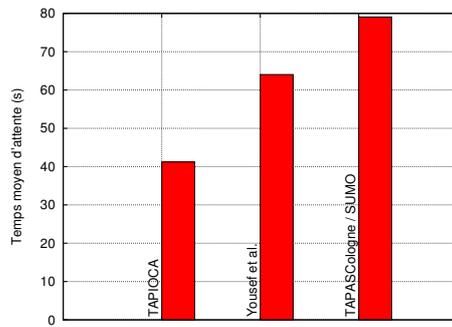
## 4. 6. 3 Scénario 3 : grilles d'intersections

La figure 4.13 montre les performances de TAPIOCA sur une grille de  $64 \times 64$  intersections (4 096 intersections). Chaque simulation dure 7 200 s et voit passer en moyenne 49 000 véhicules, avec un taux d'arrivée de  $\lambda = 8$  véhicules par seconde. D'autres simulations réalisées sur 16, 64, 256, 1 024 et 4 096 intersections, avec une intensité de  $\lambda = 0.5, 1, 2, 4$  et 8 sont disponibles en ligne<sup>3</sup>. Les trajets des véhicules ont été générés aléatoirement par un utilitaire fourni par SUMO.

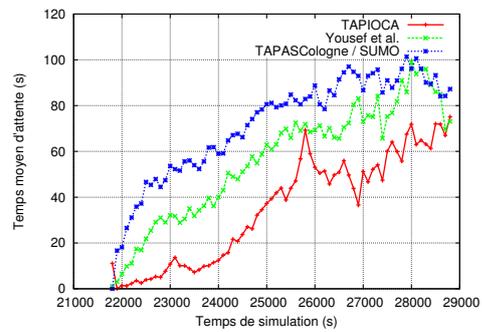
Nous pouvons voir sur les figures 4.13(a), 4.13(b) et 4.13(c) une nette différence entre notre algorithme et les plans de feux prédéterminés de SUMO : notre algorithme crée des vagues vertes et réduit le temps moyen d'attente de 97%. Comparé à Yousef *et al.*, il réduit le temps moyen d'attente de 85%. Pour le cas de SUMO, cette différence est facilement explicable : les plans de feux étant prédéterminés, ils ne sont pas capables de s'adapter au trajet des véhicules, ce qui provoque des arrêts à chaque intersection, et donc une perte de temps. A titre d'exemple, la capture d'écran 4.14

2. <http://sumo.sourceforge.net/doc/current/docs/userdoc/Data/Scenarios/TAPASCologne.html>

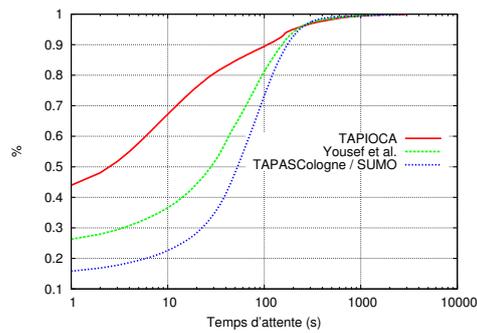
3. <http://tapioca.sfaye.com/Grid/>



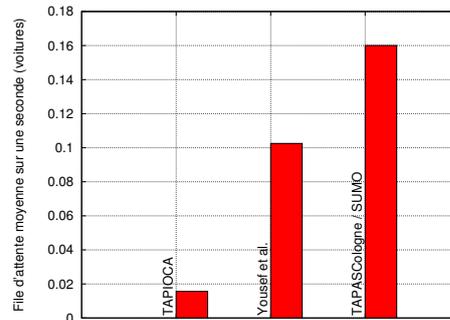
(a) Temps moyen d'attente.



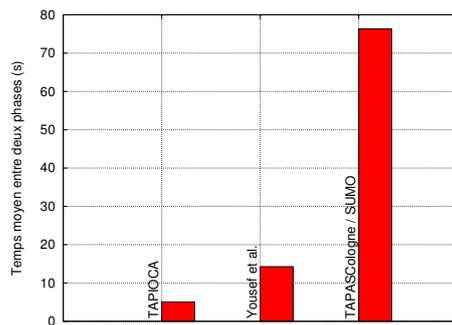
(b) Evolution du temps d'attente.



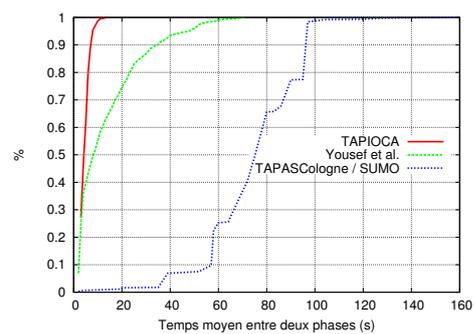
(c) CDF du temps d'attente (échelle logarithmique).



(d) Longueur de file moyenne.



(e) Intervalle moyen entre la sélection d'un mouvement.



(f) CDF de l'intervalle entre la sélection d'un mouvement.

FIGURE 4.11 – Comparaison de performances : le scénario TAPASCologne.

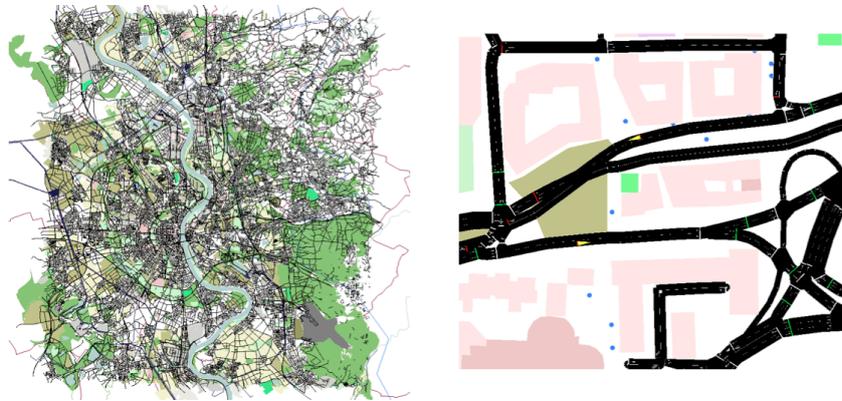
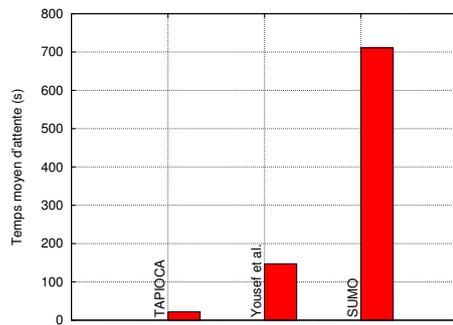
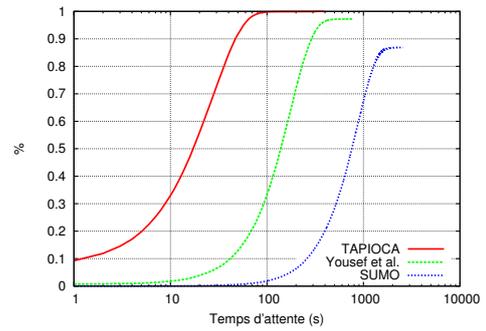


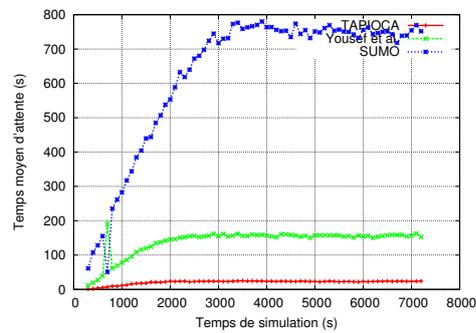
FIGURE 4.12 – Le scénario TAPASCologne.



(a) Temps moyen d'attente.



(b) CDF du temps d'attente (échelle logarithmique).



(c) Evolution du temps d'attente.

FIGURE 4.13 – Comparaison de performances : grilles d'intersections.

montre une simulation confrontant TAPIOCA à un plan de feux généré par SUMO. Le principe est simple : faire parcourir deux ensembles de véhicules d'un bout à l'autre de la grille d'intersections. Au final, les intersections gérées par TAPIOCA savent s'adapter aux arrivées de véhicules, ce qui n'est pas le cas pour un plan de feux statique, occasionnant des arrêts importants (véhicules rouges).

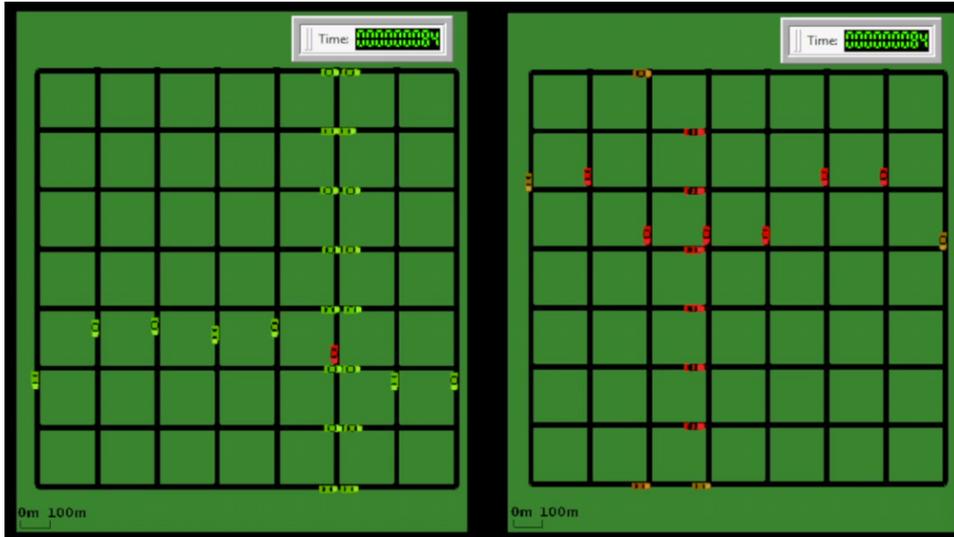


FIGURE 4.14 – Simulation confrontant TAPIOCA (à gauche) à des plans de feux statiques (à droite).

Pour le cas de Yousef *et al.*, nous pouvons l'expliquer par le fait que l'algorithme est conçu pour des temps de cycle trop longs. De plus, TAPIOCA se passe complètement de la notion de cycle et calcule les temps de phase juste avant leur application. Pour le cas de l'algorithme de Yousef *et al.*, cette détermination est faite au moment de la création du cycle : sur de grandes cartes à trafic moyen, où les apparitions et départs de véhicules sont courant, le délai entre une prise de décision et son application doit être court. Des arrêts sont donc provoqués, notamment à cause du manque de réaction de l'algorithme.

## 4. 7 Conclusion

Dans ce chapitre, nous avons proposé et évalué un algorithme adaptatif de contrôle des feux de circulation distribué sur plusieurs intersections et qui utilise un réseau de capteurs sans fil. Sur la base de trois scénarios différents, nous avons montré par simulation que cet algorithme était capable de réaliser un meilleur temps d'attente, temps de trajet et longueur de file d'attente par rapport à une solution prédéterminée, mais également par rapport à des solutions adaptatives. Outre les performances brutes de l'algorithme, ces résultats montrent qu'il existe un intérêt dans la gestion du trafic à l'échelle de l'intersection et de son voisinage direct. Ils nous montrent aussi qu'il est probablement suffisant de se limiter à une distance de communication à un saut (une intersection voisine), tout du moins dans le cas d'un scénario général. L'architecture distribuée permet de réagir rapidement à des situations de congestion

en prenant des décisions locales. Nous avons vu que la synchronisation d'intersections permettait d'améliorer les performances, mais il est probable que des messages puissent être économisées, par exemple lorsque le trafic est stable.

Notre algorithme part du principe que le réseau de communication est idéal. En réalité, la transmission sans fil est peu fiable, et peut entraîner des retards ou des pertes de paquets. Dans le chapitre suivant, nous nous intéressons à la manière dont se comporte TAPIOCA lorsque des informations manquent.



## CHAPITRE 5

---

### Des réseaux de capteurs sans fil au service des infrastructures routières

---

#### Les communications sans fil : vers une réalité

D'une manière générale, TAPIOCA (chapitre 4) peut être utilisé sur n'importe quel support de communication. Toutefois, si les contrôleurs d'intersections sont parfois connectés entre eux par câble (p. ex., fibre optique), le lien entre les capteurs est plus vraisemblablement sans fil. S'affranchir de connexions câblées permet notamment de réduire le temps de déploiement, au même titre que les prix d'installation. Si la plupart des contributions utilisant une telle architecture pour gérer les feux de circulation considèrent un réseau de communication idéal, exempt de pertes ou de collisions, nous étudions, dans ce chapitre, les effets d'une situation plus réaliste. Tout comme dans les chapitres précédents, nous assumons le choix d'utiliser un réseau de capteurs sans fil basé sur le standard IEEE 802.15.4, particulièrement adapté aux dispositifs dont la consommation énergétique peut être un frein.

TAPIOCA repose sur une transmission régulière des véhicules comptabilisés, afin que, après un certain temps (si ce n'est immédiatement), le nœud de décision reçoive les informations. Dans les réseaux de capteurs sans fil, la perte de paquets due à des perturbations ou à des congestions peut toutefois causer de réels problèmes : les données peuvent être manquantes ou retardées et donc peuvent ne pas être à jour quand elles atteignent le nœud de décision. Compte tenu de ces facteurs, nous évaluons dans ce chapitre le comportement de TAPIOCA face à des contraintes de communication réelles. Nous supposons nous reposer sur l'architecture décrite par la figure 3.1 en section 3.2, car elle reste la plus communément utilisée.

#### 5.1 Délais et problèmes de transmission

En nous basant sur une co-simulation – liant une simulation réseau à une simulation du trafic routier – nous évaluons l'architecture du réseau de capteurs sans fil dans la section 5.5 et montrons l'importance des retards qui peuvent être générés,

ainsi que des erreurs dans la comptabilisation des véhicules. En moyenne sur les simulations réalisées, le temps entre la génération et l'utilisation de mesures peut varier de 1 à 46 secondes. L'erreur de comptabilisation observée lorsque le nœud de décision calcule une nouvelle phase peut varier en fonction des retards de 6,6 % à 56,5 %. Cela signifie que le nombre de véhicules connus sur le nœud décision est potentiellement très éloigné du nombre de véhicules présents réellement, au même instant. Cela peut provoquer une perte importante dans le temps moyen d'attente des usagers. En effet, les décisions prises sur le nœud décision sont faussées par les retards de transmission. En conséquence, l'algorithme de TAPIOCA doit être modifié afin de le rendre résistant aux pertes et aux retards, dans une certaine mesure.

## 5.2 Prédiction des données perdues

Nous cherchons tout d'abord à savoir si l'effet des pertes de paquets et des retards pourrait être atténué avec un mécanisme de prédiction des données. Ce mécanisme remplacerait les données manquantes par des valeurs interpolées en cas de besoin.

Les nœuds échangent périodiquement des comptabilisations de véhicules, qui reflètent des situations à des moments différents. Afin de mettre en place un tel mécanisme et permettre aux nœuds d'interpoler les valeurs manquantes, chaque nœud  $SN$  et  $DN$  maintient et transmet une estimation du taux de véhicules qu'il comptabilise.  $\Delta_k$  est la date à laquelle le  $k^{ieme}$  véhicule est détecté par un capteur. Le temps  $\tau_k^{(i)}$  qui sépare deux de ces événements sur le nœud  $i$  peut être approché en utilisant un filtre EWMA (*Exponentially Weighted Moving Average*,  $\alpha$  étant le paramètre de filtre) :

$$\tau_k^{(i)} = \alpha \cdot (\Delta_k - \Delta_{k-1}) + (1 - \alpha) \cdot \tau_{k-1}^{(i)},$$

Tout nœud  $SN$   $i$  peut alors maintenir un taux moyen d'arrivée  $\lambda_i = 1/\tau_k^{(i)}$  qu'il transmet avec des comptabilisations de véhicules réelles à son nœud  $DN$ . Tout nœud  $DN$   $j$  peut également maintenir un taux moyen de départ, filtré,  $\mu_j = 1/\tau_k^{(j)}$ , qu'il transmet à son agrégateur.

### 5.2.1 Prédiction sur le nœud $DN$

Les nœuds  $DN$  s'attendent à recevoir des données périodiquement des nœuds  $SN$ , chaque  $\Delta_{SN}$  secondes, qui devrait être inférieure à une durée de phase. L'effet de la valeur de  $\Delta_{SN}$  est étudié dans la section 5.5.

Les nœuds fonctionnent de manière asynchrone et les décalages de l'horloge et les accès au support de communication peuvent ajouter des retards à la transmission du message. De ce fait, les nœuds  $DN$  définissent pour chaque nœud  $SN$  un *timer* (c.-à-d. une temporisation), dont l'expiration est programmée après  $\delta_{SN}$  secondes et qui est remis à zéro chaque fois qu'un message est reçu du nœud  $SN$ . Lorsque le timer expire, le nœud  $DN$  conclut qu'une donnée a été perdue.

Le nombre d'arrivées sur le nœud  $SN$   $i$  à la date  $t$ ,  $\eta_i^A(t)$ , se déduit simplement de la valeur de ce paramètre à la date  $\tau_{SN_i}$  en ajoutant la différence de temps multipliée par le taux interpolé des arrivées,  $\lambda_i$  :

$$\eta_i^A(t) = \eta_i^A(\tau_{SN_i}) + (t - \tau_{SN_i}) \cdot \lambda_i .$$

Si  $\delta_{SN_i}$  est généralement plus grand que  $\Delta_{SN}$  pour faire face aux pertes de paquets et aux retards excessifs, il peut également être réglé à une valeur inférieure à  $\Delta_{SN}$ , afin de fournir des estimations intermédiaires sur le nombre de véhicules, en le lissant.

Lorsque  $\delta_{SN}$  expire ou quand un message est reçu d'un nœud  $SN$   $i$ , le nœud  $DN$   $j$  situé sur la même voie calcule le nombre total de véhicules présent sur cette voie. Il est noté  $\eta_j = \eta_i^A - \eta_j^D$  et est fonction de  $\eta_i^A$  et des départs totaux mesurés sur  $j$ ,  $\eta_j^D$ . Le nœud  $DN$  envoie ensuite quatre informations à son agrégateur de direction qui gère la direction entrante à laquelle  $DN$  appartient :  $\eta_j$ ,  $\lambda_i$ ,  $\mu_j$  et le temps actuel  $t$ .

L'agrégateur de direction maintient un timer  $\Delta_{DN} \geq \min\{\Delta_{SN}, \delta_{SN}\}$ . Lorsque ce compteur est atteint, il transmet les dernières données agrégées au nœud de décision.

## 5. 2. 2 Prédiction sur le nœud décision

Le second niveau de prédiction est sur le nœud de décision qui estime une valeur récente du nombre de véhicules sur chaque voie, sur la base de la dernière information dont il dispose. Le nœud de décision ajuste d'abord le nombre de véhicules qu'il possède en estimant les nouvelles arrivées sur la voie de chaque nœud  $DN$   $j$ . Si l'on désigne par  $\eta_j(t)$  l'estimation du nombre de véhicules sur la voie commune entre le nœud  $SN$   $i$  et le nœud  $DN$   $j$  à la date actuelle  $t$ , et par  $\eta_j(\tau_j^{DN})$  la dernière comptabilisation d'arrivées reçue depuis le nœud  $DN$   $j$  à la date  $\tau_j^{DN}$ , nous obtenons :

$$\eta_j(t) = \eta_j(\tau_j^{DN}) + (t - \tau_j^{DN} - \tau^s) \cdot \lambda_i .$$

Le temps de démarrage,  $\tau^s$ , ne dispose que d'une influence pour des fréquences très élevées de transmission. Lorsque le feu pour les mouvements concernés a été vert depuis la dernière mesure (et seulement dans ce cas), nous estimons les départs de façon similaire :

$$\eta_j(t) = \max(\eta_j(\tau_j^{DN}) - (t - \tau_j^{DN} - \tau^s) \cdot \mu_j, 0) .$$

## 5. 3 Gestion de plusieurs intersections

Nous choisissons de limiter la transmission des messages de synchronisation pour le seul cas où l'intersection de réception est moins chargée (et donc a un nombre total de véhicules plus faible) que l'intersection d'émission (sec. 4. 4). Un message arrive de manière asynchrone à une intersection qui pourrait avoir à faire face à plusieurs demandes au cours de la même phase. Dans ce cas, nous sélectionnons la stratégie qui

maximise le nombre attendu de véhicules qui traverseraient l'intersection au cours de la phase suivante, en ajoutant une contrainte sur les mouvements. Cette stratégie permet de réduire les probabilités d'encombrement et de perte de paquets sur le canal sans fil, en plus de ne pas transmettre des informations inutilement. Si des requêtes venant d'intersections voisines sont ignorées ou perdues, cela n'a pas grand impact, car les véhicules ne sont pas bloqués et continuent d'arriver et de partir : l'intersection peut les gérer localement. De plus, les messages peuvent être facilement retransmis.

Les opérations et les contraintes sur des intersections adjacentes sont les mêmes que celles décrites en début de chapitre. Cependant, ils sont amplifiés. En effet, les intersections adjacentes sont généralement suffisamment éloignées (chapitre 3), dans ce cas il n'y a plus d'interférences avec le réseau de capteurs, et par conséquent de perte de paquets. TAPIOCA laisse chaque intersection prendre sa propre décision localement – comme si elle était isolée – et essaye de synchroniser les intersections adjacentes avec des messages de synchronisation, la perte de paquets ne compromet donc pas le réseau. Elle provoque, dans le pire des cas, une décision locale au lieu d'une tentative pour créer une vague verte. De plus, les messages de synchronisation sont déjà affectés par notre mécanisme de prédiction, et sont datés : l'intersection réceptrice connaît donc bien le nombre de véhicules qui arrivent à un instant donné.

## 5. 4 Inventaire des messages

La figure 5.1 résume l'algorithme de TAPIOCA et le tableau 5.1 résume toutes les communications qui se produisent entre les capteurs. Les valeurs de la fréquence d'émission et de la taille des paquets sont des valeurs moyennes qui reflètent les simulations effectuées plus loin dans ce chapitre. Cependant, ces valeurs peuvent être adaptées à volonté pour un scénario particulier. Nous décrivons ci-après chaque message existant.

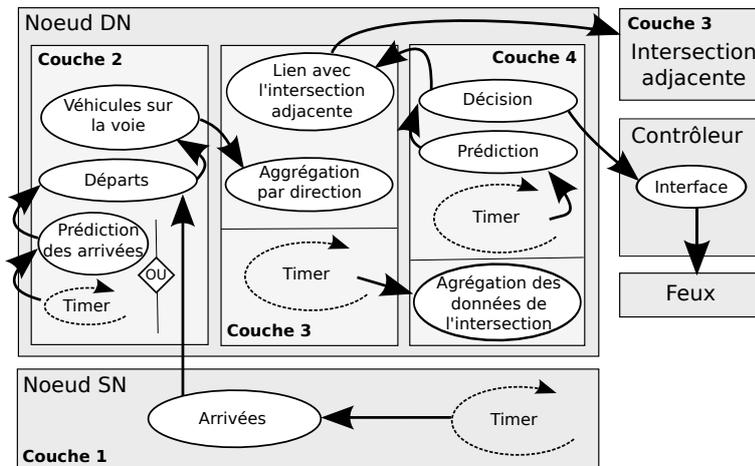


FIGURE 5.1 – Représentation des flux de communication et des principales procédures de notre algorithme.

Messages	Nœuds source et destination	Fréquence	Taille
Arrivées <sup>(a)</sup>	SN → DN	Tous les $\Delta_{SN}$	56 octets
Comptabilisation des véhicules	DN → agreg.	Tous les $\delta_{SN}$ ou lorsque <sup>(a)</sup> arrive	184 octets
Agrégation Comptabilisation des véhicules	agreg. → decision	Tous les $\Delta_{DN}$	472 octets
Caractéristiques de la phase suivante	decision → {contrôleur; agrégateurs}	1 pkt/phase	64 octets
Synchronisation	agreg. → agreg. voisin	1 paquet/phase (asynchrone)	200 octets
Signature des véhicules (optionel)	DN → DN voisin	1 paquet/phase (asynchrone)	108 octets

TABLE 5.1 – Résumé des communications.

Le premier type de messages contient les arrivées comptabilisées par un nœud  $SN$ . Il est envoyé à chaque déclenchement du timer  $\Delta_{SN}$  depuis un nœud  $SN$  au nœud  $DN$  situé en amont sur sa voie. Ce message étant transmis régulièrement, sa perte impliquerait juste une moins grande fraîcheur des données sur le nœud décision, et donc un plus grand taux d'erreur dans la perception du nombre de véhicules sur l'intersection.

Chaque nœud  $DN$  prend en compte le nombre de départs qu'il a comptabilisé afin de pouvoir calculer, à l'aide des arrivées reçues, le nombre de véhicules sur sa voie. Cette donnée est envoyée de façon régulière au nœud agrégateur responsable de sa direction. Ce message est transmis à chaque réception du message précédent ou à l'expiration du timer  $\delta_{SN}$ , provoquant une interpolation des arrivées. En cas de perte de ce message, de la même manière que précédemment, cela impliquerait juste une moins grande fraîcheur des données sur le nœud décision, étant donné que le message est retransmis régulièrement.

Chaque nœud agrégateur envoie, de manière régulière, un message agrégé contenant notamment les comptabilisations des véhicules des voies auxquelles il est rattaché. Ce message est à destination du nœud décision et est envoyé de façon régulière, toutes les  $\Delta_{DN}$  secondes. Sa perte provoque les mêmes effets que les deux cas précédents. Ce message faisant partie du processus de remontées et de comptabilisation du nombre de véhicules, un retard ou une perte n'impliquerait qu'un plus grand taux d'erreur au moment de décider de la prochaine phase, étant donné le délai plus important.

Un nœud décision est responsable du calcul des phases et envoie un message au contrôleur et aux nœuds agrégateurs. Le premier sert à lancer le feu vert. Sa perte implique donc de graves conséquences. Nous pouvons toutefois imaginer qu'un mécanisme de sécurité est présent à l'intérieur des contrôleurs, permettant d'éviter qu'une phase ne continue indéfiniment de se dérouler si aucun message n'est reçu. De même, nous pouvons facilement imaginer un mécanisme d'accusé de réception, qui ne serait pas coûteux étant donné que ce type de message n'est envoyé qu'une fois à chaque phase. Le deuxième message est envoyé aux nœuds agrégateurs afin que chacun puisse communiquer les informations relatives à la nouvelle phase sur les intersections voisines. Si ce message, ou le message envoyé par les nœuds agrégateurs aux intersections voisines, venait à être perdu, l'impact serait uniquement visible sur les performances

de notre algorithme. Les décisions se reposant avant tout sur un mécanisme de décision local, cela empêcherait – ou retarderait – probablement la création de vagues vertes.

## 5. 5 Simulations

### 5. 5. 1 Description des scénarios

Nous évaluons TAPIOCA en utilisant une co-simulation entre SUMO 0.17 et OMNeT++ 4.2 [VH08], liés ensemble à l'aide du framework Veins 2.1<sup>1</sup> [SGD11]. Notons que Veins est à la base conçu pour simuler un réseau véhiculaire. Toutefois, nous l'avons modifié de manière à intégrer un réseau de capteurs sans fil répondant aux caractéristiques qui nous intéressent (chapitre 3), et simuler des effets réseaux proches de la réalité (p. ex., perte de paquets). Les nœuds sont ainsi déployés selon l'architecture décrite sur la figure 3.1 et reposent sur le standard de communication IEEE 802.15.4 (*non-beacon enabled*), utilisant le protocole CSMA. Leur contrôleur d'interface réseau met en œuvre une carte d'interface réseau *Texas Instruments CC 2420 802.15.4*. Enfin, le modèle de propagation utilisé est le même que celui décrit à la section 3. 3. 3.

Les figures 5.2, 5.3, et 5.4 présentent des évaluations réalisées sur une et dix intersections adjacentes, générées aléatoirement par SUMO avec une intensité de trafic faible ( $\lambda = \frac{1}{3}$  véhicules par seconde) et une intensité de trafic supérieure (1 véhicule par seconde). Chaque simulation s'étale sur une durée de 3 600 s.

Nos résultats montrent la comparaison entre cinq méthodes. Tout d'abord, un plan de feux statique généré par SUMO. Ensuite, un algorithme adaptatif issu de la littérature ([YAKS10]) et TAPIOCA : les deux sont centralisés et ne sont pas soumis à des contraintes réseaux, ils fonctionnent indépendamment de l'équipement. Les deux dernières méthodes mettent en œuvre TAPIOCA réparti sur un réseau de capteurs sans fil respectivement avec et sans le mécanisme de prédiction décrit en 5. 2. Ces deux dernières méthodes sont simulées à l'aide de différentes valeurs de timers  $\Delta_{SN}$  et  $\Delta_{DN}$  : 1, 2, 4, 8, 16, 32 et 64 s, soit 49 configurations possibles.

Les résultats complets et d'autres scénarios (par exemple, des arrivées aléatoires ou en croissance) sont disponibles en ligne<sup>2</sup>.

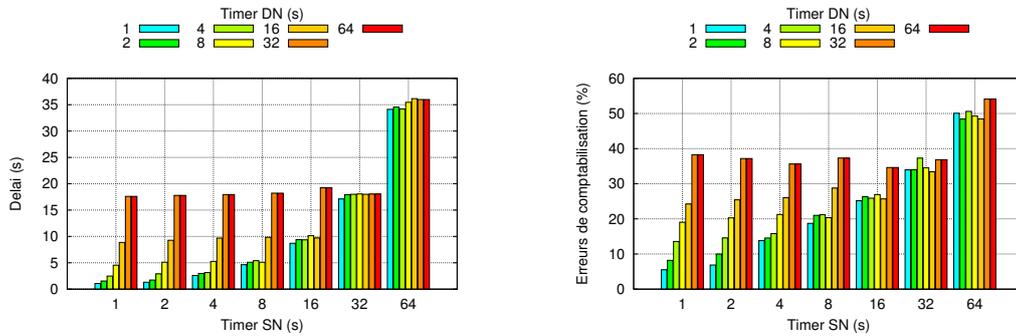
### 5. 5. 2 Evaluation des problèmes réseaux

Les figures en 5.2 nous montrent les problèmes qui sont rencontrés sur le réseau. La figure 5.2(a) représente le délai moyen entre les mesures et leur utilisation au niveau du nœud de décision. La figure 5.2(b) représente les erreurs de comptabilisation de véhicules. Ces résultats montrent que ces valeurs peuvent augmenter rapidement en fonction des valeurs de timers configurées. En configurant par exemple  $\Delta_{SN}$  et

1. <http://veins.car2x.org>

2. <http://tapioca.sfaye.com/WSN/>

$\Delta_{DN}$  à 16 secondes, nous constatons que le délai moyen entre la transmission d'un message et son utilisation est de 10 secondes. Ce délai, sur une simple intersection isolée, entraîne en moyenne 25% d'erreurs sur l'estimation du nombre de véhicules comptabilisés.



(a) Délai moyen entre la détection d'un véhicule et sa prise en compte par le nœud de décision.

(b) Erreur de comptabilisation moyenne.

FIGURE 5.2 – Simulations sur une intersection isolée (1/2).

## 5. 5. 3 Effet du mécanisme de prédiction

### 5. 5. 3. 1 Performances sur une intersection isolée

Selon la figure 5.3(c) et en moyenne sur l'ensemble de nos simulations, nous constatons que  $\Delta_{DN}$  doit être au moins égal au quart de  $\Delta_{SN}$  pour s'assurer que l'estimation ne devienne pas exagérément fausse. Les figures 5.3(a) et 5.3(b) montrent l'efficacité de notre mécanisme de prédiction avec les deux flux d'arrivées décrits en introduction à cette section. L'axe des abscisses représente différentes combinaisons de timers  $\Delta_{SN}$  et de timers  $\Delta_{DN}$ .

Nous constatons en premier lieu que le timer  $\Delta_{DN}$  semble avoir un impact plus significatif que le timer  $\Delta_{SN}$ . En effet, le timer  $\Delta_{DN}$  est celui qui permet l'envoi des données au nœud de décision. Il est donc primordial et influe plus facilement sur les retards : quelle que soit la valeur du timer  $\Delta_{SN}$ , si le timer  $\Delta_{DN}$  est important, un délai sera toujours présent. À l'inverse, si  $\Delta_{DN}$  est faible et que  $\Delta_{SN}$  est élevé, les arrivées sont envoyées à moins grande fréquence, mais sont reçues très rapidement sur le nœud de décision. De plus, une phase peut couvrir cette fréquence d'envoi des arrivées, l'impact est donc plus faible.

Nous constatons également que la charge du système a un impact direct sur le comportement de notre algorithme, et donc sur le temps moyen d'attente des usagers. Pour une charge faible ( $\lambda = 1/3$ ), les temps moyens d'attente sont légers et l'amélioration suggérée par notre mécanisme d'interpolation est légère, mis à part pour des timers élevés. Pour une charge plus importante ( $\lambda = 1$ ), les temps moyens d'attente sont nettement plus élevés et les différences entre un scénario avec et sans mécanisme d'interpolation sont importantes. Nous constatons par ailleurs que notre mécanisme

d'interpolation semble plus efficace lorsque le trafic est plus important, ceci pouvant être expliqué par une plus grande régularité dans les arrivées de nouveaux véhicules.

### 5.5.3.2 Performances sur plusieurs intersections

Pour ces évaluations, nous avons choisi de présenter un scénario intermédiaire, en réglant  $\Delta_{SN} = \Delta_{DN} = 8$  secondes.

Les résultats de simulation présentés sur les figures 5.4(a) et 5.4(b) montrent que toutes les implémentations de TAPIOCA réalisent de meilleurs temps d'attente et de trajets moyens que ceux générés par des plans de feux prédéterminé de SUMO : 93,8 % meilleurs pour la version centralisée, 88 % pour la version réseau de capteurs sans fil avec notre mécanisme de prédiction et 75 % meilleur pour la version de base avec un réseau de capteurs sans fil uniquement.

La figure 5.4(c) montre la proportion de scénarios dans lesquels chaque méthode est la plus efficace, en fonction du nombre de véhicules qui atteignent le meilleur temps d'attente. Nous pouvons voir que TAPIOCA est presque toujours le meilleur choix, et que la mise en œuvre avec un réseau de capteurs sans fil et la méthode de prédiction est la meilleure solution dans 40 % des cas.

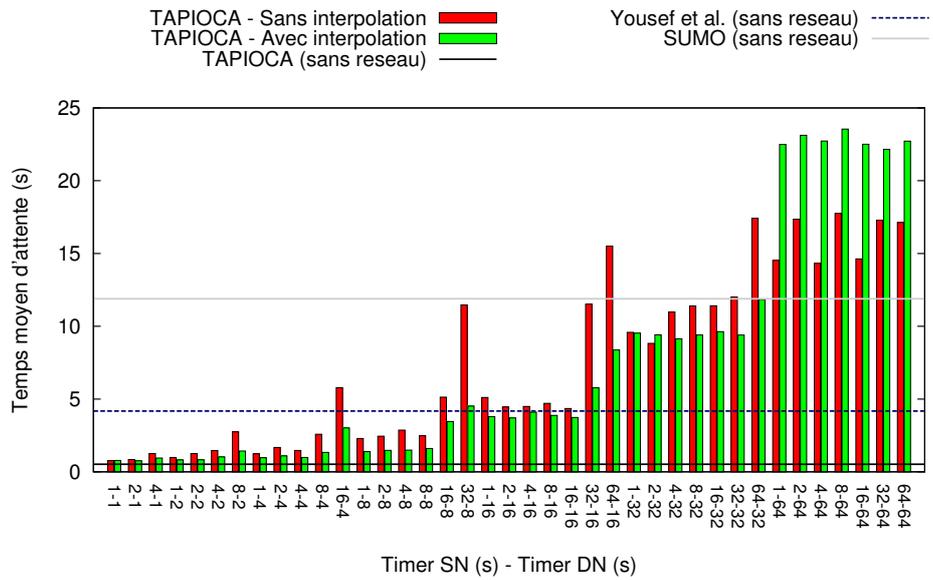
Certains résultats semblent nous indiquer que TAPIOCA, avec un réseau de capteurs sans fil et le mécanisme de prédiction, est plus efficace que la version centralisée (par exemple, sur les figures 5.3(b) et 5.4(c)). Nous pouvons supposer que notre méthode de prédiction, en plus de prendre en considération les retards de transmission, permet d'intégrer – à l'aide du taux d'arrivée – des véhicules que les capteurs n'ont eux-mêmes pas encore détectés.

## 5.6 Conclusion

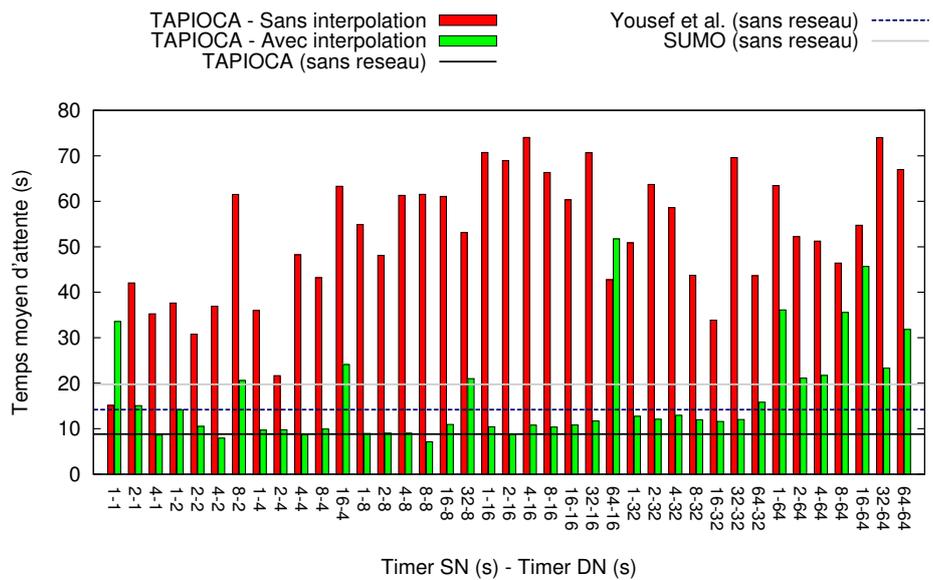
Dans ce chapitre, nous proposons et évaluons un algorithme adaptatif de contrôle des feux de circulation, distribué sur plusieurs intersections et qui utilise un réseau de capteurs sans fil. Nous montrons l'efficacité du réseau de capteurs sans fil et l'intérêt de gérer le trafic localement sur une intersection. Nous proposons notamment un mécanisme de prévision qui nous permet d'alléger les effets provoqués par les retards et les pertes de paquets.

Bien entendu, des pistes restent à aborder. Par exemple, nous pouvons imaginer étudier l'économie d'énergie et limiter les communications, en changeant dynamiquement les timers sur la base de la stabilité du taux d'arrivée et du taux de départ. De plus, nous limitons ici la communication entre les intersections directement voisines. Cependant, nous pourrions bénéficier d'informations provenant d'une plus grande distance, en particulier lors de la construction de vagues vertes, même si nous avons la *sentiment* que cette opération serait plus coûteuse que bénéfique.

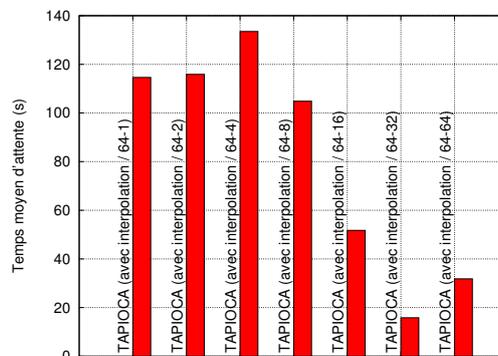
Dans les évaluations réalisées en section 5.5, nous nous sommes reposés sur un faible nombre de scénarios. Le paramètre  $\alpha$  du filtre EWMA a été réglé de façon empirique afin que l'estimation des véhicules soit la plus juste possible. Cela nous a



(a) Temps moyen d'attente pour  $\lambda = \frac{1}{3}$ .

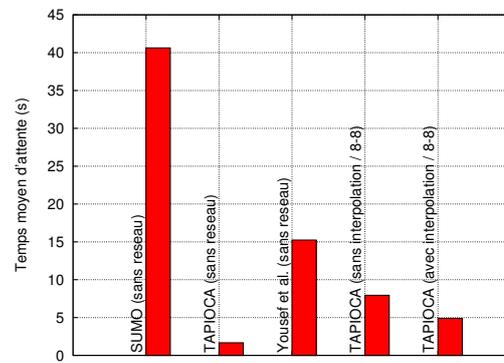


(b) Temps moyen d'attente pour  $\lambda = 1$ .

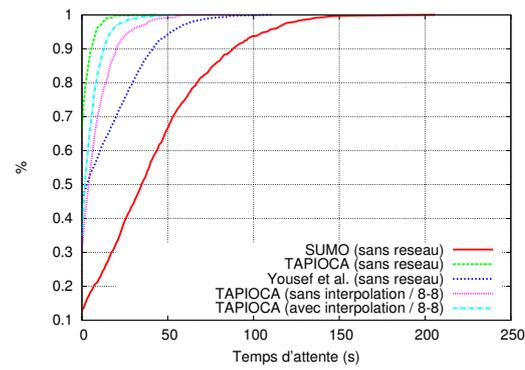


(c) Temps moyen d'attente pour  $\lambda = 1$  et  $\Delta_{DN} \leq \Delta_{SN}$ .

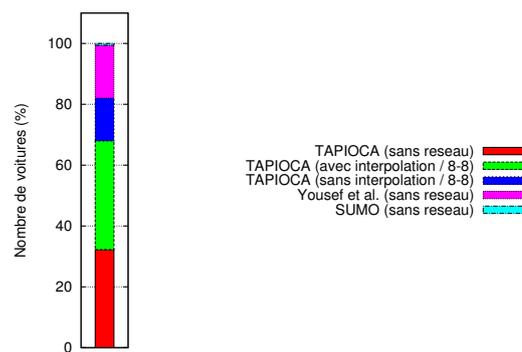
FIGURE 5.3 – Simulations sur une intersection isolée (2/2).



(a) Temps moyen d'attente.



(b) Temps moyen d'attente (CDF).



(c) Répartition de la meilleure stratégie.

FIGURE 5.4 – Simulations sur dix intersections adjacentes avec  $\lambda = \frac{1}{3}$  et  $\Delta_{SN} = \Delta_{DN} = 8$ .

laissé pensé que dans un cas général, ce paramètre variait en fonction de la stabilité du trafic. Selon nos observations, plus le trafic est stable (ou important), plus le paramètre  $\alpha$  peut être élevé. À l'inverse, plus il est instable (ou faible), et plus le paramètre  $\alpha$  peut être bas, afin de réagir plus rapidement aux variations de trafic. Une perspective de ce travail seraient d'étudier ces aspects, et notamment ce deuxième cas, à mesure qu'un paramètre  $\alpha$  trop faible n'irait peut-être pas avec les contraintes réseaux (notamment les pertes de paquets).

Les résultats présentés dans ce chapitre nous laissent penser que TAPIOCA possède davantage de difficultés quand la charge est trop importante. Si le mécanisme d'interpolation suggéré semble bien se comporter dans nos simulations, il nous faut toutefois étudier l'impact que peuvent avoir des scénarios avec un trafic fortement encombré. Dans le chapitre suivant, nous proposons donc d'étudier les cas d'embouteillages, afin de comprendre les différences fondamentales avec des scénarios plus classiques.



## CHAPITRE 6

---

### Caractérisation et gestion des embouteillages

---

#### Des capteurs pour résorber les embouteillages

Dans ce chapitre, nous nous intéressons au cas particulier des réseaux embouteillés. La littérature montre que lorsque la charge dépasse un certain seuil, le comportement du réseau routier est fondamentalement différent du cas fluide. Les métriques que nous utilisons traditionnellement sont moins efficaces. C'est pourquoi nous proposons ici un algorithme alternatif, spécifique à cette situation et tirant parti du réseau de capteurs déployé. Nous baptisons cet algorithme BASALTE (*distributed and Adaptive Signalization ALgorithm for saturated intersections*).

### 6.1 État de l'art des systèmes embouteillés

#### 6.1.1 États de trafic et formation des embouteillages

Lors de la mise en place de routes, le concepteur de voirie détermine leur capacité, et notamment leur nombre de voies, en fonction d'un certain niveau de circulation. Afin de considérer l'ensemble des types de véhicules, l'unité UVP (*unité de véhicule particulier*) est utilisée. Tandis qu'un véhicule léger représente 1 UVP, un poids lourd en représente 2 et un cycle 0,3. Ainsi, une route à quatre voies, sans terre plein central, a une capacité moyenne de 10 000 à 13 000 UVP par jour (ceci pouvant varier en fonction des conditions de circulation, de la pente de la route, etc.).

Traditionnellement, les embouteillages se forment lorsque la quantité de trafic subi par une route approche la capacité de cette route, ou la dépasse. Cette hausse de trafic est souvent constatée aux heures de pointes. La capacité de la route peut également, par exemple, être revue à la baisse si des accidents ou des obstacles matériels s'y trouvent. Enfin, le comportement individualiste des conducteurs mène souvent à des ralentissements. Si un véhicule appartenant à une file ralentit ou n'a pas une vitesse constante, des perturbations sont créées et peuvent se propager de véhicule en véhicule, pour finalement ralentir l'ensemble de la file [Sug+08]. Cet effet

est habituellement appelé effet accordéon, un terme à l'origine popularisé par Denis Niquette, un chroniqueur de circulation de la première chaîne de Radio-Canada [Wik].

Les embouteillages et leurs formations sont décrits en différents niveaux d'états de trafic par la littérature ([LCP75 ; AM97 ; DHS08 ; HMB11]). [LCP75] donne en particulier de bonnes définitions sur les états de la performance du trafic, orientés autour du mécanisme de formation des files de véhicules et représentés sur la figure 6.1.

Lorsque la demande en trafic sur une intersection est nettement inférieure à la capacité de cette intersection (c.-à-d., le débit qu'une intersection est capable d'absorber), aucune file ne se forme lorsqu'un feu vert passe au rouge. Elle est alors décrite comme étant *non-saturée ou sous-saturée*.

Dans le cas où la demande approche ou dépasse la capacité de l'intersection, des files se forment et il reste des véhicules à l'issue d'un feu vert. Dans ce cas, un embouteillage se forme localement. Ce cas de figure est décrit comme étant *saturé*. Lorsque la file qui se forme reste constante au cours du temps, on parle de *saturation stable*. Lorsque cette file a tendance à augmenter progressivement, on parle de *saturation instable*.

Dans les cas où le trafic est supérieur à la capacité d'accueil d'une direction entrante, cette dernière a tendance à se remplir rapidement au fil des cycles, jusqu'à être totalement pleine et bloquer totalement les arrivées venant de l'intersection voisine commune à cette direction. Les effets provoqués par le délai d'attente ne sont alors plus locaux. Dans le cas où l'encombrement d'une intersection influe sur les intersections lui envoyant des véhicules, on parle de *spillback* (ou *déversement-retour*). Ce type de scénario est qualifié de *sursaturé*. Aucune solution de contrôle efficace n'existe pour ce type de cas, toutes faisant office de moyen de prévention [Yan+13]. À ce moment, l'embouteillage formé a tendance à s'étendre, depuis une ou plusieurs intersections jusqu'à potentiellement bloquer temporairement toute ou partie d'une artère principale ou d'un réseau de plusieurs routes, créant une *fermeture* [AAG11], comme représenté sur la figure 6.2. Dans ce cas, le temps de résolution du problème dépend notamment de la taille de la zone congestionnée et de la capacité des intersections, se situant en bordure de cette zone, à évacuer les véhicules.

Dans certains cas extrêmes, nous pouvons faire face à un *interblocage* [CES71], comme représenté par la figure 6.3. La seule solution pour résoudre le problème est que les véhicules reculent ou fassent des manœuvres de dégagement. La plus simple règle à appliquer pour éviter les interblocages serait que les véhicules n'entrent sur une intersection que si ils sont assurés de ne pas y être stoppés.

Pour une direction entrante, un scénario de saturation instable commence à se manifester à partir du moment où  $V/C > 0.9$  [Cj+12], avec  $V/C$  le rapport espace occupé (volume) sur capacité (sec. 2.3.3). Un scénario de sursaturation se manifeste quant à lui lorsque  $V/C > 1$ . Dans ce cas, l'infrastructure subit une demande en trafic qu'elle n'est pas capable d'accueillir. Les méthodes de gestion de feux de circulation gérant ce type de situations ne peuvent qu'alléger la situation, en faisant leur possible pour maximiser le débit (en attendant que la charge ne baisse) et envoyer le trafic vers des zones moins embouteillées.

De là, nous voyons toute la différence entre un scénario sous-saturé et un scénario sursaturé. Le premier peut se fier aisément au délai d'attente des véhicules pour déterminer une stratégie de contrôle des feux de circulation. Dans le deuxième cas, ce délai n'est pas nécessairement le plus représentatif car il ne tient pas compte des facteurs propres aux embouteillages. Les auteurs de [DRK04] montrent, en comparant de nombreux modèles d'analyse des délais (p. ex., files d'attente, Webster [Web58]), que les intersections à feux de circulation ne peuvent pas à la fois maximiser leur capacité et minimiser leur délai lorsqu'elles approchent de conditions saturées ou sursaturées. De cette manière, nous identifions clairement deux situations différentes à traiter, l'une maximisant le débit de véhicules, l'autre minimisant la file de la route critique.

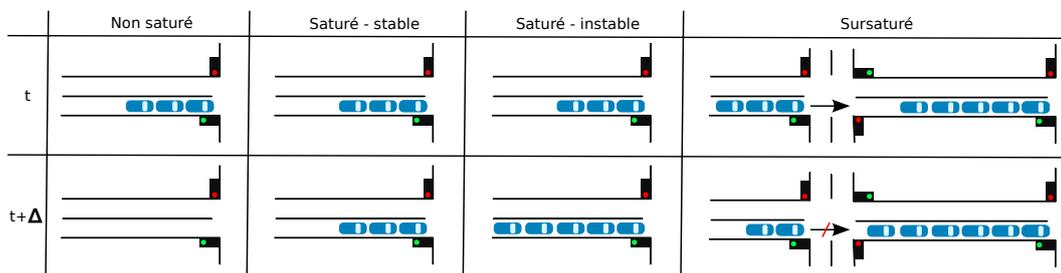


FIGURE 6.1 – Les différents états de trafic.

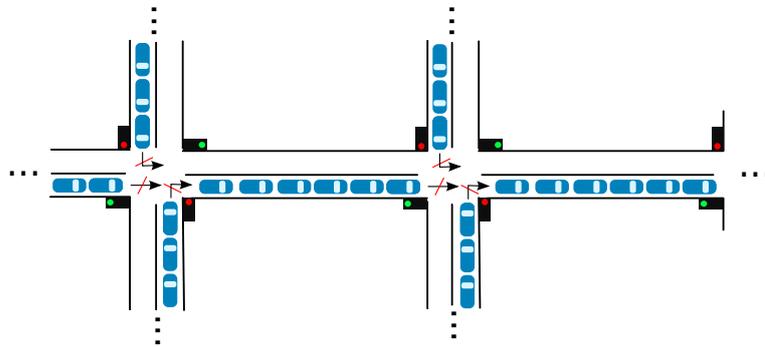


FIGURE 6.2 – Le réseau routier est bloqué, aucune place n'est libre entre les deux intersections. Ce cas de figure étend logiquement l'importance de l'embouteillage.

Dans les précédents chapitres, nous avons étudié une architecture de surveillance capable de comptabiliser le nombre de véhicules et de prendre des décisions locales. Nous pouvons facilement imaginer que cette architecture est capable de détecter les cas où le trafic atteint un certain seuil de saturation, afin d'adapter son comportement en conséquence. Par exemple, en gardant un historique de la taille d'une file et en regardant le nombre de cycles d'augmentation stricte. Dans ce chapitre, nous décrivons un algorithme de gestion des feux de circulation applicable dans les cas où le trafic est embouteillé, en privilégiant les contraintes décrites par la littérature dans la section suivante.

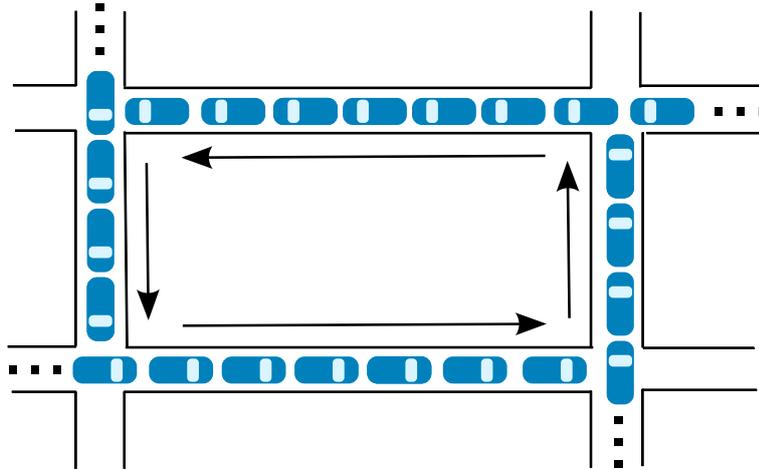


FIGURE 6.3 – Présence d'un interblocage : les véhicules doivent faire marche arrière pour débloquer la situation.

## 6. 1. 2 Contrôle des feux de circulation dans des conditions saturées

Dans des conditions allant jusqu'à la sursaturation, les techniques d'optimisation doivent prendre en compte des critères ayant directement rapport aux caractéristiques de la saturation, tels que la taille des files comme objectif ou contrainte. Les paragraphes ci-dessous résument les travaux, essentiellement développés durant ces 40 dernières années, touchant à l'optimisation des feux de circulation dans des conditions de saturation ou de sursaturation. D'une manière générale, nous constatons deux axes empruntés par la littérature : certains tentent d'allouer des ressources *temporelles* aux véhicules les plus en difficulté, tandis que d'autres tentent de leur allouer des ressources *spatiales* [Yan+13].

### 6. 1. 2. 1 Allocation de ressources temporelles

Les premiers travaux sur le contrôle des feux de circulation dans des conditions saturées ont été réalisés sur des feux prédéterminés par Gazis *et al.*. ([Gaz64 ; GP65]). Sur une intersection, leur idée est d'allouer le temps inutilisé par les phases non saturées sur les phases possédant plus de véhicules, afin de favoriser la décharge des files résiduelles. Ainsi, le délai total peut être réduit en donnant le feu au vert aux files possédant un nombre important de véhicules, plutôt que de le donner aux files possédant peu ou pas de véhicules. De cette manière, le débit sur l'intersection est plus important, les files étant gardées à un niveau acceptable. Cette technique est largement utilisée par des stratégies de contrôle adaptatives ou encore par la NEMA en Amérique du nord. Il est en effet possible pour un contrôleur de feux d'étendre ou de terminer une phase plus tôt, et d'allouer le temps restant à une phase plus saturée [Yan+13]. D'Ans et Gazis [DG76] étendent par la suite le travail de Gazis [Gaz64] à n'importe quel nombre de feux sur la base de plans de feux prédéterminés, en tentant de minimiser le temps perdu par les véhicules au sein des files. Ils prouvent que résoudre le problème de sursaturation implique une allocation

optimale des routes aux conducteurs, un choix favorisant les mauvais critères pouvant facilement mener à une aggravation de la situation.

[Lon68] propose une méthode qui gère en temps réel les files de manière à minimiser le nombre d'intersections secondaires bloquées. L'algorithme fonctionne en faisant varier la durée des feux verts entre un minimum et un maximum afin de garder des files de niveaux comparables. Son algorithme fonctionne sur une intersection isolée et sur un réseau de quatre intersections. Sa philosophie de contrôle est basée sur le fait que les feux de circulation ne peuvent pas libérer les files qui sont dans un état de sursaturation, leur fonction est donc prévenir ce cas de figure en maintenant les files à un niveau acceptable, afin qu'elles ne bloquent pas les intersections adjacentes.

Michalopoulos et Stephanopoulos [MS77] proposent une politique de contrôle sur une, puis sur deux intersections sursaturées avec des rues à sens unique. Leur étude considère, en plus de la taille des files d'attente comme contrainte, le temps de traversée entre deux intersections. Dans [MSS81], les auteurs développent une politique de contrôle en temps réel minimisant le délai total des intersections soumises à des contraintes de longueur de files d'attente. Leur stratégie est plus efficace qu'un plan de feu prédéterminé mais est uniquement applicable sur des intersections isolées.

[LRKS86] et [Rat88] suggèrent deux stratégies de contrôle, l'une jouant sur la durée des feux verts et l'autre contrôlant la longueur des files le long des artères, de manière à limiter la probabilité de spillback. D'autres auteurs tentent également de retarder ou d'éliminer le blocage entre intersections. [LCP75] propose une approche donnant le feu vert à une file lorsque sa taille devient supérieure ou égale à une longueur prédéterminée. [AM97] proposent de minimiser le nombre de spillbacks tout en maximisant le nombre de véhicules qui peuvent passer sur une artère de plusieurs intersections sursaturées et dont les rues de croisement possèdent une demande moyenne.

[HHD12] indique que les vagues vertes créées classiquement par les systèmes de contrôle ne sont pas directement applicables à un scénario sursaturé. En effet, la création d'une bande de progression (sec. 2. 2. 3. 5) repose traditionnellement sur le temps de traversée théorique liant les intersections les unes aux autres. En se servant de cette valeur comme décalage dans un scénario embouteillé, un flux de véhicules arrivant se retrouverait systématiquement ralenti, voire bloqué, par des véhicules déjà présents. Si une intersection reste saturée pendant un délai assez long, une telle coordination est susceptible d'aggraver la situation, en créant des spillbacks sur les intersections voisines émettrices ([LWMH09 ; LC11]). Pour cela, plusieurs auteurs se sont penchés sur l'estimation d'une bande de progression variable en fonction des informations de trafic courants (p. ex. [SG96 ; HTYJ11]). Dans ce cas, une bande de progression, habituellement fixe, est déterminée en fonction du trafic. Le problème est que pour s'assurer que beaucoup de véhicules passent cette bande de progression, elle doit durer longtemps, ce qui tend vers un scénario fixe et réduit l'intérêt d'un système adaptatif (une phase avec un temps de feu grand, les autres avec un plus faible).

Certains auteurs se penchent sur des scénarios limités, et utilisent ou génèrent souvent des plans de feux prédéterminés. Les auteurs de [YLC06] utilisent un algorithme génétique et un plan de feux fixe pour déterminer le séquencage d'un réseau

de trois intersections pour une période de sursaturation de trois minutes. Zhang *et al.* [ZYL10] utilisent un algorithme génétique ainsi qu'un cycle et des feux verts de longueur fixe afin de construire un plan de feux prédéterminé pour une artère traversant 5 intersections surchargées.

Les auteurs de [PMUI00] utilisent un algorithme génétique qu'ils expérimentent sur une artère principale de quatre intersections. Ils utilisent trois fonctions objectifs : minimisation de délai, minimisation de délai modifié avec une fonction de pénalité, et maximisation de débit. Ils prouvent que l'utilisation du premier objectif produit un temps de feu supérieur comparé aux autres stratégies et à la stratégie proposée par TRANSYT-7F<sup>1</sup>, un simulateur macroscopique de trafic routier et d'optimisation des temps de feux.

Chang *et al.* [CBX10] tentent de maximiser le débit tout en gérant les files du système. Leur modèle est testé sur une artère de deux intersections et augmente l'utilisation de la capacité d'une intersection, ce qui génère un délai d'attente plus bas de 22% en comparaison à TRANSYT-7F.

Les auteurs de [CM00] proposent d'ajuster le calcul du temps de cycle formulé par Webster [Web58] à différents niveaux de saturation du trafic.

Certains auteurs s'essayent également à des méthodes moins habituelles. Putha *et al.* [PQ10] utilisent un algorithme de colonie de fourmis pour résoudre le problème de la coordination de signaux pour un réseau sursaturé. Robertson et Bretherton [RB74] et Gartner [Gar83] utilisent des méthodes par programmation dynamique, décomposant des problèmes complexes en plusieurs problèmes plus simple mais souffrant de problèmes de passage à l'échelle, et demandant de connaître des données précises et nombreuses au cours du temps.

### 6. 1. 2. 2 Allocation de ressources spatiales

D'autres auteurs se sont penchés sur l'utilisation de la place disponible sur chaque intersection. [Gor69] propose un algorithme tentant de rendre égal pour chaque phase le rapport entre la taille des files et la capacité maximale. De récentes études [BUB06] montrent que l'utilisation de la place libre comme critère peut retarder la propagation d'un embouteillage, en coupant une phase lorsqu'il y a des restrictions sur les intersections voisines. Ceci peut empêcher les effets spillback [AAG11] : lorsqu'un capteur n'enregistre plus de véhicule pendant le feu vert, c'est que les véhicules n'ont nul part où aller. [Yan+13] montre par simulation qu'une méthode privilégiant des critères de place génère un temps d'attente moins élevé que des méthodes s'appuyant sur une bande de progression classique et de temps.

### 6. 1. 3 Limites de la littérature

L'une des principales limites de la littérature repose sur le fait que les systèmes proposés soient généralement centralisés, trop complexes ou trop éloignés des systèmes actuels pour être abordés dans la réalité (p. ex., colonies de fourmis). De nom-

1. <http://mctrans.ce.ufl.edu/featured/TRANSYT-7F/>

breux auteurs cités précédemment considèrent en effet pouvoir agir simultanément sur toutes les intersections d'un réseau, souvent par l'intermédiaire d'algorithmes génétiques ou de méthodes issues de l'optimisation combinatoire. Outre le fait que ces approches seraient difficilement tolérantes aux fautes et implémentables, il nous faut souligner la complexité des calculs et l'importance des données à conserver afin de rendre le système efficace, surtout dans des cas de contrôle adaptatif ou en temps réel ([Lon68 ; Gor69 ; LCP75]). Les algorithmes génétiques imposent par exemple de garder en mémoire une grande quantité d'informations. Les méthodes propres à l'optimisation combinatoire, souvent complexes, nécessiteraient des calculs longs, les paramètres étant nombreux et l'optimisation d'une séquence de feux n'étant pas triviale [HMB11]. De plus, l'échange quasi permanent des informations de trafic et de décisions rendrait un tel système non viable sur une échelle plus large que les scénarios étudiés. Dans le cas de systèmes prédéterminés, si la constitution ou l'utilisation de plans de feux statiques pour une demande connue ne pose pas de problème de dimensionnement, le manque de temps réel et d'adaptation pose problème.

Ensuite, notons pour un certain nombre d'auteurs le manque de généralisation de leur algorithme, qui est conçu pour un scénario particulier ou qui ne considère pas des métriques importantes dans des cas d'embouteillage. Par exemple, Gazis [Gaz64] suppose que le temps de traversée ainsi que le stockage des véhicules entre deux intersections est négligeable. En d'autres termes, il néglige les contraintes spatiales de distance entre les carrefours et la place prise par les files d'attente. Singh et Tamura [ST74] proposent une méthode ne prenant pas en compte les interférences entre les véhicules arrivant et ceux déjà présents sur une intersection. Leur étude est donc limitée, car lorsqu'une intersection déborde sur une autre, il nous faut pouvoir identifier et traiter la situation. [MS77] considère davantage de métriques, mais ne les a appliqué qu'à une route à sens unique avec un séquençage limité. [MSS81] propose une politique de contrôle en temps réel, testée et validée mais uniquement valable pour les intersections isolées.

En résumé, de nombreuses recherches dans le domaine du contrôle du trafic routier pour des environnements saturés reposent sur des hypothèses trop restrictives pour être appliquées dans un système réel [AM97]. Les modèles sont souvent régis par des contraintes statiques, ou à l'inverse sont trop complexes pour passer l'échelle de la réalité. Ils manquent souvent de généralisation et prennent parfois en compte des hypothèses irréalistes de nos jours.

## 6. 2 *BASALTE* : a *distriButed* and *Adaptive Signalization ALgorithm* for *saTurated intErsections*

Dans cette partie, nous proposons une solution de contrôle des feux de circulation sur des intersections saturées, baptisée *BASALTE* (*distriButed and Adaptive Signalization ALgorithm for saTurated intErsections*). Cet algorithme est distribué sur chaque intersection et se sert des constats faits par la littérature pour proposer

une méthodologie de prévention et de déconstruction des embouteillages. La distribution des capteurs permet une réaction rapide aux embouteillages, tandis que la résolution du problème peut être gérée indépendamment par plusieurs points.

### 6.3 Architecture

Cet algorithme se base sur l'architecture entièrement adaptative décrite en section 3.2. En comparaison aux autres architectures, de nouveaux nœuds sont utilisés sur les voies sortantes de l'intersection : les nœuds *AN*. Ces nœuds permettent notamment de savoir où les véhicules vont une fois sortis de l'intersection, ce qui nous semble essentiel afin de mesurer et agir sur un embouteillage. Nous définissons deux couches de communication nous permettant de mener à bien les différentes opérations de notre algorithme, comme représenté sur la figure 6.4. Bien que le nombre de capteurs nécessaire soit plus important que pour TAPIOCA, cette nouvelle hiérarchie est toutefois plus légère, car nous souhaitons exploiter la capacité de chaque nœud à pouvoir décider du changement des feux. Cette intuition nous permettrait, *a priori*, de gagner en réactivité et en souplesse.

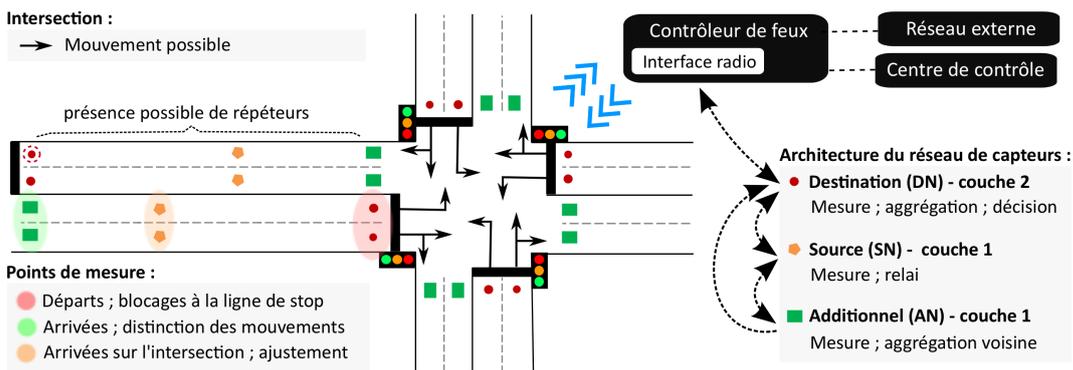


FIGURE 6.4 – Architecture de BASALTE.

La logique que nous souhaitons expérimenter est basée sur le fait que chaque nœud *DN* est responsable des mouvements autorisés par sa voie. Afin de pouvoir prendre une décision, chaque nœud *DN* (couche 2) reçoit les informations des nœuds *SN* et *AN* (couche 1) qui sont pertinentes pour lui. Les nœuds *DN* ont la possibilité de communiquer entre eux afin de se transmettre des informations, ainsi qu'avec un nœud intermédiaire, qui arrange et tri les requêtes reçues. Ce nœud intermédiaire pourrait être un nœud *DN* élu parmi ceux existant, mais nous supposons par la suite qu'il s'agit d'une interface reliée au contrôleur de feux, ou le contrôleur de feux lui-même. Le détail de l'algorithme associé est donné dans la section suivante.

La figure 6.5 représente le diagramme de collaboration de BASALTE, décrivant les relations existantes entre les nœuds et les opérations de l'algorithme, que nous détaillons dans la section suivante.

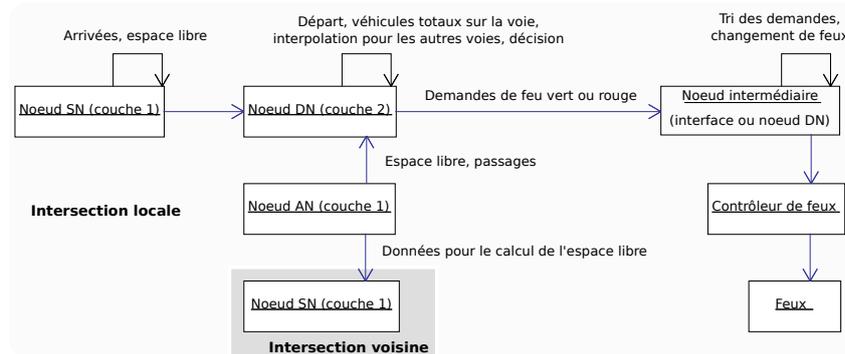


FIGURE 6.5 – Diagramme de collaboration de BASALTE pour le cas d'un réseau multi-sauts.

## 6. 4 Algorithme de contrôle des feux de circulation

Nous décrivons ici BASALTE en trois étapes. Tout d'abord, les nœuds *DN* acquièrent une vision de l'état du carrefour au moyen des mesures des différents capteurs (sec. 6. 4. 1). Ils décident alors de demander au contrôleur le passage d'un feu au rouge ou au vert en fonction de leur perception (sec. 6. 4. 2). Le contrôleur reçoit toutes les demandes et prend finalement une décision en fonction du potentiel d'écoulement du trafic (sec. 6. 4. 3).

### 6. 4. 1 Étape 1 : métriques

Dans une première phase, les nœuds *SN* et *AN* collectent le nombre de véhicules qu'ils voient passer. Ils transmettent ensuite, de manière régulière, ces informations à chacun des nœuds *DN* qui leur est associé. Pour un nœud *SN*, le nœud *DN* associé est celui situé sur la même voie que lui. Pour un nœud *AN*, les nœuds *DN* associés sont les nœuds dont les voies ont la possibilité de mener à la voie du nœud *AN*.

Le coût des différentes tâches collaboratives entre les capteurs est évalué dans le tableau 6.1. Par exemple, le nombre de véhicules sur une voie peut être évalué au minimum en comparant les données issues d'un nœud *SN* et d'un nœud *DN*, mais peut être évalué plus précisément en ajoutant les données provenant de nœuds *AN*. En effet, une file peut être plus grande que la distance séparant les nœuds *SN* des nœuds *DN*.

L'ensemble des métriques utilisées dans ce chapitre – ainsi que les explications sur la manière de les obtenir – sont décrits dans les paragraphes suivants. En particulier, chaque nœud *DN* reçoit et stocke, au fil du temps, les valeurs des métriques qui sont propres à sa voie. Ces données sont agrégées puis partagées aux nœuds *DN* des voies adjacentes qui possèdent des mouvements communs. Cette coopération permet à chacun d'évaluer le nombre de véhicules présents sur les voies de ses mouvements, ainsi que l'espace disponible sur les intersections voisines atteignables. À l'aide de

ces valeurs, le changement des feux peut être géré sur chaque nœud DN au nom des mouvements qui sont possibles sur sa voie.

Métriques de voies	Coopération minimale	Coopération maximale
Nombre de véhicules	1 SN + 1 DN	1 AN + 1 SN + 1 DN
Espace libre	1 SN + 1 DN	1 AN + 1 SN + 1 DN
Taux moyen d'arrivée (EWMA)	1 AN	
Taux de véhicules moyen sur une voie empruntant le mouvement $(a, b)$ (EWMA)	1 DN + nombre de voies en destination de $b$	
Métriques de mouvements	Coopération minimale	Coopération maximale
Nombre de véhicules	1 DN	Tous les DN de $a$
Espace libre	<i>Idem</i>	

TABLE 6.1 – Principales métriques utilisées.

#### 6. 4. 1. 1 Taille d'une file d'attente

##### Sur une voie

Chaque nœud  $SN$  transmet de façon régulière (p. ex., toutes les minutes) deux informations à son nœud  $DN$ , situé sur une voie commune que nous notons  $x$ . D'une part, les véhicules qu'il a comptabilisé, et d'autre part le taux d'arrivée moyen au temps  $t$ , noté  $\lambda^x(t)$ , qui est ajusté sur la base d'un modèle EWMA. À la réception au temps  $t$ , un nœud  $DN$  peut facilement calculer le nombre de véhicules présents sur sa voie, noté  $\eta^x(t)$ . Il lui suffit en effet de soustraire le nombre de véhicules qu'il a lui-même enregistré (c.-à-d., départs) au nombre de véhicules transmis par le nœud  $SN$ .

Nous nous plaçons ici dans un scénario où les données sont transmises en quasi temps-réel. Toutefois, comme nous le constatons au chapitre précédent, un certain temps peut passer entre le moment où une comptabilisation est mesurée et le moment où elle est transmise, puis utilisée. De plus, les données provenant des capteurs peuvent être perdues, retardées ou tous simplement non disponibles si la fréquence d'acquisition est faible. Les nœuds DN utilisent alors un mécanisme d'interpolation afin de compenser ce manque. À l'aide des informations acquises, il est facile pour un nœud  $DN$  d'estimer l'évolution de la file d'attente sur sa voie. Si nous définissons  $\tau^\Delta$  comme étant le temps écoulé depuis la dernière mise à jour de la file d'attente par le nœud, nous obtenons au temps  $t$ , quel que soit  $x$  :

$$\eta^x(t) = \eta^x(t - \tau^\Delta) + \tau^\Delta \cdot \lambda^x(t - \tau^\Delta) - \eta_D^x, \quad (6.1)$$

avec  $\eta_D^x$  le nombre de départs comptabilisés depuis la dernière mise à jour de  $\eta^x$ .

##### Sur un mouvement

Une fois la file d'attente de la voie courante calculée, l'objectif pour le nœud  $DN$  est d'estimer le nombre de véhicules qui veulent emprunter les mouvements que sa voie propose. En effet, tout comme pour TAPIOCA, nous considérons que le changement d'état d'un feu doit être fait par rapport à un ensemble de mouvements, et non par rapport à une file en particulier. Fonctionner par mouvement permet en

particulier de contrôler la destination des véhicules ainsi que d'éviter les interblocages sur l'intersection.

Nous définissons la taille d'une file d'attente pour un mouvement  $(a, b)$  comme étant égale à la somme des files d'attente de chaque voie appartenant à la direction entrante  $a$  :

$$\eta^{(a,b)}(t) = \sum_{x \in a} \eta^x(t) \cdot \alpha_x^{(a,b)}, \quad (6.2)$$

où  $\alpha_x^{(a,b)}$  représente le taux moyen de véhicules situés sur la voie  $x$  qui empruntent le mouvement  $(a, b)$ . Par défaut, si une voie accueille un seul mouvement, nous notons alors  $\alpha_x^{(a,b)} = 1$ . Dans le cas où une voie permet aux véhicules d'aller vers  $D > 1$  destinations (c.-à-d. plusieurs mouvements possibles), nous appliquons  $\alpha_x^{(a,b)} = 1/D$ . Si l'architecture le permet, ce taux peut éventuellement être ajusté. Ceci est rendu possible avec l'utilisation des nœuds  $AN$ , qui peuvent échanger les signatures électromagnétiques des véhicules avec les nœuds  $DN$ , afin de savoir quels chemins sont empruntés et à quelle fréquence.

Étant donné que nous souhaitons rendre chaque nœud  $DN$  responsable d'une potentielle décision, l'estimation de  $\eta^{(a,b)}(t)$  doit se faire directement sur chaque nœud  $DN$ . Pour se faire, nous choisissons, à chaque fin de phase, de faire communiquer les nœuds  $DN$  entre eux, sur chaque direction, pour échanger les informations que chacun possède sur sa file.

Le temps de latence entre les différentes communications pouvant être important, chaque nœud  $DN$  peut maintenir, de façon indépendante, l'évolution des files représentatives de ses mouvements. Pour cela, il lui suffit de mettre à jour la taille des files d'attente pour chacune des voies connues. Pour sa voie, il lui suffit d'appliquer la méthode décrite plus haut en tenant compte du taux d'arrivée et des départs comptabilisés. Pour les autres voies, cette opération est également possible. D'une part, la mise à jour des nouvelles arrivées peut se faire en ayant connaissance du taux moyen d'arrivée pour chaque voie, communiquée par les nœuds  $DN$  voisins. D'autre part, les départs peuvent être pris en compte avec un taux moyen de départ ou en se référant directement à la voie sur laquelle le nœud  $DN$  se situe. Nous pouvons par exemple supposer qu'à chaque départ enregistré sur le nœud  $DN$ , un départ se produit également sur les autres voies, ou que le débit est similaire.

### 6. 4. 1. 2 Espace libre

La capacité totale de la voie  $x$  est invariante dans le temps, et est notée  $C^x$ . Elle est exprimée en nombre de véhicules et est égale à la longueur de la route à laquelle appartient  $x$ , divisé par la longueur moyenne d'un véhicule  $L_{veh}$ , écarts compris.  $L_{veh}$  peut être fixé (p. ex. 6 m estimés [Gor+05]) ou évalué à l'aide de capteurs au sol. Cette manière de procéder est simple, mais nous pouvons également imaginer que les capteurs soient capables de différencier chaque type de véhicule, afin de les compter en UVP (sec. 6. 1), ce qui nous permettrait d'obtenir des valeurs plus justes (approximation selon le type de véhicule).

A partir de cette information, nous pouvons déterminer l'espace libre sur une route comme étant la différence entre sa capacité totale et le nombre total de véhicules qu'elle contient. Cette information peut être transmise de manière régulière aux nœuds  $AN$  de cette route, qui comptabilisent les nouveaux véhicules allant l'occuper. Chaque nœud  $AN$  envoie, de façon périodique à chaque nœud  $DN$  de son intersection, la place disponible, avec d'éventuelles informations supplémentaires telles que le taux de passage. Ceci afin que les nœuds  $DN$  puissent interpoler les futures estimations de cette valeur.

Notons que, comme indiqué au chapitre 2, d'autres approches sont également envisageables : une caméra peut être utilisée au niveau d'un feu pour estimer la longueur, en mètres, de la file d'attente, et ainsi déterminer la place restante. Des capteurs peuvent également être positionnés sur la chaussée, séquentiellement ou sur rails, afin de définir des seuils de remplissage.

## 6. 4. 2 Étape 2 : agrégation et prise de décision distribuée

Un nœud  $DN$  peut donc gérer temporairement et de manière autonome l'évolution des métriques dont il a connaissance. Fort de ces données et estimations, chaque nœud  $DN$  est alors capable d'évaluer l'impact d'un mouvement  $(a, b)$  sur le carrefour.

L'algorithme 6.1 décrit la procédure qui permet à un nœud  $DN$  de savoir si le feu auquel il est rattaché est vert ou non, et d'envoyer en conséquence des demandes de changement de feux au contrôleur. Les nœuds  $DN$  demandent explicitement le passage de feu au vert lorsque des véhicules sont en attente sur sa voie ou ses mouvements, et que la route de destination n'est pas pleine (lignes 4 à 10). Ils demandent le feu au rouge dans le cas contraire (lignes 11 à 14). Leur demande est assortie de leur évaluation du nombre de véhicules qui pourront traverser le carrefour (non représenté dans l'algorithme 6.1).

Contrairement à la plupart des travaux de la littérature, l'adaptation au trafic se fait de façon dynamique, et ne passe pas par l'estimation d'un temps de feu vert sur un ensemble prédéterminé de phases. Il s'agit également un bon moyen pour détecter les embouteillages. En effet, si aucun véhicule n'est en mouvement sur un nœud  $DN$ , alors soit le feu est rouge, soit les véhicules sont bloqués.

**Algorithme 6.1:** algorithme de prise de décision sur un nœud *DN*


---

**Paramètres :** *mouvements*, liste des mouvements de l'intersection.  
**Variables :**  $\eta^{a,b}$ , nombre de véhicules représentatifs du mouvement  $(a, b)$ ;  $\epsilon^a$ , espace libre sur la route  $a$ ; *booleenVert*, booléen identifiant si le feu est vert (1) ou non (0).

```

1 booleenVert ← 0 // Initialisation de la variable globale
2 Procédure DecisionDN() :
   /* Aucun véhicule n'a été détecté depuis le dernier appel de la procédure.
   Trois possibilités : le feu est rouge, des véhicules sont arrêtés, ou
   aucun véhicule n'est présent. */
3 si derniersVehicules() == 0 alors
   /* Cas 1 : le feu est rouge. Les métriques sont mises à jour et si
   possible, une demande pour passer le feu au vert est envoyée. */
4   si booleenVert == 0 alors
5     pour chaque  $(a, b) \in \textit{mouvements}$  faire
6       interpolation(a,b);
7       si  $\epsilon^b > 0$  et  $\eta^{(a,b)} > 0$  alors
8         demandeVert(a,b);
9       finsi
10    finprch
11   /* Cas 2 : le feu est vert. Deux possibilités : (1) les véhicules sont
   stoppés par un embouteillage, ou (2) il n'y a plus de véhicules.
   Dans tous les cas, le passage du feu au rouge est demandé. */
12   sinon si booleenVert == 1 alors
13     demandeRouge(a,b);
14   finsi
15   booleenVert ← 0;
16   /* Au moins un véhicule a été détecté depuis le dernier appel de la
   procédure. Le feu est donc encore considéré comme vert. Aucune action
   n'est à effectuer. */
17   sinon
18     booleenVert ← 1;
19   finsi
20 finproc

```

---

**interpolation(a,b)** : met à jour les métriques du mouvement  $(a, b)$  conformément à la section 6. 4. 1.

**derniersVehicules()** : retourne le nombre de véhicules qui sont passés depuis la dernier appel de la procédure.

**demandeVert(a,b)** : le nœud *DN* envoie une demande de passage au feu vert au contrôleur.

**demandeRouge(a,b)** : le nœud *DN* envoie une demande de passage au feu rouge au contrôleur.

---

### 6. 4. 3 Étape 3 : mise en place des feux de circulation

Le contrôleur, quant-à-lui, récupère l'ensemble des demandes et prend, à intervalles réguliers, une décision sur les feux qui doivent être verts ou rouges. Si plusieurs demandes concernent le même mouvement, il ne prend en compte que la dernière. Il sélectionne ensuite le mouvement qui représente le plus grand potentiel d'écoulement de véhicules. Ce mouvement doit être compatible avec les mouvements actifs et ne doit pas avoir déjà été sélectionné pour l'itération du cycle en cours de déroulement. Un mouvement avec une grande capacité d'écoulement entrant en conflit avec un mouvement actif devra patienter, ce qui permet de garantir une certaine stabilité. Le feu étant en place, cette procédure est répétée pour chaque nouvelle demande, dans le cas où d'autres mouvements pourraient se greffer à ceux en place. Un feu deviendra rouge lorsque plus aucun véhicule ne sera présent sur sa voie incidente, lorsqu'un message d'arrêt sera reçu, ou alors lorsqu'il sera resté vert durant un temps prédéfini, pris par défaut au temps de cycle maximal en France : 120 secondes. Dans le cas d'un passage au rouge, cette procédure est répétée pour compléter les mouvements restants.

L'algorithme ainsi défini est dynamique et réactif. Il raisonne mouvement par mouvement plutôt que phase par phase et diffère de la littérature qui se base plutôt sur la sélection de la phase prédéterminée la plus adaptée. Cette méthode présente l'avantage de permettre des combinaisons de mouvements plus souples, ne sélectionnant pas les mouvements dont les voies de destination sont saturées. En revanche, sa performance est très dépendante de celle du réseau de communication sous-jacent. En effet, des pertes de paquets ou des retards auront un impact sur la pertinence des données utilisées et donc sur son efficacité.

### 6. 4. 4 Résumé des messages

Le tableau 6.2 décrit les messages qui sont transmis sur le réseau. Comme nous le constatons, si les nœuds AN et SN sont sollicités régulièrement pour diffuser, entre autres, les nouvelles arrivées, les nœuds DN envoient des informations essentiellement à chaque fin de phase.

Message	Nœud source et destination	Fréquence	Taille
Ordre de changement de feu	DN → Interface	Variable	~15 octets
(1) Ordre d'arrêt de feu et données locales	DN → Interface	Variable	~40 octets
(2) Mise à jour des données sur chaque DN	Interface → tous les DN	A chaque fin de phase, en réponse à (1)	~35 octets
(3) Remontée pour le calcul de l'espace libre	Interface ou DN → AN (intersection voisine)	A la suite de (1) ou de (2)	~15 octets
Espace libre	AN → DN	A la suite de (3)	~15 octets
Arrivées lointaines	AN → SN	Chaque minute	~15 octets
Arrivées proches	SN → DN	Chaque minute	~15 octets
Calcul du coefficient $\alpha$	tous les AN d'une destination → DN	Chaque minute	Variable

TABLE 6.2 – Messages transmis sur le réseau, fréquence et taille des paquets.

## 6. 5 Simulations

Nous évaluons notre algorithme par une co-simulation entre le simulateur de trafic routier SUMO 0.19 et le simulateur réseau OMNeT++ 4.2, liés à l'aide du framework Veins 2.1. Les nœuds sont déployés conformément à l'architecture entièrement adaptative décrite en 3. 2 et utilisent le protocole IEEE 802.15.4 (*non-beacon*, *CSMA*). La fréquence de transmission des messages est configuré conformément au tableau 6.2.

Nos simulations sont réalisées sur un réseau de trois intersections généré aléatoirement par SUMO et subissant un taux d'arrivée de  $\lambda$  véhicules par seconde pendant 3 600 secondes. Les trajets des véhicules ont été générés aléatoirement par un utilitaire fourni par SUMO. La figure 6.6(a) représente l'évolution du temps d'attente des véhicules en fonction de la fréquence d'échantillonnage des capteurs dans le cas d'une charge faible ( $\lambda = 1$  véhicule/s). La figure 6.6(b) représente la même évolution dans un cas de forte charge ( $\lambda = 2$  véhicule/s, conduisant à un embouteillage). Notre algorithme avec et sans interpolation est comparé avec un cas idéal dans lequel le réseau de communication est parfait (sans délai ni perte) et au plan de feu prédéterminé par SUMO en guise de référence. La figure 6.6(c) représente le débit d'émission moyen de chaque capteur en fonction de la fréquence d'échantillonnage et les figures 6.7(a) et 6.7(b) représentent les temps de trajet médians des véhicules pour différentes fréquences d'échantillonnage, avec et sans interpolation. Tout comme pour les autres chapitres, des résultats plus complets sont disponibles en ligne<sup>2</sup>.

Ces résultats montrent qu'il existe une fréquence d'acquisition optimale. Une fréquence plus rapide provoque des collisions et des pertes de paquets que le mécanisme d'interpolation peut compenser en partie. Une fréquence plus lente conduit à une estimation imprécise de l'état du réseau. Il est aussi intéressant de remarquer que le mécanisme d'interpolation est efficace même lorsque l'algorithme fonctionne à sa fréquence optimale. La comparaison entre les deux niveaux de charge montre que l'intérêt du mécanisme d'interpolation augmente dans les cas embouteillés. Les taux d'arrivées et de départs ayant tendance à être plus stables dans ce derniers cas, ce mécanisme conduit à des valeurs plus proches de la réalité, en comparaison à un scénario plus léger où des files en réalité vides peuvent être sélectionnées.

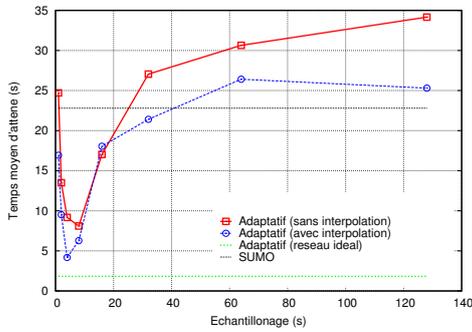
## 6. 6 Perspectives

Ce travail étant l'un des derniers travaux effectués au cours de cette thèse, de nombreux autres aspects sont abordables. Toutefois, les principes proposés – nés de l'intuition de vouloir pousser la collaboration entre les capteurs à son maximum – ont le potentiel de pouvoir réagir très finement et rapidement à des situations d'embouteillages.

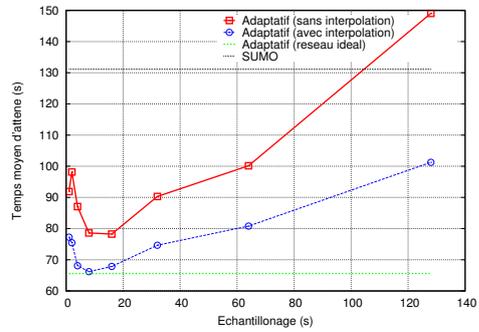
L'aspect principal qu'il nous faudrait explorer est la collaboration entre intersections. Dans la version présentée de notre algorithme, les intersections collaborent uniquement en vue de calculer la place disponible sur une route. Toutefois, nous

---

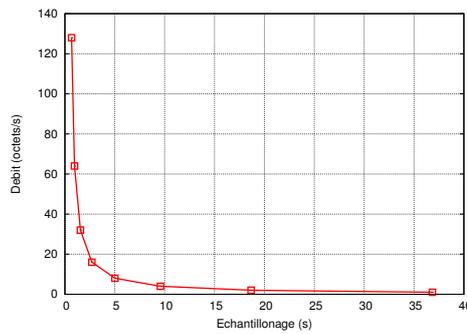
2. <http://basalte.sfaye.com/>



(a) Temps d'attente vs. fréquence d'échantillonnage ( $\lambda = 1$ ).

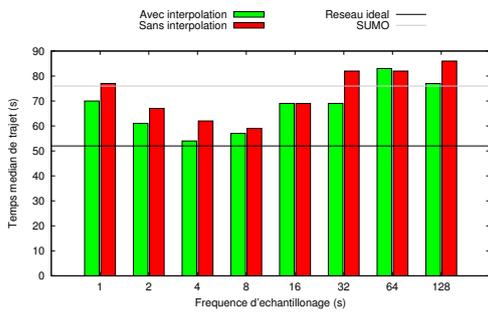


(b) Temps d'attente vs. fréquence d'échantillonnage ( $\lambda = 2$ ).

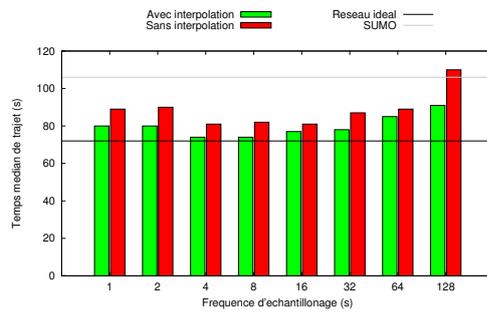


(c) Fréquence d'échantillonnage vs. débit moyen par nœud.

FIGURE 6.6 – Résultats en fonction de la fréquence d'échantillonnage.



(a) Temps médian de trajet ( $\lambda = 1$ ).



(b) Temps médian de trajet ( $\lambda = 2$ ).

FIGURE 6.7 – Performances du système de transport intelligent.

pourrions bénéficier de cette collaboration pour synchroniser les feux entre eux, afin de favoriser l'absorption d'un embouteillage. Nous pouvons tout à fait imaginer un mécanisme asynchrone, similaire à celui qui est proposé sur TAPIOCA et basé sur un score, comme représenté sur la figure 6.8. Un tel mécanisme pourrait, par exemple, hiérarchiser les intersections en fonction de leur charge, de manière à créer des chemins de feux verts menant à la bordure de la zone embouteillée.

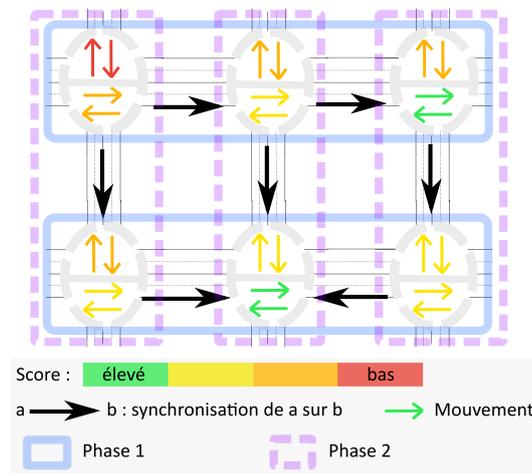


FIGURE 6.8 – Exemple d'une synchronisation des feux verts.

De plus, outre la collaboration avec une intersection voisine, nous pourrions bénéficier de la communication avec des intersections situées à plus grande distance. Dans le cas d'une situation de trafic normale, nous avons le *sentiment* que ceci ne changera rien. Dans une situation où le trafic est embouteillé, nous pouvons imaginer communiquer avec les intersections situées en bordure de la zone congestionnée, de manière à comprendre l'étendue de l'embouteillage et cerner de meilleures stratégies. Nous pouvons également imaginer intégrer des probabilités sur le nombre de véhicules pouvant arriver à plusieurs minutes d'intervalles, afin de préparer leur arrivée.

Ensuite, un point de passage obligatoire serait de présenter des résultats issus de simulations plus étendues que le réseau de trois intersections présenté ici. Ce premier scénario nous montre le potentiel de BASALTE, qui a été conçu pour être généralisable à tout type d'intersection. Toutefois, nous gagnerions à profiter de cette généralisation afin d'étudier des cas de scénarios plus étendus, avec ou sans mécanisme de synchronisation entre les intersections.

Enfin, il nous paraît intéressant de coupler l'algorithme présenté à TAPIOCA, afin d'alterner les deux stratégies en fonction du niveau d'encombrement. Nous pourrions notamment bénéficier de meilleures performances. Une idée similaire a déjà été développée par [CS04a]. Ils utilisent une méthode de gestion pour les intersections sous-saturées, par l'intermédiaire de TRANSYT-7F (sec. 2. 3. 4. 1), ainsi qu'une méthode de gestion pour les intersections sursaturées. Leur couplage est testé sur un réseau de 12 intersections sursaturées, entourées de 13 intersections sous-saturées, et fourni de meilleurs résultats que TRANSYT-7F seul.

## 6. 7 Conclusion

Nous avons présenté dans ce chapitre un algorithme de gestion des feux de circulation traitant le cas des intersections saturées.

Utiliser un tel réseau permet d'augmenter le nombre de points de mesure du trafic et de mettre en place une architecture distribuée plus réactive et plus adaptative que les systèmes traditionnels. En effet, en raisonnant mouvement par mouvement, nous contrôlons finement où les usagers peuvent aller, plutôt que de raisonner par phase et risquer de conduire certains véhicules à une voie saturée.

De plus, afin de gérer les situations d'embouteillage, nous utilisons des métriques d'espace plutôt que de temps. Ceci nous permet de nous focaliser sur le nombre de véhicules qui peuvent se déplacer d'une intersection à une autre, et non sur une stratégie reposant sur des métriques temporelles (p. ex. temps d'attente depuis le dernier feu vert).

Cependant, malgré la faible quantité de trafic générée par les mesures dans un tel environnement, l'impact des pertes de paquets est sensible au niveau de l'application et il existe une fréquence d'acquisition optimale. Les défauts de données peuvent être compensés efficacement par un mécanisme d'interpolation simple. Le déploiement d'un tel réseau au niveau d'une infrastructure routière complète créera, en outre, une diversité de chemins dont il sera possible de tirer profit pour compenser les pertes et pour tirer parti d'informations provenant d'intersection éloignées.

## CHAPITRE 7

---

### Conclusion générale et perspectives

---

Cette thèse apporte plusieurs contributions au domaine des systèmes de transport intelligents. Elle propose des solutions distribuées de gestion des feux de circulation basées sur des capteurs. Les contributions portent sur les solutions algorithmiques réparties de gestion des feux de circulation et sur les solutions de déploiement des capteurs sur la chaussée.

Comme en témoigne le chapitre 2 d'état de l'art, la littérature traitant des systèmes de transport traditionnels est abondante. Dans ce chapitre, nous identifions les modèles, équipements et réseaux de communication habituellement en place. Il est ainsi fréquent que les équipements de la circulation soient reliés à un centre de contrôle. Ce dernier agrège généralement les mesures réalisées sur le terrain, définit des stratégies de gestion des feux, prend des décisions quant à la planification urbaine et détecte les accidents ou infractions. En revanche, le recours à un centre de contrôle augmente le temps de latence, s'avère généralement coûteux et restreint l'évolution du réseau. Les systèmes distribués n'utilisent pas de centre de contrôle et permettent de décider localement de la meilleure stratégie à adopter. Cela, en fonction d'une vision locale qu'ils peuvent éventuellement partager de proche en proche, afin de couvrir une large zone urbaine.

Le chapitre 3 aborde la question du déploiement en étudiant l'installation de réseaux de capteurs sans fil sur plus d'une cinquantaine de villes. Nous prouvons, en particulier, que le graphe résultant de ce déploiement est très partitionné, mais qu'il peut être propice au développement d'architectures étendues à quelques intersections. Nous montrons également que la distribution des degrés a tendance à suivre une loi gamma, et que les composantes connexes des graphes étudiés peuvent facilement être reliées avec des nœuds supplémentaires.

Le chapitre 4 se recentre sur des zones urbaines plus restreintes que celles qui ont été étudiées, en présentant un algorithme de contrôle des feux de circulation agissant sur plusieurs intersections, nommé TAPIOCA. Cet algorithme a la particularité d'être implanté sur chaque intersection, qui peuvent prendre une décision locale et coopérer avec leurs voisines en vue de créer des chemins de feux verts successifs. Toutefois, si, à ce stade, les performances sur le temps moyen d'attente nous semblent

prometteuses, l'évaluation réalisée ne considère pas les faiblesses d'une communication sans fil. Le chapitre 5 répond à ce problème en considérant que le réseau de capteurs doit être réaliste, en se heurtant à des délais et à des interférences. Nous proposons ainsi un mécanisme d'interpolation et prouvons par simulation que TAPIOCA est prêt à être déployé sur un scénario réel.

Le chapitre 6 aborde le cas particulier des embouteillages qui nécessitent des métriques d'espace plutôt que des métriques temporelles. Nous définissons un nouvel algorithme, baptisé BASALTE, et utilisons pleinement chaque capteur comme un élément décisif de la détermination des phases de feux.

Cette thèse a été construite avec une vision pragmatique, non pas en cherchant à proposer les meilleures performances algorithmiques, mais en cherchant à développer une certaine manière de penser, qui nous a paru pertinente et applicable dans le cas de la gestion de la circulation urbaine.

Les modèles et algorithmes présentés nous permettent de définir ce qu'il est possible ou non de faire en alliant un système distribué à un système de transport intelligent. Si de nombreuses pistes restent à explorer, les résultats obtenus nous permettent de croire que l'utilisation d'une logique distribuée est adaptée à la gestion de la circulation urbaine. Non seulement, cette dernière est possible à l'échelle de plusieurs intersections, mais elle semble, en plus, réaliste et prometteuse. Les méthodes de déploiement suggérées offrent une capillarité fine, permettant l'installation de nœuds à des endroits parfois inaccessibles par d'autres technologies. Les algorithmes développés, bien que perfectibles, nous permettent de réaliser que notre intuition de départ était correcte. En faisant collaborer plusieurs nœuds entre eux, ils offrent une certaine autonomie aux systèmes de transport intelligents, en plus de pouvoir s'appliquer sur des zones inaccessibles aux centres de contrôle traditionnels. En réponse à la problématique posée en introduction, nous pouvons conclure que l'utilisation de tels réseaux – et par extension, de systèmes distribués – représente un réel atout pour la gestion de la circulation routière. En outre, les avantages de ces réseaux en termes de déploiement, de réactivité et de coûts sont nombreux et en font une solution viable et d'avenir.

Cette thèse ouvre plusieurs perspectives de travaux futurs.

Tout d'abord, l'ensemble de nos résultats ont été établis par simulation, et non de façon expérimentale ou sur le terrain. Un déploiement réel pourrait introduire de nouvelles problématiques que nous n'avons pas abordées. De plus, certaines métriques restent à étudier. En effet, les deux simulateurs utilisés, SUMO pour le côté transport et OMNeT++ pour le côté réseau, permettent d'analyser une large panoplie d'aspects (par exemple, effets sur la pollution).

En second lieu, l'étude des graphes que nous avons réalisé peut être étendue. Tout d'abord, le modèle d'affaiblissement de propagation utilisé est simple et ne tient pas compte des bâtiments ou des interférences avec les réseaux urbains existants. Il serait intéressant de se baser sur des modèles plus complexes, comme le font par exemple les modèles de Young ou de Hate [Sey05] pour étudier le comportement de communications cellulaires dans des zones urbaines. Une autre possibilité serait de considérer que deux nœuds séparés par une large structure ne peuvent pas commu-

niquer entre eux. De plus, nous nous limitons au cas des réseaux de capteurs sans fil et d'une interface 802.15.4, alors qu'il est tout à fait possible d'imaginer des réseaux avec une portée de communication plus large. Enfin, l'ajout d'un réseau véhiculaire en marge d'un réseau distribué de capteurs nous permettrait d'étudier des graphes dynamiques, où les nœuds fixes combleraient les lacunes du graphe formé par les nœuds mobiles [NF13], et inversement. Nous pourrions en effet bénéficier d'une architecture coopérative, qui couvrirait des zones plus larges et permettrait d'aborder de nouvelles solutions de gestion du trafic. Notons que cette coopération n'est pas nécessairement limitée aux réseaux véhiculaires : elle pourrait également inclure des réseaux cellulaires, urbains et s'inscrire dans l'Internet des objets.

En marge de ceci, nous avons considéré qu'une intersection ne devait collaborer qu'avec ses voisines proches. Toutefois, il serait intéressant d'étendre cette collaboration à des intersections plus distantes. Même si nous avons le *sentiment* que ceci n'aurait pas un grand impact sur des situations de trafic stables, il serait intéressant d'étudier le cas des embouteillages. En effet, si une intersection a la possibilité de communiquer avec une intersection située en bordure de la zone embouteillée, cela pourrait permettre de mieux dérouter les véhicules. D'une part, un capteur peut constater des asymétries dans la charge de trafic, ce qui permettrait d'identifier la frontière de la zone embouteillée. Il serait alors possible d'orienter les flux de véhicules vers les intersections situées hors de cette frontière pour déconstruire un embouteillage. Il est également possible d'identifier les intersections qui génèrent une charge élevée de trafic, et de tenter de limiter l'expansion du phénomène depuis sa source.

Par ailleurs, nous avons fait le choix de présenter de façon indépendante deux visions de la gestion routière, sur des scénarios stables et des scénarios embouteillés. Toutefois, il serait bénéfique d'imaginer un système de régulation qui alterne ces deux stratégies en fonction d'un contexte existant ou préexistant. Des exemples existent dans la littérature [CS04a], mais ils sont généralement simples et n'abordent pas, par exemple, le cas où deux ensembles d'intersections voisines auraient des stratégies différentes. Un système distribué pourrait permettre à chaque intersection de choisir sa stratégie, en fonction de son niveau d'encombrement. De plus, une intersection pourrait encourager ses voisines à changer de stratégie.

Un autre aspect à développer est l'étude de la tolérance aux pannes. En effet, si les mécanismes d'interpolation proposés dans cette thèse peuvent tolérer quelques pertes de paquets et des délais, ils sont inefficaces lorsque les capteurs sont en panne. Dans un cas général, il n'y a pas de méthode pour prédire de manière permanente les données manquantes. En effectuant une analyse de corrélation sur les différentes valeurs des taux d'arrivées et de départs, il pourrait être possible de remplacer des valeurs perdues par une combinaison linéaire des autres. Cependant, la pertinence de cette méthode n'est pas garantie dans tous les cas. De même, les données venant d'intersections adjacentes pourraient être considérées.

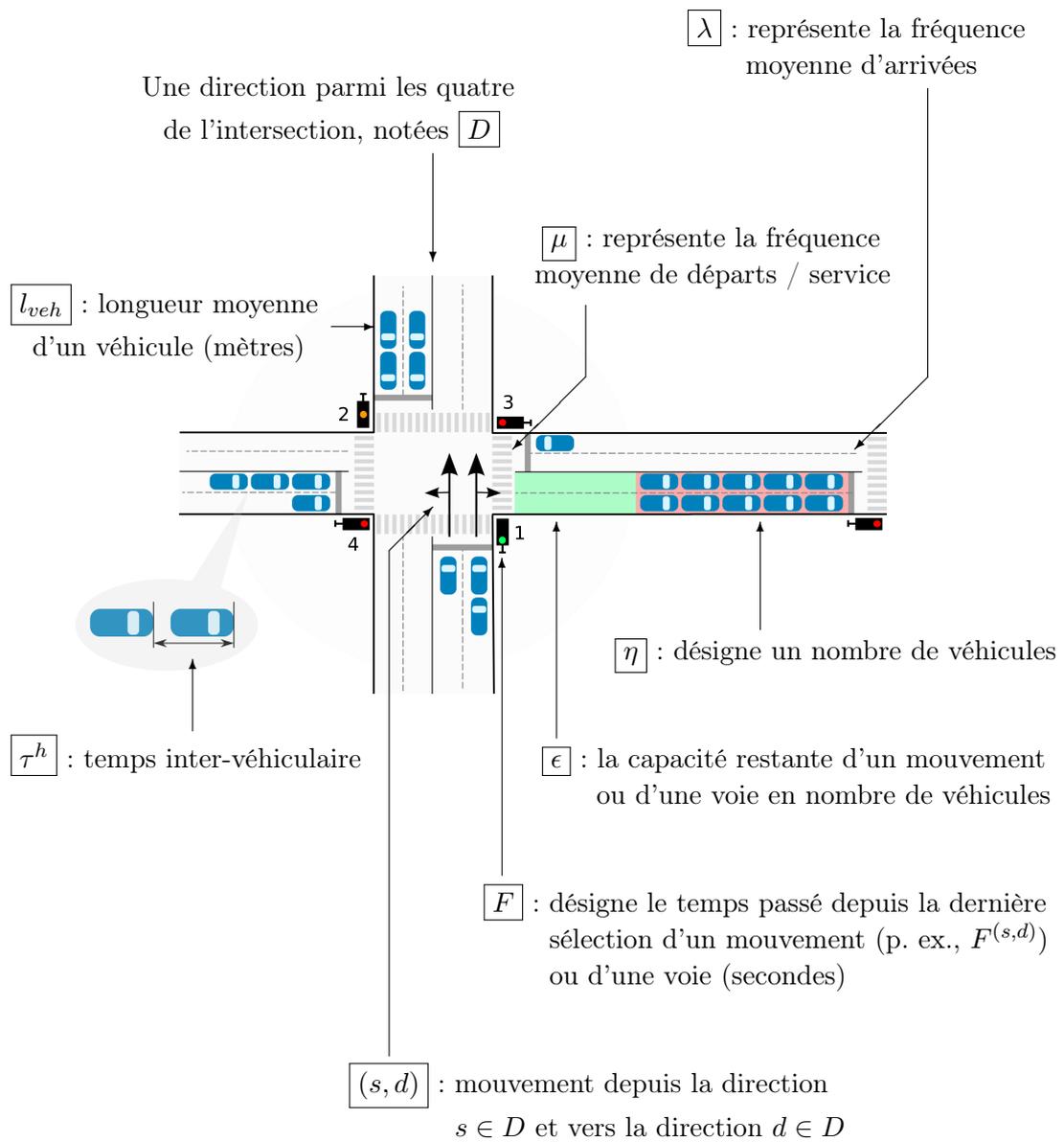
Cette thèse présente essentiellement des déploiements qui s'effectuent dans des zones urbaines développées. Toutefois, le coût apparent des capteurs utilisés permettrait, *a priori*, d'équiper des villes pauvres ou démunies de tout système de transport intelligent. De plus, le caractère indépendant de ces systèmes distribués permettrait

de se passer d'un centre de contrôle, en installant des capteurs directement sur le terrain. Outre les problématiques liées aux réseaux de capteurs, il serait également intéressant d'étudier la pertinence et le coût réel suggérés par un tel déploiement, en termes d'installation, de prix et de maintenance.

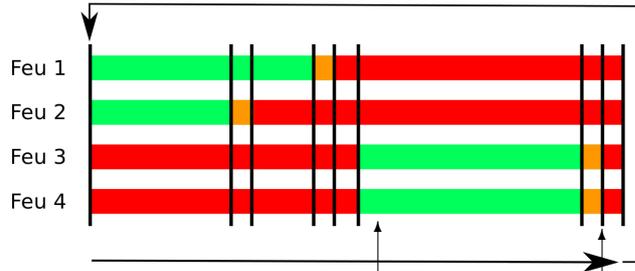
Enfin, il serait intéressant de caractériser la place que les architectures définies peuvent laisser à l'intervention humaine. En effet, l'un des intérêts d'un système distribué est d'amener une certaine autonomie à un système de transport. Cette autonomie serait-elle un frein pour les postes de régulation ? La compréhension du système, de même que l'accès aux équipements, seraient-ils fondamentalement différents des systèmes de transport actuels ? Ces questions peuvent être abordées de plusieurs points de vue : technique et réseau en premier lieu, mais également à l'échelle des sciences humaines et du comportement humain.

## ANNEXE A

## Principales notations



Plan de feu cyclique.  $\tau^c$ ,  $\tau_{max}^c$  sont respectivement le temps effectif d'un cycle et le temps maximal de cycle (secondes).



L'une des phases du plan de feu.  $\tau_i^g$ ,  $\tau_i^{g^{max}}$ ,  $\tau_i^{g^{min}}$ ,  $\tau_i^{g^{add}}$  sont respectivement le temps de feu vert effectif, maximum, minimum et additionnel de la phase  $i$

$\tau^{LT}$  : temps total perdu sur une phase (secondes), ici simplifié à la somme du temps de feu orange  $\tau^{orange}$ , du temps de sécurité  $\tau^{secure}$ , et du temps de démarrage  $\tau^s$

La notation  $(t)$  peut être utilisée pour identifier la valeur de certaines variables au temps  $(t)$ .  $\eta^{(s,d)}(t)$  est, par exemple, le nombre de véhicules présents sur le mouvement  $(s, d)$  au temps  $t$ . Lorsque  $(t)$  n'est pas précisé, la valeur la plus récente est retournée.

# ANNEXE B

---

## Liste des sigles

---

- **AWT** : temps moyen d'attente (*average waiting time*).
- **BASALTE** : *a distriButed and Adaptive Signalization ALgorithm for saTura-ted intErsections*.
- **CDF** : courbe de fréquence cumulée (*cumulative distribution fonction*).
- **CIGT** : centres d'ingénierie et de gestion du trafic.
- **Nœud DN** : nœud de destination (*destination node*).
- **Nœud SN** : nœud source (*source node*).
- **Nœud AN** : nœud additionnels / de sortie (*additional node*).
- **PC** : poste de contrôle.
- **RCSF – WSN** : réseau de capteurs sans fil – *wireless sensor network*.
- **RSU** : *road side unit*.
- **STI – ITS** : système de transport intelligent – *intelligent transportation sys-tem*.
- **TAPIOCA** : *a distribuTed and AdaPtive IntersectiOns Control Algorithm*.
- **TLC** : *trafic logic control*.
- **UVP** : unité de véhicule particulier.
- **V/C** : rapport espace occupé sur capacité.



## ANNEXE C

---

### Liste des principaux outils et simulateurs

---

- **gnuplot**<sup>1</sup>, logiciel en ligne de commande de production de graphiques.
- **NetworkX**<sup>2</sup> [HSS08], bibliothèque Python pour l'étude des graphes et des réseaux.
- **OMNeT++**<sup>3</sup> [VH08], simulateur réseau.
- **OpenStreetMap**<sup>4</sup> [HW08], carte du monde sous licence libre.
- **R**<sup>5</sup> [Tea+12], logiciel de traitement des données et d'analyse statistiques.
- **SUMO**<sup>6</sup> [BBEK11 ; KEBB12], simulateur de trafic routier.
- **Veins**<sup>7</sup> [SGD11], framework permettant de lier SUMO à OMNeT++.

---

1. <http://www.gnuplot.info/>
2. <https://networkx.github.io/>
3. <http://www.omnetpp.org/>
4. <http://www.openstreetmap.org/>
5. <http://www.r-project.org/>
6. <http://sumo-sim.org/>
7. <http://veins.car2x.org/>



## ANNEXE D

---

### Liste des publications

---

#### Journaux internationaux

- [FM13] Sébastien FAYE et Jean Frédéric MYOUPPO. “Secure and Energy-efficient Geocast Protocols for Wireless Sensor Networks based on a Hierarchical Clustered Structure”. Dans : *International Journal of Network Security* 15.3 (mai 2013), p. 151–160.
- [FMB13] Sébastien FAYE, Jean Frédéric MYOUPPO et Alain Bertrand BOMGNI. “Heterogeneous Clustering for Secure, Energy-efficient and Fault tolerant Permutation Routing in Wireless Sensor Networks”. Dans : *International Journal of Advanced Computer Science* 3.5 (2013).
- [FM12] Sébastien FAYE et Jean Frédéric MYOUPPO. “Deployment and Management of Sparse Sensor-Actuator Network in a Virtual Architecture”. Dans : *International Journal of Advanced Computer Science* 2.12 (2012).
- [FM11a] Sébastien FAYE et Jean Frédéric MYOUPPO. “An ultra hierarchical clustering-based secure aggregation protocol for wireless sensor networks”. Dans : *AISS : Advances in Information Sciences and Service Sciences* 3.9 (2011), p. 309–319.

#### Conférences internationales avec comité de lecture

- [FCD13a] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “Influence of Radio Communications on Multiple Intersection Control by a Wireless Sensor Network”. Dans : *The 13th IEEE International Conference on ITS Telecommunications (ITST 2013)*. Tampere, Finland, nov. 2013.
- [FCD12a] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “A Distributed Algorithm for Adaptive Traffic Lights Control”. Dans : *15th IEEE Intelligent Transportation Systems Conference (ITSC 2012)*. Anchorage, USA, sept. 2012.

- [FCD12b] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “A Distributed Algorithm for Multiple Intersections Adaptive Traffic Lights Control using a Wireless Sensor Networks”. Dans : *First Urban Workshop, 8th International Conference on emerging Networking Experiments and Technologies (CoNEXT)*. Nice, France, déc. 2012, p. 13–18.
- [FM11b] Sébastien FAYE et Jean Frédéric MYOUP. “Deployment of Sparse Sensor-Actuator Network in a Virtual Architecture”. Dans : *The 2011 International Conference on Wireless Networks (ICWN 2011)*. T. 11. Las Vegas, USA, juil. 2011, p. 18–21.

### Soumis pour publication

- [FC14a] Sébastien FAYE et Claude CHAUDET. “Characterizing the Topology of a Urban Wireless Sensor Network for Road Traffic Management”. Manuscript submitted for publication in : *IEEE Transactions on Vehicular Technology*. 2014.

### Publications nationales

- [FCD13b] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “Un algorithme distribué de contrôle des feux de circulation sur plusieurs intersections par un réseau de capteurs sans fil”. Dans : *Revue des Nouvelles Technologies de l’Information Ingénierie des protocoles et Nouvelles technologies de la répartition : sélection d’articles de CFIP/NOTERE 2012, RNTI-SM-2 (2013)*, p. 121–140.
- [FCD12d] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “Un algorithme distribué de contrôle des feux de circulation sur plusieurs intersections par un réseau de capteurs sans fil”. Dans : *NOTERE-CFIP 2012*. Anglet, France, oct. 2012.

### Rapports techniques

- [FC14b] Sébastien FAYE et Claude CHAUDET. *Characterizing the Topology of a Urban Wireless Sensor Network for Road Traffic Management*. Mai 2014.
- [FCD14a] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. *Multiple intersections adaptive traffic lights control using a wireless sensor networks*. Août 2014.
- [Fay12] Sébastien FAYE. *Contrôle du trafic routier urbain par un réseau fixe de capteurs sans fil*. Mar. 2012.
- [Fay11] Sébastien FAYE. *Sécurité dans des réseaux de capteurs sans fil : mécanismes de prévention*. Juin 2011.

**Posters et présentations**

- [FCD14b] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “WSN for Road Traffic Management”. Dans : *LINCS Workshop*. Paris, France, 2014.
- [FCD13c] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “WSN for Traffic Light Management”. Dans : *LINCS Workshop*. Paris, France, 2013.
- [FCD12c] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “Contrôle des feux de circulation par un réseau de capteurs sans-fil”. Dans : *Journées Spécifiques Pôle ResCom - GDR ASR - CNRS, Réseaux Véhiculaires (REVE)*. Paris, France, avr. 2012.
- [FM11c] Sébastien FAYE et Jean Frédéric MYOUPPO. “Sécurité dans des réseaux de capteurs sans fil : mécanismes de prévention”. Dans : *Journée des Jeunes Chercheurs du MIS (JJCMIS)*. Amiens, France, mai 2011.

**Ressources en ligne**

- [Fay14a] Sébastien FAYE. *Characterizing the Topology of a Urban Wireless Sensor Network for Road Traffic Management*. LGPL licence. Mai 2014. URL : <http://g.sfaye.com/>.
- [Fay14b] Sébastien FAYE. *Traffic Lights Control with BASALTE, a distriButed and Adaptive Signalization ALgorithm for saTurated intErsections*. Fév. 2014. URL : <http://basalte.sfaye.com/>.
- [Fay13] Sébastien FAYE. *Traffic Lights Control with TAPIOCA, a distribuTed and AdaPtive IntersectiOns Control Algorithm*. Fév. 2013. URL : <http://tapioca.sfaye.com/>.



---

## Bibliographie

---

- [DHMP14] Tony DUCROCQ, Michael HAUSPIE, Nathalie MITTON et Sara PIZZI. “On the Impact of Network Topology on Wireless Sensor Networks Performances Illustration with Geographic Routing”. Dans : *International Workshop on the Performance Analysis and Enhancement of Wireless Networks (PAEWN)*. Victoria, Canada, mai 2014.
- [Fay14a] Sébastien FAYE. *Characterizing the Topology of a Urban Wireless Sensor Network for Road Traffic Management*. LGPL licence. Mai 2014. URL : <http://g.sfaye.com/>.
- [Fay14b] Sébastien FAYE. *Traffic Lights Control with BASALTE, a distriButed and Adaptive Signalization ALgorithm for saTurated intErsections*. Fév. 2014. URL : <http://basalte.sfaye.com/>.
- [FC14a] Sébastien FAYE et Claude CHAUDET. “Characterizing the Topology of a Urban Wireless Sensor Network for Road Traffic Management”. Manuscript submitted for publication in : *IEEE Transactions on Vehicular Technology*. 2014.
- [FC14b] Sébastien FAYE et Claude CHAUDET. *Characterizing the Topology of a Urban Wireless Sensor Network for Road Traffic Management*. Mai 2014.
- [FCD14a] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. *Multiple intersections adaptive traffic lights control using a wireless sensor networks*. Août 2014.
- [FCD14b] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “WSN for Road Traffic Management”. Dans : *LINCS Workshop*. Paris, France, 2014.
- [PZCG14] Charith PERERA, Arkady ZASLAVSKY, Peter CHRISTEN et Dimitrios GEORGAKOPOULOS. “Sensing as a service model for smart cities supported by internet of things”. Dans : *Transactions on Emerging Telecommunications Technologies* 25.1 (2014).
- [Fay13] Sébastien FAYE. *Traffic Lights Control with TAPIOCA, a distributed and AdaPtive IntersectiOns Control Algorithm*. Fév. 2013. URL : <http://tapioca.sfaye.com/>.

- [FCD13a] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “Influence of Radio Communications on Multiple Intersection Control by a Wireless Sensor Network”. Dans : *The 13th IEEE International Conference on ITS Telecommunications (ITST 2013)*. Tampere, Finland, nov. 2013.
- [FCD13b] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “Un algorithme distribué de contrôle des feux de circulation sur plusieurs intersections par un réseau de capteurs sans fil”. Dans : *Revue des Nouvelles Technologies de l’Information* Ingénierie des protocoles et Nouvelles technologies de la répartition : sélection d’articles de CFIP/-NOTERE 2012, RNTI-SM-2 (2013), p. 121–140.
- [FCD13c] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “WSN for Traffic Light Management”. Dans : *LINCS Workshop*. Paris, France, 2013.
- [FM13] Sébastien FAYE et Jean Frédéric MYOUPPO. “Secure and Energy-efficient Geocast Protocols for Wireless Sensor Networks based on a Hierarchical Clustered Structure”. Dans : *International Journal of Network Security* 15.3 (mai 2013), p. 151–160.
- [FMB13] Sébastien FAYE, Jean Frédéric MYOUPPO et Alain Bertrand BOMGNI. “Heterogeneous Clustering for Secure, Energy-efficient and Fault tolerant Permutation Routing in Wireless Sensor Networks”. Dans : *International Journal of Advanced Computer Science* 3.5 (2013).
- [Hor13a] Berthold K.P. HORN. “Suppressing Traffic Flow Instabilities”. Dans : *IEEE Intelligent Transportation Systems Conference (ITSC 2013)*. Den Haag, Netherlands, oct. 2013.
- [Hor13b] B.K.P. HORN. ““Suppressing Traffic Flow Instabilities””. Dans : *IEEE Intelligent Transportation Systems Conference (ITSC 2013)*. Den Haag, Netherlands, oct. 2013.
- [JMMSAS13] Stewart JACKSON, Luis F MIRANDA-MORENO, Paul ST-AUBIN et Nicolas SAUNIER. “Flexible, mobile video camera system and open source video analysis software for road safety and behavioral analysis”. Dans : *Transportation Research Record : Journal of the Transportation Research Board* 2365.1 (2013), p. 90–98.
- [Lit13] Todd LITMAN. “Factors to Consider When Estimating Congestion Costs and Evaluating Potential Congestion Reduction Strategies”. Dans : *Victoria, Canada : Victoria Transport Policy Institute* (2013).
- [Liu+13] Yunhao LIU et al. “Does wireless sensor network scale? A measurement study on GreenOrbs”. Dans : *Parallel and Distributed Systems, IEEE Transactions on* 24.10 (2013).
- [NF13] Diala NABOULSI et Marco FIORE. “On the instantaneous topology of a large-scale urban vehicular network : the cologne case”. Dans : *Proceedings of the fourteenth ACM international symposium on Mobile ad hoc networking and computing*. 2013.

- [SL13] Gérard SCEMAMA et Patrick LAFFONT. “CLAIRE-SITI Système pour la supervision intelligente, intermodale, intégrée des réseaux de transport de surface”. Dans : *TEC Transport environnement circulation* 219 (jan. 2013), 11 p. URL : <http://hal.archives-ouvertes.fr/hal-00875742>.
- [WQY13] Aoxiang WU, Liqun QI et Xiaoguang YANG. “Mechanism Analysis and Optimization of Signalized Intersection Coordinated Control under Oversaturated Status”. Dans : (2013).
- [Yan+13] LI YAN et al. “Review of traffic signal control methods under oversaturated conditions”. Dans : *Journal of Traffic and Transportation Engineering* 13.4 (2013).
- [Cj+12] Dong CHUN-JIAO et al. “Temporal-spatial characteristic of urban expressway under jam flow condition”. Dans : *Journal of Traffic and Transportation Engineering* 12.3 (2012), p. 73–79.
- [CPSS12] M. COLLOTTA, G. PAU, VM SALERNO et G. SCATÁ. “Wireless Sensor Networks to Improve Road Monitoring”. Dans : *InTechOpen* (2012).
- [Dim12] Catalin DIMON. “Contributions à la modélisation et la commande des réseaux de trafic routier”. Thèse de doct. Ecole Centrale de Lille, 2012.
- [Fay12] Sébastien FAYE. *Contrôle du trafic routier urbain par un réseau fixe de capteurs sans fil*. Mar. 2012.
- [FCD12a] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “A Distributed Algorithm for Adaptive Traffic Lights Control”. Dans : *15th IEEE Intelligent Transportation Systems Conference (ITSC 2012)*. Anchorage, USA, sept. 2012.
- [FCD12b] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “A Distributed Algorithm for Multiple Intersections Adaptive Traffic Lights Control using a Wireless Sensor Networks”. Dans : *First Urban Workshop, 8th International Conference on emerging Networking Experiments and Technologies (CoNEXT)*. Nice, France, déc. 2012, p. 13–18.
- [FCD12c] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “Contrôle des feux de circulation par un réseau de capteurs sans-fil”. Dans : *Journées Spécifiques Pôle ResCom - GDR ASR - CNRS, Réseaux Véhiculaires (REVE)*. Paris, France, avr. 2012.
- [FCD12d] Sébastien FAYE, Claude CHAUDET et Isabelle DEMEURE. “Un algorithme distribué de contrôle des feux de circulation sur plusieurs intersections par un réseau de capteurs sans fil”. Dans : *NOTERE-CFIP 2012*. Anglet, France, oct. 2012.
- [FM12] Sébastien FAYE et Jean Frédéric MYOUPPO. “Deployment and Management of Sparse Sensor-Actuator Network in a Virtual Architecture”. Dans : *International Journal of Advanced Computer Science* 2.12 (2012).

- [HHD12] Qing HE, K Larry HEAD et Jun DING. “PAMSCOD : Platoon-based arterial multi-modal signal control with online data”. Dans : *Transportation Research Part C : Emerging Technologies* 20.1 (2012), p. 164–184.
- [KESNM12] Lutful KARIM, Tarek EL SALT, Nidal NASSER et Qusay H. MAHMOUD. “The significant impact of a set of topologies on wireless sensor networks”. Dans : *EURASIP Journal on Wireless Communications and Networking* (2012).
- [KEBB12] Daniel KRAJZEWICZ, Jakob ERDMANN, Michael BEHRISCH et Laura BIEKER. “Recent Development and Applications of SUMO - Simulation of Urban MObility”. Dans : *International Journal On Advances in Systems and Measurements*. International Journal On Advances in Systems and Measurements 5.3&4 (2012). URL : <http://elib.dlr.de/80483/>.
- [Mao+12] Xufei MAO et al. “CitySee : Urban CO2 Monitoring with Sensors”. Dans : *Infocom*. 2012.
- [Tea+12] R Core TEAM et al. “R : A language and environment for statistical computing”. Dans : (2012).
- [AAG11] Montasir M ABBAS, Zain M ADAM et Douglas GETTMAN. “Development and Evaluation of Optimal Arterial Control Strategies for Oversaturated Conditions”. Dans : *Transportation Research Record : Journal of the Transportation Research Board* 2259.1 (2011), p. 242–252.
- [ANR11] F.A. AL-NASSER et H. ROWAIHY. “Simulation of dynamic traffic control system based on wireless sensor network”. Dans : *2011 IEEE Symposium on Computers Informatics (ISCI)*. Mar. 2011, p. 40–45.
- [BBEK11] M. BEHRISCH, L. BIEKER, J. ERDMANN et D. KRAJZEWICZ. “SUMO - Simulation of Urban MObility : An Overview”. Dans : *The Third International Conference on Advances in System Simulation (SIMUL 2011)*. Barcelona, Spain, oct. 2011, p. 63–68.
- [Fay11] Sébastien FAYE. *Sécurité dans des réseaux de capteurs sans fil : mécanismes de prévention*. Juin 2011.
- [FM11a] Sébastien FAYE et Jean Frédéric MYOUPPO. “An ultra hierarchical clustering-based secure aggregation protocol for wireless sensor networks”. Dans : *AISS : Advances in Information Sciences and Service Sciences* 3.9 (2011), p. 309–319.
- [FM11b] Sébastien FAYE et Jean Frédéric MYOUPPO. “Deployment of Sparse Sensor-Actuator Network in a Virtual Architecture”. Dans : *The 2011 International Conference on Wireless Networks (ICWN 2011)*. T. 11. Las Vegas, USA, juil. 2011, p. 18–21.
- [FM11c] Sébastien FAYE et Jean Frédéric MYOUPPO. “Sécurité dans des réseaux de capteurs sans fil : mécanismes de prévention”. Dans : *Journée des Jeunes Chercheurs du MIS (JJCMIS)*. Amiens, France, mai 2011.

- [HMB11] Ali HAJBABAIE, Juan C MEDINA et Rahim F. BENEKOHAL. *Traffic Signal Coordination and Queue Management in Oversaturated Intersection*. By. Rap. tech. 2011.
- [HTYJ11] Peifeng HU, Zongzhong TIAN, Zhenzhou YUAN et Shunping JIA. “Variable-Bandwidth Progression Optimization in Traffic Operation”. Dans : *Journal of Transportation Systems Engineering and Information Technology* 11.1 (2011), p. 61–72.
- [IBM11] IBM. *Sondage mondial IBM sur la pénibilité du navettage : congestion routière en baisse, pénibilité nettement en hausse*. Juin 2011. URL : <http://www.ibm.com/news/ca/fr/2011/09/08/u251344d61892u91.html>.
- [IEE11] IEEE STANDARD FOR LOCAL AND METROPOLITAN AREA NETWORKS. *Part 15.4 : Low-Rate Wireless Personal Area Networks (LR-WPANs)*. 2011.
- [LL11] Binyang LIU et Weifang LIU. “Evaluation of traffic control methods at traffic circles”. Dans : *Control and Decision Conference (CCDC), 2011 Chinese*. Mai 2011, p. 3371–3377.
- [LC11] Yue LIU et Gang-Len CHANG. “An arterial signal optimization model for intersections experiencing queue spillback and lane blockage”. Dans : *Transportation research part C : emerging technologies* 19.1 (2011), p. 130–144.
- [Puc+11] Daniele PUCCINELLI et al. “The Impact of Network Topology on Collection Performance”. Dans : *8th European conference on Wireless sensor networks*. 2011.
- [SGD11] Christoph SOMMER, Reinhard GERMAN et Falko DRESSLER. “Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis”. Dans : *IEEE Transactions on Mobile Computing* 10.1 (2011), p. 3–15. DOI : [10.1109/TMC.2010.133](https://doi.org/10.1109/TMC.2010.133).
- [Whi11] Gerald Beresford WHITHAM. *Linear and nonlinear waves*. T. 42. John Wiley & Sons, 2011.
- [ZY11] G. ZHANG et Wang Y. “Optimizing Minimum and Maximum Green Time Settings for Traffic Actuated Control at Isolated Intersections”. Dans : *Intelligent Transportation Systems, IEEE Transactions on* 99 (2011), p. 1–10.
- [ZCW11] B. ZHOU, J. CAO et H. WU. “Adaptive Traffic Light Control of Multiple Intersections in WSN-Based ITS”. Dans : *73rd IEEE Vehicular Technology Conference (VTC Spring)*. 2011.
- [CWLL10] J. CAO, H. WU, X. LIU et Y. LAI. “iSensNet : an infrastructure for research and development in wireless sensor networks”. Dans : *Frontiers of Computer Science in China* 4.3 (2010), p. 339–353.
- [Cha10] CHANDRA, REGGIE AND GREGORY, CHRIS. “InSync Adaptive Traffic Signal Technology : Real-Time Artificial Intelligence Delivering Real-World Results”. Dans : (juil. 2010).

- [CBX10] Jinil CHANG, Bryan BERTOLI et Wuping XIN. “New signal control optimization policy for oversaturated arterial systems”. Dans : *Transportation Research Board 89th Annual Meeting*. 10-3760. 2010.
- [FF10] Nicholas J FUDALA et Michael D FONTAINE. *Work Zone Variable Speed Limit Systems : Effectiveness and System Design Issues*. Rap. tech. 2010.
- [Hak10] Mordechai HAKLAY. “How good is volunteered geographical information? A comparative study of OpenStreetMap and Ordnance Survey datasets”. Dans : *Environment and planning. B, Planning & design* 37.4 (2010).
- [HZY10] Duan HOULI, Li ZHIHENG et Zhang YI. “Multiobjective reinforcement learning for traffic signal control using vehicular ad hoc network”. Dans : *EURASIP J. Adv. Signal Process* 2010 (mar. 2010), 7 :1–7 :7.
- [KSN10] Ankit KESHARWANI, Vaishali SADAPHAL et Maitreya NATU. “Empowering Bus Transportation System Using Wireless Sensor Networks”. Dans : *7th International Conference on High Performance Computing (HiPC 2010)*. Goa, India, déc. 2010.
- [Vrt] “La nouvelle dynamique stationnement”. Dans : *Ville Rail et Transports* 507 (nov. 2010), p. 30–33.
- [Li10] Zichuan LI. “Modeling arterial signal optimization with enhanced cell transmission formulations”. Dans : *Journal of Transportation Engineering* 137.7 (2010), p. 445–454.
- [PQ10] Rahul PUTHA et Luca QUADRIFOGLIO. “Using Ant Colony Optimization for Solving Traffic Signal Coordination in Oversaturated Networks”. Dans : *Transportation Research Board 89th Annual Meeting*. 10-2669. 2010.
- [XCBM10] Wuping XIN, Jinil CHANG, Bryan BERTOLI et A MOHAMAD. “Integrated adaptive traffic signal control with real-time decision support”. Dans : *Transportation Research Board 89th Annual Meeting*. 10-3511. 2010.
- [YAKS10] Khalil M. YOUSEF, Jamal N. AL-KARAKI et Ali M. SHATNAWI. “Intelligent Traffic Light Flow Control System Using Wireless Sensors Networks”. Dans : *Journal of Information Science and Engineering* 26.3 (mai 2010).
- [ZYL10] Lihui ZHANG, Yafeng YIN et Yingyan LOU. “Robust signal timing for arterials under day-to-day demand variations”. Dans : *Transportation Research Record : Journal of the Transportation Research Board* 2192.1 (2010), p. 156–166.
- [ZL10] Qiaoxian ZHENG et Ming LI. “The Methods of Traffic Circle Problem”. Dans : *International Conference on Logistics Engineering and Intelligent Transportation Systems (LEITS 2010)*. Nov. 2010, p. 1–4.

- [LWMH09] Henry X LIU, Xinkai WU, Wenteng MA et Heng HU. “Real-time queue length estimation for congested signalized intersections”. Dans : *Transportation research part C : emerging technologies* 17.4 (2009), p. 412–427.
- [Min09] MINISTÈRE DE L'ÉCOLOGIE, DE L'ÉNERGIE ET DU DÉVELOPPEMENT DURABLE ET DE LA MER. *Instruction Interministérielle sur la Signalisation Routière, sixième partie : feux de circulation permanents*. 2009.
- [MZ09] Hassnaa MOUSTAFA et Yan ZHANG. *Vehicular networks : techniques, standards, and applications*. Auerbach publications, 2009.
- [SL09] D SCHRANK et T LOMAX. *The 2009 Urban Mobility Report. College Station : Texas Transportation Institute, Texas A&M University*. 2009.
- [SS09] Matt SELINGER et Luke SCHMIDT. “A review of the cost, maintenance and reliability of popular adaptive traffic control technologies”. Dans : *Adaptive Traffic Control Systems in the United States* (2009).
- [SC+09] Pang SHAN-CHEN et al. “A Study on the Model of the Traffic Signs in the Traffic Circle”. Dans : *International Conference on Information Engineering and Computer Science (ICIECS 2009)*. Déc. 2009, p. 1–4.
- [SG09] Daniel SPERLING et Deborah GORDON. *Two billion cars : driving toward sustainability*. Oxford University Press, 2009.
- [VS09] Vasos VASSILIOU et Charalambos SERGIOU. “Performance Study of Node Placement for Congestion Control in Wireless Sensor Networks”. Dans : *WSN Workshop, held in conjunction with NTMS 2009*. 2009.
- [Xia+09] Laisheng XIAO et al. “Research on Traffic Monitoring Network and Its Traffic Flow Forecast and Congestion Control Model Based on Wireless Sensor Networks”. Dans : *Measuring Technology and Mechatronics Automation, International Conference on* 1 (2009), p. 142–147.
- [ZZ09] Xianyang ZENG et Hongxia ZHENG. “The Intelligent Control and Modeling of a Traffic Circle”. Dans : *International Conference on Information Engineering and Computer Science (ICIECS 2009)*. Déc. 2009, p. 1–4.
- [ZYC09] Fuqiang ZOU, Bo YANG et Yitao CAO. “Traffic light control for a single intersection based on wireless sensor network”. Dans : *9th International Conference on Electronic Measurement & Instruments (ICEMI 2009)*. Beijing, China, août 2009.
- [CGML08] I. CORREDOR, A.B. GARCÍA, J.F. MARTÍNEZ et P. LÓPEZ. “Wireless Sensor Network-based system for measuring and monitoring road traffic”. Dans : *6th Collaborative Electronic Communications and eCommerce Technology and Research (COLLECTeR 2008)*. Madrid, Spain, juin 2008.

- [DHS08] Richard W DENNEY, Larry HEAD et Kevin SPENCER. *Signal timing under saturated conditions*. Rap. tech. US Dept. of Transportation, Federal Highway Administration, Office of Operations, 2008.
- [HSS08] Aric A. HAGBERG, Daniel A. SCHULT et Pieter J. SWART. “Exploring network structure, dynamics, and function using NetworkX”. Dans : *Proceedings of the 7th Python in Science Conference (SciPy2008)*. Pasadena, CA USA, août 2008, p. 11–15.
- [HW08] Mordechai HAKLAY et Patrick WEBER. “Openstreetmap : User-generated street maps”. Dans : *Pervasive Computing, IEEE 7.4* (2008), p. 12–18.
- [HKV08] Amine HAOUI, Robert KAVALER et Pravin VARAIYA. “Wireless magnetic sensors for traffic surveillance”. Dans : *Transportation Research Part C : Emerging Technologies* 16.3 (2008).
- [HSZK08] Jianming HU, Jingyan SONG, Mingchen ZHANG et Xiaojing KANG. “Topology optimization for urban traffic sensor network”. Dans : *Tsinghua Science & Technology* 13.2 (2008).
- [LH08] Stefan LÄMMER et Dirk HELBING. “Self-control of traffic lights and vehicle flows in urban road networks”. Dans : *Journal of Statistical Mechanics : Theory and Experiment* 2008.04 (2008), P04019.
- [MY08] Wanjing MA et Xiaoguang YANG. “Design and Evaluation of an Adaptive Bus Signal Priority System Base on Wireless Sensor Network”. Dans : *11th International IEEE Conference on Intelligent Transportation System (ITSC 2008)*. Oct. 2008, p. 1073 –1077.
- [Mac+08] Robert P MACCUBBIN et al. *Intelligent transportation systems benefits, costs, deployment, and lessons learned : 2008 update*. Rap. tech. 2008.
- [Mur+08] Rohan Narayana MURTY et al. “Citysense : An urban-scale wireless sensor network and testbed”. Dans : *Technologies for Homeland Security, 2008 IEEE Conference on*. 2008.
- [Sug+08] Yuki SUGIYAMA et al. “Traffic jams without bottlenecks—experimental evidence for the physical mechanism of the formation of a jam”. Dans : *New Journal of Physics* 10.3 (2008), p. 033001.
- [TQSS08] Malik TUBAISHAT, Qi QI, Yi SHANG et Hongchi SHI. “Wireless Sensor-Based Traffic Light Control”. Dans : *5th IEEE Conference on Consumer Communications and Networking (CCNC 2008)*. Las Vegas, USA, fév. 2008.
- [US 08] US DEPT. OF TRANSPORTATION, FEDERAL HIGHWAY ADMINISTRATION, OFFICE OF OPERATIONS. *Traffic Signal Timing Manual*. Rap. tech. 2008. URL : <http://ops.fhwa.dot.gov/publications/fhwahop08024/>.

- [VH08] András VARGA et Rudolf HORNIG. “An overview of the OMNeT++ simulation environment”. Dans : *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*. ICST (Institute for Computer Sciences, Social-Informatics et Telecommunications Engineering). 2008, p. 60.
- [Yin08] Yafeng YIN. “Robust optimal traffic signal timing”. Dans : *Transportation Research Part B : Methodological* 42.10 (2008), p. 911–924.
- [BPC07] O. BANIAS, R.-E. PRECUP et D.-I. CURIAE. “Problem Setting and Modeling in Vehicles and Pedestrians Traffic Control Using Sensor Networks”. Dans : *4th International Symposium on Applied Computational Intelligence and Informatics (SACI 2007)*. Mai 2007, p. 83–88.
- [BKRF07] W.J. BARNES, T.I. KING, H.H. REFAI et J.E. FAGAN. “A Wireless Sensor Network Simulation for Highway Intersection Collision Prevention”. Dans : *Conference on Intelligent Transportation Systems (ITSC 2007)*. 2007, p. 173–177.
- [DP07] Maëlle DUCARNE et Thomas PÉRINEAU. *Commande adaptative et optimale d’un carrefour à feu*. Rap. tech. 2007.
- [Hou+07] N.B. HOUNSELL et al. “Using global positioning system for bus priority in London : traffic signals close to bus stops”. Dans : *Intelligent Transport Systems, IET* 1.2 (juin 2007), p. 131–137.
- [Sen] *Measurement of Sensys<sup>TM</sup> Wireless Sensor Battery Life*. Rap. tech. Sensys Networks, 2007.
- [MK07] Luz Elena Y. MIMBELA et Lawrence A. KLEIN. *Summary of vehicle detection and surveillance technologies used in intelligent transportation systems*. Federal Highway Administration, Intelligent Transportation Systems Joint Program Office, 2007.
- [PB07] David R. PRITCHARD et Ron N. BULIUNG. “Transportation Analysis and Research Using Free/Open Source Software”. Dans : (2007).
- [TSS07] M. TUBAISHAT, Y. SHANG et H. SHI. “Adaptive traffic light control with wireless sensor networks”. Dans : *4th IEEE Conference on Consumer Communications and Networking*. 2007.
- [WPL07] Y. WEN, JL PAN et JF LE. “Survey on Application of Wireless Sensor Networks Traffic Monitoring”. Dans : *First International Conference on Transportation*. T. 246. American Society of Civil Engineers, 1801 Alexander Bell Drive Reston VA 20191-4400 USA. 2007, p. 2079–2084.
- [ACCJ06] Montasir ABBAS, Hassan CHARARA, Nadeem CHAUDHARY et Younsu JUNG. *Distributed architecture and algorithm for robust real-time progress evaluation and improvement*. Texas Transportation Institute, Texas A & M University System, 2006.

- [BUB06] Scott BEAIRD, Thomas URBANIK et Darcy M BULLOCK. “Traffic signal phase truncation in event of traffic flow restriction”. Dans : *Transportation Research Record : Journal of the Transportation Research Board* 1978.1 (2006), p. 87–94.
- [BMP06] Florence BOILLOT, Sophie MIDENET et Jean-Claude PIERRELÉE. “The real-time urban traffic control system CRONOS : Algorithm and experiments”. Dans : *Transportation Research Part C : Emerging Technologies* 14.1 (2006), p. 18–38.
- [CCCT06] Wenjie CHEN, Lifeng CHEN, Zhanglong CHEN et Shiliang TU. “WITS : A Wireless Sensor Network for Intelligent Transportation System”. Dans : *First International Multi-Symposiums on Computer and Computational Sciences (IMSCCS 2006)*. T. 2. Juin 2006, p. 635–641.
- [CLP06] Paolo CRUCITTI, Vito LATORA et Sergio PORTA. “Centrality measures in spatial networks of urban streets”. Dans : *Physical Review E* 73.3 (2006).
- [LZ06a] Z. LIAO et L. ZHAO. “Wireless Sensor Networks Help to Improve the Traffic Safety in Residential Communities”. Dans : *6th International Conference on ITS Telecommunications*. 2006, p. 973–978.
- [LZ06b] Zaiyi LIAO et Lian ZHAO. “Wireless Sensor Networks Help to Improve the Traffic Safety in Residential Communities”. Dans : *6th International Conference on ITS Telecommunications Proceedings*. Juin 2006, p. 973–978.
- [Min06] MINNESOTA DEPARTMENT OF TRANSPORTATION. *Traffic Signals 101*. 2006. URL : <http://www.dot.state.mn.us/trafficeng/publ/signals101/index.html>.
- [OMT06] OMT. *Conférence sur les déplacements écologiques en Europe*. 2006. URL : <http://www.busandcoach.travel/download/factsheets/frgreen.pdf>.
- [PCL06a] Sergio PORTA, Paolo CRUCITTI et Vito LATORA. “The network analysis of urban streets : a dual approach”. Dans : *Physica A : Statistical Mechanics and its Applications* 369.2 (2006).
- [PCL06b] Sergio PORTA, Paolo CRUCITTI et Vito LATORA. “The network analysis of urban streets : a primal approach”. Dans : *Environment and Planning B : Planning and Design* 33 (2006).
- [Tan+06] Feilong TANG et al. “Combining Wireless Sensor Network with Grid for Intelligent City Traffic”. Dans : *11th Asia-Pacific Conference, ACSAC 2006*. T. 4186. Shanghai, China, sept. 2006, p. 260–269.
- [YLC06] Yuan L. Y., Zhu Y. L. et Qu L. C. “Research and implementation of transportation monitoring system based on distributed wireless sensor network”. Dans : *Computer Engineering* 32 (2006), p. 249–251.

- [Che+05] S.Y. CHEUNG et al. “Traffic measurement and vehicle classification with single magnetic sensor”. Dans : *Transportation Research Record : Journal of the Transportation Research Board* 1917 (déc. 2005).
- [Gor+05] R.L. GORDON et al. *Traffic control systems handbook*. US Dept. of Transportation, Federal Highway Administration, Office of Operations, 2005. URL : <http://ops.fhwa.dot.gov/publications/fhwahop06006/>.
- [Nts] *National Traffic Signal Report Card, National Transportation Operations Coalition*. 2005.
- [Sey05] John S SEYBOLD. *Introduction to RF propagation*. John Wiley & Sons, 2005.
- [SHSX05] Xi SHI, Jianming HU, Jingyan SONG et Yangsheng XU. “Parameter optimization of wireless communication system in video surveillance network”. Dans : *Information Acquisition, 2005 IEEE International Conference on*. IEEE. 2005.
- [WLZS05] Chen WENJIE, Chen LIFENG, Chen ZHANGLONG et Tu SHILIANG. “A realtime dynamic traffic control system based on wireless sensor network”. Dans : *International Conference Workshops on Parallel Processing (ICPP 2005)*. Juin 2005, p. 258 –264.
- [Wen+05] Chen WENJIE et al. “An intelligent guiding and controlling system for transportation network based on wireless sensor network technology”. Dans : *Fifth International Conference on Computer and Information Technology (CIT 2005)*. Sept. 2005, p. 810 –814.
- [CS04a] Tang-Hsien CHANG et Guey-Yin SUN. “Modeling and optimization of an oversaturated signalized network”. Dans : *Transportation Research Part B : Methodological* 38.8 (2004), p. 687–707.
- [CCV04] Sinem COLERI, Sing Yiu CHEUNG et Pravin VARAIYA. “Sensor Networks for Monitoring Traffic”. Dans : *In Allerton Conference on Communication, Control and Computing*. 2004.
- [CS04b] Gábor CSÁNYI et Balázs SZENDRŐI. “Fractal–small-world dichotomy in real-world networks”. Dans : *Physical Review E* 70.1 (2004), p. 016122.
- [DRK04] Francois DION, Hesham RAKHA et Youn-Soo KANG. “Comparison of delay estimates at under-saturated and over-saturated pre-timed signalized intersections”. Dans : *Transportation Research Part B : Methodological* 38.2 (2004), p. 99–122.
- [Gur+04] N. GURA et al. “Comparing elliptic curve cryptography and RSA on 8-bit CPUs”. Dans : *Cryptographic Hardware and Embedded Systems-CHES 2004* (2004), p. 925–943.
- [HMS04] N. B. HOUNSELL, F. N. MCLEOD et B. P. SHRESTHA. “Bus priority at traffic signals : Investigating the options”. Dans : *IEE Conference Publication*. 501. London : Institution of Engineering et Technology, 2004, p. 287–294.

- [IA04] Mika ISHIZUKA et Masaki AIDA. “Performance study of node placement in sensor networks”. Dans : *24th International Conference on Distributed Computing Systems Workshops (ICDCS)*. Tokyo, Japan, 2004.
- [LC04] Hong K LO et Andy HF CHOW. “Control strategies for oversaturated traffic”. Dans : *Journal of Transportation Engineering* 130.4 (2004), p. 466–478.
- [STYW04] H. SAWANT, Jindong TAN, Qingyan YANG et Qizhi WANG. “Using Bluetooth and sensor networks for intelligent transportation systems”. Dans : *7th International IEEE Conference on Intelligent Transportation Systems*. Oct. 2004, p. 767 –772.
- [WVJK04] M. WIERING, Veenen J. V., Vreeken J. et A. KOOPMAN. “Intelligent Traffic Light Control”. Dans : *Technical report UU-CS-2004-029*. 2004.
- [WVVK04] M. WIERING, J. VREEKEN, J. van VEENEN et A. KOOPMAN. “Simulation and optimization of traffic in a city”. Dans : *Intelligent Vehicles Symposium, 2004 IEEE*. Juin 2004, p. 453 –458.
- [Iee] “Coexistence of Wireless Personal Area Networks with Other Wireless Devices Operating in Unlicensed Frequency Bands”. Dans : *IEEE Standard 802.15.2-2003* (2003).
- [Pap+03] Markos PAPAGEORGIOU et al. “Review of road traffic control strategies”. Dans : *Proceedings of the IEEE* 91.12 (2003), p. 2043–2067.
- [Pen03] Matthew PENROSE. *Random Geometric Graphs*. Oxford Studies in Probability, 2003.
- [TG03] Kim J. T. et Courage K. G. “Evaluation and design of maximum green time settings for traffic actuated control”. Dans : *Transportation Research Record* 1852 (2003), p. 246–255.
- [ASSC02] Ian F. AKYILDIZ, Weilian SU, Yogesh SANKARASUBRAMANIAM et Erdal CAYIRCI. “Wireless sensor networks : a survey”. Dans : *Computer Networks* 38.4 (2002), p. 393–422.
- [CS02a] Xiao-Feng CHEN et Zhong-ke SHI. “Real-coded genetic algorithm for signal timing optimization of a single intersection”. Dans : *Machine Learning and Cybernetics, 2002. Proceedings. 2002 International Conference on*. T. 3. IEEE. 2002, p. 1245–1248.
- [CS02b] Xiao-Feng CHEN et Zhong-Ke SHI. “Real-coded genetic algorithm for signal timing optimization of a single intersection”. Dans : *2002 International Conference on Machine Learning and Cybernetics*. T. 3. 2002, p. 1245 –1248.
- [DFMN02] D. W. DEY, S. FITZSIMONS, A. MORRIS et D. NG. “Adaptive Traffic Signal Interconnect in Menlo Park and Sunnyvale, CA”. Dans : (2002).

- [JKZ02] T. JURDZIŃSKI, M. KUTYŁOWSKI et J. ZATOPIAŃSKI. “Efficient algorithms for leader election in radio networks”. Dans : *21th annual symposium on Principles of distributed computing*. 2002.
- [Per+02] A. PERRIG et al. “SPINS : Security protocols for sensor networks”. Dans : *Wireless networks* 8.5 (2002), p. 521–534.
- [TNKF02] Sei TAKAHASHI, Hideo NAKAMURA, Hiroshi KAZAMA et Tomokazu FUJIKURA. “Genetic algorithm approach for adaptive offset optimization for the fluctuation of traffic flow”. Dans : *The IEEE 5th International Conference on Intelligent Transportation Systems*. 2002, p. 768 –772.
- [WZ02] Wu WEI et Yi ZHANG. “FL-FN based traffic signal control”. Dans : *IEEE International Conference on Fuzzy Systems*. T. 1. 2002, p. 296 –300.
- [Bin01] Ella BINGHAM. “Reinforcement learning in neurofuzzy traffic signal control”. Dans : *European Journal of Operational Research* 131.2 (2001), p. 232–241.
- [GPA01] N.H. GARTNER, F.J. POORAN et C.M. ANDREWS. “Implementation of the OPAC adaptive control strategy in a traffic signal network”. Dans : *IEEE Intelligent Transportation Systems*. 2001, p. 195 –200.
- [HHRC01] Wolfgang S HOMBURGER, Jerome W HALL, William R REILLY et Sullivan Edward C. *Fundamentals of Traffic Engineering (15th ed.)*, UCB-ITS-CN-01-1. Jan. 2001.
- [HB01] Serge P HOOGENDOORN et Piet HL BOVY. “State-of-the-art of vehicular traffic flow modelling”. Dans : *Proceedings of the Institution of Mechanical Engineers, Part I : Journal of Systems and Control Engineering* 215.4 (2001), p. 283–303.
- [CM00] DingXin CHENG et Carroll J MESSER. “Modification of Webster’s Minimum Delay Cycle Length Equation Based on HCM 2000”. Dans : (2000).
- [CSS00] Debashish CHOWDHURY, Ludger SANTEN et Andreas SCHADSCHNEIDER. “Statistical physics of vehicular traffic and some related systems”. Dans : *Physics Reports* 329.4 (2000), p. 199–329.
- [Ker00] Boris S KERNER. “Phase transitions in traffic flow”. Dans : *Traffic and Granular Flow’99*. Springer, 2000, p. 253–283.
- [Kna00] Ara N. KNAIAN. “A Wireless Sensor Network for Smart Roadbeds and Intelligent Transportation Systems”. Mém.de maîtr. Massachusetts Institute of Technology, juin 2000.
- [KSS00] Wolfgang KNOSPE, Ludger SANTEN, Andreas SCHADSCHNEIDER et Michael SCHRECKENBERG. “Towards a realistic microscopic description of highway traffic”. Dans : *Journal of Physics A : Mathematical and general* 33.48 (2000), p. L477.

- [MK00] Luz Elena Y MIMBELA et Lawrence A KLEIN. *A Summary of vehicle detection and surveillance technologies used in intelligent transportation systems*. [Washington, DC] : Federal Highway Administration, Intelligent Transportation Systems Joint Program Office, 2000.
- [PMUI00] Byungkyu " Brian " PARK, Carroll J MESSER et Thomas URBANIK II. "Enhanced genetic algorithm for signal-timing optimization of oversaturated intersections". Dans : *Transportation Research Record : Journal of the Transportation Research Board* 1727.1 (2000), p. 32–41.
- [Wie00] Marco WIERING. "Multi-Agent Reinforcement Learning for Traffic Light Control". Dans : *Seventeenth International Conference on Machine Learning*. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 2000, p. 1151–1158.
- [BA99] Albert-László BARABÁSI et Réka ALBERT. "Emergence of scaling in random networks". Dans : *Science* 286.5439 (oct. 1999).
- [Mar99] K MARQUESS. "Physical model sub-group discussion and questions". Dans : *IEEE 802.15/138R0* (1999).
- [BSS98] Robert BARLOVIC, Ludger SANTEN, Andreas SCHADSCHNEIDER et Michael SCHRECKENBERG. "Metastable states in cellular automata for traffic flow". Dans : *The European Physical Journal B-Condensed Matter and Complex Systems* 5.3 (1998), p. 793–800.
- [KF98] J. H. KELL et I. J. FULLERTON. "Manual of Traffic Signal Design, 3rd ed. Englewood Cliffs, NJ : Inst. Transp. Eng., Prentice-Hall". Dans : (1998).
- [Nat98] NATIONAL ELECTRICAL MANUFACTURERS ASSOCIATION. *NEMA TS2 Traffic Controller Assemblies with NTCIP Requirements*. 1998.
- [Sch98] Andreas SCHAFER. "The global demand for motorized mobility". Dans : *Transportation Research Part A : Policy and Practice* 32.6 (1998), p. 455–477.
- [SBG98] A SKABARDONIS, R L BERTINI et B R GALLAGHER. "Development and application of control strategies for signalized intersections in coordinated systems". Dans : *Transportation research record* 1634 (1998), p. 110–117.
- [WS98] Duncan J. WATTS et Steven H. STROGATZ. "Collective dynamics of 'small-world' networks". Dans : *Nature* 393.6684 (1998).
- [ALB97] Ghassan ABU-LEBDEH et Rahim F BENEKOHAL. "Development of traffic control and queue management procedures for oversaturated arterials". Dans : *Transportation Research Record : Journal of the Transportation Research Board* 1603.1 (1997), p. 119–127.
- [AM97] Gye-Hyeong AHN et Randy B. MACHEMEHL. *Methodology for Traffic Signal Timing in Oversaturated Arterial Networks*. Rap. tech. 1997.
- [Akç97] Rahmi AKÇELİK. "Lane-by-lane modelling of unequal lane use and flares at roundabouts and signalised intersections : the SIDRA solution". Dans : *Traffic Engineering+ Control* 38.7 (1997), p. 388–99.

- [PMN97] André de PALMA, Fabrice MARCHAL et Yurii NESTEROV. “METROPOLIS : Modular system for dynamic traffic simulation”. Dans : *Transportation Research Record : Journal of the Transportation Research Board* 1607.1 (1997), p. 178–184.
- [SC97] J.C. SPALL et D.C. CHIN. “Traffic-responsive signal timing for system-wide traffic control”. Dans : *Transportation Research Part C : Emerging Technologies* 5.3-4 (1997), p. 153–163.
- [Cha96] Edmond Chin-Ping CHANG. “Guidelines for Actuated Controllers in Coordinated Systems”. Dans : *Transportation Research Record : Journal of the Transportation Research Board* 1554 (1996), p. 61–73.
- [CVS96] Gang-len CHANG, Meenakshy VASUDEVAN et Chih-chiang SU. “Modelling and evaluation of adaptive bus-preemption control with and without Automatic Vehicle Location systems”. Dans : *Transportation Research Part A : Policy and Practice* 30.4 (juil. 1996), p. 251–268.
- [SG96] Chronis STAMATIADIS et Nathan H GARTNER. “MULTIBAND-96 : a program for variable-bandwidth progression optimization of multiarterial traffic networks”. Dans : *Transportation Research Record : Journal of the Transportation Research Board* 1554.1 (1996), p. 9–17.
- [MR95] Michael MOLLOY et Bruce REED. “A critical point for random graphs with a given degree sequence”. Dans : *Random Structures and Algorithms* 6.2-3 (1995).
- [Dag94] Carlos F DAGANZO. “The cell transmission model : A dynamic representation of highway traffic consistent with the hydrodynamic theory”. Dans : *Transportation Research Part B : Methodological* 28.4 (1994), p. 269–287.
- [L.93] Orcutt F. L. “The Traffic Signal Book. Englewood Cliffs, NJ : Prentice-Hall”. Dans : (1993).
- [Boi+92] F. BOILLOT et al. “Optimal signal control of urban traffic networks”. Dans : 472. London, 1992, p. 182–186.
- [HMS92] K. Larry HEAD, Pitu B. MIRCHANDANI et Dennis SHEPPARD. “Hierarchical Framework for Real-Time Traffic Control”. Dans : *Transportation Research Record 1360, Transportation Research Board, National Research Council, Washington DC* (1992).
- [HKD92] W.S. HOMBURGER, J.H. KELL et Perkins D.D. “Fundamentals of Traffic Engineering, Institute of Transportation Studies, University of California at Berkeley, (13th edition)”. Dans : (1992).
- [NS92] Kai NAGEL et Michael SCHRECKENBERG. “A cellular automaton model for freeway traffic”. Dans : *Journal de physique I* 2.12 (1992), p. 2221–2229.
- [RB91] D.I. ROBERTSON et R.D. BRETHERTON. “Optimizing networks of traffic signals in real time-the SCOOT method”. Dans : *IEEE Transactions on Vehicular Technology* 40.1 (fév. 1991).

- [Rou91] NM ROUPHAIL. "Cycle-by-cycle analysis of congested flow at signalized intersections". Dans : *ITE (Institute of Transportation Engineers) Journal;(United States)* 61.3 (1991).
- [BB90] RD BRETHERTON et GT BOWEN. "Recent enhancements to SCOOT-SCOOT version 2.4". Dans : *Road Traffic Control, 1990., Third International Conference on*. IET. 1990, p. 95–98.
- [Kim90] Y KIM. *Development of Optimization Models for Signalized Intersections During Oversaturated Conditions*. Rap. tech. Ph. D. Dissertation, Texas A&M University, College Station, Texas, 1990.
- [Lee90] C.C. LEE. "Fuzzy logic in control systems : fuzzy logic controller. II". Dans : *IEEE Transactions on Systems, Man and Cybernetics* 20.2 (1990), p. 419–435.
- [MW90] Roess R.P. MCSHANE W.R. "Traffic Engineering, Prentice Hall". Dans : (1990).
- [GZE89] Courage K. G., Luh J. Z. et Wallace C. E. "Development of guidelines for implementing computerized timing designs at traffic actuated signals, Two Volumes on Arterial System Implementation, Transp. Res. Center, Gainesville". Dans : (1989).
- [Rat88] Ajay K RATHI. "A control scheme for high traffic density sectors". Dans : *Transportation Research Part B : Methodological* 22.2 (1988), p. 81–101.
- [CL87] B. M. CHARD et C. J LINES. "Transyt - the latest developments". Dans : *Traffic engineering & control* 28.7-8 (1987), p. 387–390.
- [LRKS86] Edward B LIEBERMAN, AK RATHI, GF KING et SI SCHWARTZ. *Congestion-based control scheme for closely spaced, high traffic density networks*. 1057. 1986.
- [Lin85] F. B. LIN. "Optimal timing settings and detector lengths of presence model full-actuated control". Dans : *Transportation Research Record* 1010 (1985), p. 37–45.
- [SY84] J SHIBATA et T YAMAMOTO. "Detection and control of congestion in urban road networks". Dans : *Traffic engineering & control* 25.9 (1984), p. 438–444.
- [Gar83] Nathan H GARTNER. "OPAC : A demand-responsive strategy for traffic signal control". Dans : *Transportation Research Record* 906 (1983).
- [HFT83] J.J. HENRY, J.L. FARGES et J. TUFFAL. "The prodyn real time traffic algorithm". Dans : *4th Conference on Control Transportation System* 472 (1983), p. 305–309.
- [MSS81] Panos G MICHALOPOULOS, Gregory STEPHANOPOULOS et George STEPHANOPOULOS. "An application of shock wave theory to traffic signal control". Dans : *Transportation Research Part B : Methodological* 15.1 (1981), p. 35–51.

- [SD80] A.G. SIMS et K.W. DOBINSON. “The Sydney coordinated adaptive traffic (SCAT) system philosophy and benefits”. Dans : *IEEE Transactions on Vehicular Technology* 29.2 (mai 1980).
- [MS77] Panos G MICHALOPOULOS et George STEPHANOPOULOS. “Oversaturated signal systems with queue length constraints-I : Single intersection, II : Systems of intersections”. Dans : *Transportation Research* 11.6 (1977), p. 413–428.
- [DG76] GC D’ANS et DC GAZIS. “Optimal control of oversaturated store-and-forward transportation networks”. Dans : *Transportation Science* 10.1 (1976), p. 1–19.
- [LCP75] Bumjung LEE, Kenneth W CROWLEY et Louis J PIGNATARO. “Better use of signals under oversaturated flows”. Dans : *Transportation Research Board Special Report* 153 (1975).
- [RB74] DI ROBERTSON et RD BRETHERTON. “Optimum control of an intersection for any known sequence of vehicle arrivals”. Dans : *Proceedings of the 2nd IFAC/IFIP/IFORS Symposium on Traffic Control and Transportation Systems*. 1974.
- [ST74] MG SINGH et H TAMURA. “Modelling and hierarchical optimization for oversaturated urban road traffic networks”. Dans : *International Journal of Control* 20.6 (1974), p. 913–934.
- [CES71] Edward G COFFMAN, Melanie ELPHICK et Arie SHOSHANI. “System deadlocks”. Dans : *ACM Computing Surveys (CSUR)* 3.2 (1971), p. 67–78.
- [Gor69] Robert L GORDON. “A technique for control of traffic at critical intersections”. Dans : *Transportation Science* 3.4 (1969), p. 279–288.
- [Lon68] D LONGLEY. “A control strategy for a congested, computer-controlled traffic network”. Dans : *Transportation Research/UK/* (1968).
- [GP65] Denos C GAZIS et Renfrey B POTTS. “The oversaturated intersection”. Dans : *Proc. 2nd Int. Symp. on the Theory of Road Traffic Flow*. 1965, p. 221–237.
- [Gaz64] Denos C GAZIS. “Optimum control of a system of oversaturated intersections”. Dans : *Operations Research* 12.6 (1964), p. 815–831.
- [ER59] Paul ERDÖS et Alfréd RÉNYI. “On Random Graphs I”. Dans : *Publicationes Mathematicae* 6 (1959).
- [Gil59] Edgar N. GILBERT. “Random graphs”. Dans : *Annals of Mathematical Statistics* 30 (1959).
- [CHM58] Robert E CHANDLER, Robert HERMAN et Elliott W MONTROLL. “Traffic dynamics : studies in car following”. Dans : *Operations research* 6.2 (1958), p. 165–184.
- [Web58] Fo Vo WEBSTER. *Traffic signal settings*. Rap. tech. 1958.

- [LW55] Michael J LIGHTHILL et Gerald Beresford WHITHAM. “On kinematic waves. II. A theory of traffic flow on long crowded roads”. Dans : *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 229.1178 (1955), p. 317–345.
- [War52] John Glen WARDROP. “ROAD PAPER. SOME THEORETICAL ASPECTS OF ROAD TRAFFIC RESEARCH.” Dans : *ICE Proceedings : Engineering Divisions*. T. 1. 3. Thomas Telford. 1952, p. 325–362.
- [GSE46] Bruce Douglas GREENSHIELDS, Donald SCHAPIRO et Elroy L ERICKSEN. *Traffic performance at urban street intersections*. Rap. tech. 1946.
- [GCM+35] BD GREENSHIELDS, Ws CHANNING, Hh MILLER et al. “A study of traffic capacity”. Dans : *Highway research board proceedings*. T. 1935. National Research Council (USA), Highway Research Board. 1935.
- [802] *802.11-2012 - IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. Rap. tech. IEEE Std 802.11™-2012. IEEE-Inst.
- [Abd] Zeeshan Raza ABDY. “Fuzzy Logic Traffic Signal Control”. Dans : ().
- [KTYR] Gholamreza KHORASANI, Ashkan TATARI, Ali YADOLLAHI et Milad RAHIMI. *Evaluation of Intelligent Transport System in Road Safety*.
- [Weba] *Les transports intelligents*. URL : <http://www.transport-intelligent.net/>.
- [Les] Matthieu LESCIEUX. *Application à la commande floue*. URL : [http://auto.polytech.univ-tours.fr/automatique/AUA/ressources/Introduction\\\_logique\\\_floue.ppt](http://auto.polytech.univ-tours.fr/automatique/AUA/ressources/Introduction\_logique\_floue.ppt).
- [Webb] *Performance evaluation of Los Angeles adaptive traffic control system (ATCS) on an arterial corridor*. URL : <http://leonard.csusb.edu/outreach/documents/PERFORMANCEEVALUATIONOFLOSANGELES.pdf>.
- [Webc] *Poisson inter-arrival time generation & CDF function*. URL : <http://engineering.dartmouth.edu/~eric/engs027/outlines/19DiscrEvSim.pdf>.
- [Uni] UNIVERSITY OF IDAHO-MOSCOW. *Traffic Signal Training*. URL : <http://www.webs1.uidaho.edu/niattproject/>.
- [Wik] WIKIPEDIA. URL : [http://fr.wikipedia.org/wiki/Embouteillage\\_%28route%29](http://fr.wikipedia.org/wiki/Embouteillage_%28route%29).
- [ZCZW] B. ZHOU, J. CAO, X. ZENG et H. WU. “Adaptive Traffic Light Control in Wireless Sensor Network-Based Intelligent Transportation System”. Dans : *Vehicular Technology Conference Fall (VTC 2010-Fall)*, p. 1–5.



# Contrôle et gestion du trafic routier urbain par un réseau de capteurs sans fil

Sébastien FAYE

**RESUMÉ.** Cette thèse vise à étudier l'emploi de systèmes distribués afin de mettre en œuvre des *systèmes de transport intelligents* grâce à un *réseau de capteurs sans fil*. Les contributions présentées visent à démontrer la pertinence et l'intérêt que de tels systèmes peuvent représenter, face aux contraintes fixées par l'infrastructure routière actuelle.

**MOTS-CLEFS :** systèmes de transport intelligents, systèmes distribués, réseaux de capteurs sans fil.

---

**ABSTRACT.** The aim of this thesis is to study the use of distributed systems in order to implement *intelligent transportation systems* via a *wireless sensor network*. The presented contributions aim to demonstrate the relevance and the value of these systems in the face of the constraints of existing road infrastructure.

**KEY-WORDS :** intelligent transportation systems, distributed systems, wireless sensor networks.

