



**HAL**  
open science

## Partage d'informations dans les réseaux de communication inter-véhiculaire

Nicolas Cenerario

► **To cite this version:**

Nicolas Cenerario. Partage d'informations dans les réseaux de communication inter-véhiculaire. Informatique [cs]. Université de Valenciennes et du Hainaut-Cambrésis, 2010. Français. NNT : 2010VALE0013 . tel-03004104

**HAL Id: tel-03004104**

**<https://uphf.hal.science/tel-03004104v1>**

Submitted on 13 Nov 2020

**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.

Thèse de Doctorat

---

## Partage d'informations dans les réseaux de communication inter-véhiculaire

---

Présentée par  
**Nicolas CENERARIO**

Pour l'obtention du grade de  
**Docteur de l'Université de Valenciennes  
et du Hainaut-Cambésis**

Spécialité : Automatique et Informatique  
des Systèmes Industriels et Humains

Discipline : Informatique

Soutenue publiquement le 31 mars 2010 devant le jury composé de :

Directeur de thèse : Sylvain LECOMTE, Professeur à l'Université de Valenciennes  
Co-Directeur de thèse : Thierry DELOT, Maître de conférences à l'Université de Valenciennes  
Rapporteurs : Bruno DEFUDE, Professeur à TELECOM & Management SudParis  
Florence SEDES, Professeur à l'Université Paul Sabatier de Toulouse  
Examineurs : Sergio ILARRI, Maître de conférences à l'Université de Saragosse (Espagne)  
Vasile-Marian SCUTURICI, Maître de conférences à l'INSA de Lyon  
Président du jury : Gilles GRIMAUD, Professeur à l'Université de Lille 1

*Comprendre est le commencement d'approuver.*

[Baruch Spinoza]



---

## Remerciements

---

Faire une thèse n'est pas donné à tout le monde. Si par moment, la détermination m'a fait défaut, les gens qui m'entouraient m'ont toujours soutenu et m'ont poussé à m'accrocher. Sans eux, les choses se seraient passées différemment. Je remercie donc toutes les personnes qui ont contribué, de près ou de loin, à la réussite de ce projet.

Tout d'abord, je tiens à remercier ceux qui m'ont donné l'opportunité d'entreprendre ma thèse. La première personne que je remercie est Thierry Delot, mon encadrant de thèse, pour l'intérêt qu'il m'a porté depuis des années. Depuis mon IUP, il a été celui qui m'a encadré lors de nombreux projets. A l'époque, il m'a fait découvrir les Pocket PCs. C'est lui qui m'a passé le virus des nouvelles technologies et qui a révélé mon intérêt pour la recherche. C'est avec lui que j'ai partagé aussi bien les joies que les doutes qui ont ponctué nos travaux. Aujourd'hui, aucun merci n'est assez fort pour représenter la gratitude que je lui dois. Sans lui, rien n'aurait été possible.

La deuxième personne que je remercie est mon directeur de thèse, Sylvain Lecomte, qui a permis à Thierry de m'encadrer. Durant toute la durée de ma thèse, il a su respecter mon autonomie tout en me prodiguant les conseils les plus avisés. Son intégrité professionnelle sans faille font de lui une personne sur laquelle j'ai pu compter en toute circonstance.

Je remercie ensuite les personnes qui ont accepté de consacrer du temps à l'évaluation de mes travaux. Les rapporteurs, Florence Sedes et Bruno Defude, ont accepté de passer beaucoup de temps à la lecture et à l'analyse de mon manuscrit. Leurs interrogations, toujours justifiées, m'ont permis d'avancer jusqu'au jour de la soutenance. Je remercie également les examinateurs, Sergio Ilarri et Vasile-Marian Scuturici, et le président du jury, Gilles Grimaud, pour l'intérêt qu'ils ont porté à mes travaux. Rien ne leur imposait de participer à ma soutenance et pourtant, ils ont répondu présent. Ils ont sacrifié une partie de leur emploi du temps pour ma soutenance et je les remercie très sincèrement.

Je remercie aussi toutes les personnes qui ont participé au projet VESPA. Quelque soit leur degré d'implication, leur présence et leur intérêt pour mes travaux m'ont permis de prendre confiance

en moi. Grâce aux personnes que j'ai croisées en conférence et qui m'ont aidé à réaliser mes démonstrations, celles qui ont décidé de rejoindre l'aventure en proposant de nouvelles pistes de recherche et celles qui m'ont simplement encouragé, j'ai pu trouver les ressources nécessaires pour aller jusqu'au bout. Un grand merci donc, à toutes ces personnes qui se reconnaîtront.

Je remercie ensuite Jocelyne et Pierre (mes parents), Guillaume et Julie (mon frère et sa copine), mes grands-parents, mes oncles et tantes, mes cousins et cousines et bien évidemment Dana pour m'avoir encouragé et pour avoir rendu cette période moins difficile.

Merci aussi à mes amis du laboratoire qui ont su faire du « rez-de-chaussé » un endroit convivial où le travail ne semblait pas si difficile. Tous les autres amis qui m'ont permis de m'évader et de me faire penser à d'autres choses. Enfin, je remercie celle qui m'a accompagné dans la dernière ligne droite et qui a su faire preuve de patience. Je sais que je n'ai pas toujours été facile à vivre et pourtant, je n'ai jamais été abandonné.

Mille mercis pour toutes ces personnes que j'ai la chance de compter dans mes amis et dans ma famille.

Enfin, même si ces trois ans semblent être les trois seuls qui comptent, toutes les années précédentes m'ont permis de me construire jusqu'à finalement, faire de moi celui que je suis devenu. Je remercie toutes les personnes qui ont contribué à faire de moi la personne que je suis.

---

# Table des matières

---

<b>Chapitre 1 :</b>	<b>Introduction</b>	<b>1</b>
1	Motivations . . . . .	2
2	Notre contexte du partage d'informations . . . . .	3
2.1	Définitions . . . . .	3
2.2	Contraintes de l'environnement . . . . .	4
2.3	Différents problèmes à résoudre . . . . .	5
3	Objectifs de la thèse . . . . .	5
4	Plan de ce mémoire . . . . .	6
<b>Chapitre 2 :</b>	<b>Un domaine de recherche récent</b>	<b>7</b>
1	Les différentes architectures . . . . .	8
1.1	La robustesse d'une infrastructure . . . . .	9
1.2	La coopération entre véhicules . . . . .	10
1.3	Une architecture hybride . . . . .	12
2	Protocoles de communication . . . . .	13
2.1	Protocoles de routage . . . . .	14

---

2.2	Protocoles géocast . . . . .	15
2.3	Protocoles de dissémination . . . . .	17
3	Gestion des informations . . . . .	19
3.1	Evaluation de la pertinence . . . . .	20
3.2	La gestion des ressources . . . . .	21
3.3	Interrogation des données . . . . .	22
4	Autres travaux . . . . .	24
5	Conclusion . . . . .	26

---

**Chapitre 3 : Evaluation de la pertinence 29**

---

1	Représentation des informations . . . . .	30
1.1	Classification des événements . . . . .	30
1.2	Représentation des données . . . . .	32
1.3	Exemples de messages échangés . . . . .	34
2	Estimation de la pertinence des événements . . . . .	35
2.1	Vecteur mobilité & direction . . . . .	35
2.2	Probabilité de rencontre . . . . .	40
3	Conclusion . . . . .	45

---

**Chapitre 4 : Evaluation continue de requêtes 47**

---

1	Définitions . . . . .	48
2	Besoins . . . . .	49
2.1	Hypothèses . . . . .	49
2.2	Besoins . . . . .	50
3	Difficultés . . . . .	50
4	Optimisations . . . . .	50
4.1	Evaluation incrémentale . . . . .	52
4.2	Monitoring de l'évaluateur . . . . .	57
5	Conclusion . . . . .	62

---

<b>Chapitre 5 :</b>	<b>Dissémination de l'information</b>	<b>63</b>
1	Objectifs liés à la dissémination . . . . .	64
2	Protocole de dissémination . . . . .	66
2.1	Unicast, Multicast ou Broadcast ? . . . . .	66
2.2	Zone de dissémination des événements . . . . .	68
2.3	Economie de la bande passante . . . . .	69
3	Mise en place du protocole de dissémination . . . . .	72
3.1	Données nécessaires . . . . .	73
3.2	File d'attente d'émission . . . . .	74
3.3	Réception d'événements et prise de décision . . . . .	76
4	Le cas des ressources . . . . .	77
4.1	Protocole de réservation . . . . .	78
4.2	Election d'un nouveau coordinateur . . . . .	80
5	Conclusion . . . . .	81
<b>Chapitre 6 :</b>	<b>Prototype, simulation et expérimentations</b>	<b>83</b>
1	Un prototype... . . . .	84
1.1	Architecture matérielle . . . . .	84
1.2	Architecture de VESPA . . . . .	85
1.3	VESPA dans les véhicules . . . . .	87
2	Un simulateur... . . . .	89
2.1	Le choix du simulateur . . . . .	89
2.2	Notre simulateur . . . . .	90
3	Expérimentations . . . . .	91
3.1	Evaluation du mécanisme d'évaluation de la pertinence . . . . .	93
3.2	Evaluation du protocole de dissémination . . . . .	96
3.3	Evaluation du protocole de réservation . . . . .	99
4	Conclusion . . . . .	103

**Chapitre 7 : Conclusion et Perspectives** **105**

---

1	Conclusion . . . . .	105
2	Perspectives . . . . .	107

**Bibliographie** **111**

---

---

# Table des figures

---

<b>Chapitre 3 :</b>	<b>Evaluation de la pertinence</b>	<b>29</b>
3.1	Vecteur mobilité calculé avec un grand intervalle $[t_{n-i}, t_n]$ . . . . .	37
3.2	Vecteur mobilité calculé avec un petit intervalle $[t_{n-i}, t_n]$ . . . . .	37
3.3	Vecteur mobilité calculé avec un intervalle moyen . . . . .	38
3.4	Illustration d'un changement de référentiel . . . . .	38
3.5	Représentation géométrique de $\Delta d$ et $\Delta t$ . . . . .	42
<b>Chapitre 6 :</b>	<b>Prototype, simulation et expérimentations</b>	<b>83</b>
6.1	Terminaux choisis pour le développement de VESPA. . . . .	85
6.2	L'architecture de VESPA. . . . .	86
6.3	VESPA signalant une place de stationnement . . . . .	88
6.4	VESPA signalant un danger sur la route . . . . .	88
6.5	Premier environnement de simulation. . . . .	91
6.6	Premier environnement de simulation. . . . .	92
6.7	Second environnement de simulation. . . . .	92

---

6.8	Pourcentage de véhicules informés. . . . .	94
6.9	Avantage de l'utilisation de vecteurs adaptatifs. . . . .	95
6.10	Evolution des messages disséminés dans un environnement peu dense. . . . .	96
6.11	Evolution des messages disséminés dans un environnement très dense. . . . .	97
6.12	Temps nécessaire pour atteindre la queue de la chaîne de dissémination en environnement peu dense. . . . .	98
6.13	Temps nécessaire pour atteindre la queue de la chaîne de dissémination en environnement très dense. . . . .	98
6.14	Environnement de simulation d'un parking. . . . .	100
6.15	Résultats des expérimentations pour le protocole de réservation. . . . .	102

---

## Liste des tableaux

---

<b>Chapitre 3 :</b>	<b>Evaluation de la pertinence</b>	<b>29</b>
3.1	Représentation des événements . . . . .	33
3.2	Exemple de messages diffusés pour un événement immobile indépendant de la direction . . . . .	34
3.3	Exemple de messages diffusés pour un événement immobile dépendant de la direction . . . . .	35
3.4	Différentes valeurs des coefficients en fonction des configurations testées. . . . .	45

4.1	Exemple de table contenant des événements. . . . .	52
4.2	Résultat de la requête : « Quelles sont les places de parking libres ? » . . . . .	53
4.3	Résultat de la requête : « Quelles sont les places de parking libres que je peux rencontrer ? » . . . . .	54
4.4	Exemple 1 de donnée ajoutée pouvant modifier les résultats de la requête. . . . .	54
4.5	Nouvel ensemble $S_{r_s}$ . . . . .	55
4.6	Résultat mis à jour de la requête : « Quelles sont les places de parking libres que je peux rencontrer ? » . . . . .	55
4.7	Exemple 2 de donnée ajoutée pouvant modifier les résultats de la requête. . . . .	55
4.8	Exemple de donnée supprimée de la base de données. . . . .	55
4.9	Nouvel ensemble $S_{r_s}$ après suppression de la donnée. . . . .	56
4.10	Résultat mis à jour de la requête : « Quelles sont les places de parking libres que je peux rencontrer ? » après suppression de la donnée. . . . .	56
4.11	Ensemble $S_{r_s}$ pour la requête « Quels sont les événements dangereux que je risque de rencontrer sur ma route ? » . . . . .	56
4.12	Résultats initiaux de la requête « Quels sont les événements dangereux que je risque de rencontrer sur ma route ? » . . . . .	56
4.13	Exemple d'une information à mettre à jour . . . . .	57
4.14	Résultats mis à jour de la requête « Quels sont les événements dangereux que je risque de rencontrer sur ma route ? » . . . . .	57
4.15	Résultat de l'évaluation initiale des $\Delta_p$ et $\Delta_e$ de toutes les données. Les lignes grisées représentent les données répondant à la requête ( $\Delta_p > 0$ ). . . . .	59
4.16	Les cellules grisées correspondent aux données sélectionnées pour la réévaluation après un déplacement de 25 mètres. . . . .	60
4.17	Résultat de la requête après la réévaluation suite au déplacement de 25 mètres. . . . .	61
4.18	Les cellules grisées correspondent aux données sélectionnées pour la réévaluation après un déplacement de 25+75 mètres. . . . .	61
4.19	Résultat de la requête après la réévaluation suite au déplacement de 25+75 mètres. . . . .	61

**Chapitre 5 :    Dissémination de l'information** **63**

---

5.1 Représentation des DisseminationInformation. . . . . 73

---

## Liste des algorithmes

---

**Chapitre 5 :    Dissémination de l'information** **63**

---

5.1 runDisseminate() . . . . . 74

5.2 evenementRecu(Evenement e) . . . . . 76

5.3 void declareInterest(Event re, Coordinator c) . . . . . 79

5.4 void confirmlInterest(Event re, Coordinator c) . . . . . 79

5.5 void advertiseParking(Event re) . . . . . 80

5.6 void electNewCoordinator() . . . . . 81



# 1

---

## Introduction

---

Depuis quelques années, l'informatique prend de plus en plus de place dans nos vies et presque tout s'informatise. Dans la dernière décennie, nous avons vu se démocratiser les téléphones portables, les *Pocket PCs*<sup>1</sup>, les assistants personnels et discrètement, l'informatique a envahi notre quotidien. Les évolutions techniques apportent sans cesse de nouvelles perspectives et aujourd'hui, grâce aux réseaux sans fil, une nouvelle dimension s'est développée : la mobilité.

Cette mobilité offre la perspective de nouvelles utilisations de l'informatique. Ainsi, nous avons vu se développer un certain nombre d'applications et de services exploitant cette mobilité pour répondre aux besoins des utilisateurs. Aujourd'hui, ce sont nos véhicules qui sont envahis par l'électronique et l'informatique. Cela n'est pas sans raison. La voiture est le moyen de transport le plus utilisé. Le nombre de véhicules sur nos routes ne cesse de croître. Les problèmes qui pouvaient sembler mineurs il y a une cinquantaine d'années sont devenus critiques. Malheureusement, les embouteillages sont le quotidien d'un grand nombre d'automobilistes, les accidents restent toujours trop nombreux et trop meurtriers et la pollution causée par les véhicules devient une préoccupation mondiale.

---

1. En français : Ordinateurs de poche.

Tous ces problèmes ne sont pas nouveaux mais ils s'intensifient de manière exponentielle avec le temps. Les véhicules qui étaient jusqu'alors presque ignorés par l'informatique deviennent peu à peu un domaine de recherche primordial. En effet, l'informatique est une réponse possible à tous ces problèmes. Depuis des dizaines d'années, la recherche opérationnelle s'intéresse, en partie, aux problèmes de transport de marchandises. En optimisant les trajets des différents transporteurs et en définissant les meilleures tournées de véhicules, elle a permis de limiter le temps nécessaire à des livreurs pour apporter leurs marchandises. L'objectif n'était pas de rendre la route plus confortable ou plus sûre mais de diminuer les coûts liés au transport de marchandises pour optimiser les profits des différentes compagnies. Ces optimisations sont principalement réalisées en amont, c'est-à-dire avant le départ des véhicules. Aujourd'hui, grâce aux systèmes d'information, il est possible de réaliser des optimisations similaires pendant le trajet, non pas pour augmenter un profit mais pour rendre la conduite plus agréable à tous les usagers de la route. Il est donc possible de gérer une flotte de véhicules en temps réel pour optimiser les différents itinéraires. De plus, grâce aux systèmes d'information distribués, nous pouvons envisager d'aider non pas une flotte de véhicules appartenant à une compagnie mais tous les véhicules présents sur la route, y compris les particuliers, permettant ainsi de limiter le nombre d'embouteillages et par la même occasion, de limiter l'impact des véhicules sur l'environnement.

La technologie actuelle nous donne l'opportunité d'offrir à tous les usagers de nos routes des informations permettant d'éviter un certain nombre d'accidents et, de manière plus générale, un grand nombre de désagréments. Grâce aux modules de positionnement et aux modules de communication sans fil, des terminaux peuvent être installés dans les différents véhicules pour assister tous les conducteurs.

## 1 Motivations

Le but de cette thèse est de proposer aux automobilistes un nouveau moyen d'obtenir des informations sur la route. Pour l'instant, les automobilistes ne possèdent que leurs yeux et leurs oreilles pour analyser l'environnement routier. L'analyse de la situation est donc limitée puisque les informations disponibles ne concernent que l'entourage très proche de chacun des conducteurs et les prises de décisions ne sont donc pas toujours adaptées. Dans le cas d'un accident, les automobilistes ne prennent en compte cette information que lorsqu'ils arrivent sur le lieu de l'accident. Dans le cas d'un embouteillage, l'information remonte aussi loin que les automobilistes activent leurs feux de détresse. Un automobiliste n'est donc prévenu de la présence d'un embouteillage que lorsqu'il arrive à la queue de celui-ci. Pour une ambulance, l'activation des sirènes permet de prévenir tous les véhicules dans un certain rayon. Les automobilistes n'ont cette information que lorsque l'ambulance est déjà proche. Pour les véhicules devant laisser l'ambulance se frayer un chemin, il est déjà tard. Pour d'autres véhicules, l'information ne doit pas être prise en compte si l'ambulance circule dans une rue proche mais différente ou dans l'autre sens de circulation. L'objectif de cette thèse est donc de présenter une solution permettant aux automobilistes d'accéder à des informations pertinentes les aidant à conduire mieux.

## 2 Notre contexte du partage d'informations

Nos recherches se placent dans le contexte des réseaux de communication inter-véhiculaire. En effet, il s'agit pour nous de proposer une solution pour le partage d'informations entre les différents véhicules. La technologie utilisée actuellement pour permettre un échange d'informations entre les automobilistes est obsolète. Mis à part l'avertisseur sonore et les différents signaux lumineux, les automobilistes n'ont aucun moyen de communiquer entre eux. Notre partage d'informations repose sur l'utilisation des nouvelles technologies. Cependant, nous sommes liés au contexte routier. Equiper les véhicules avec des terminaux communicants représente le point de départ. Il s'agit pour nous de prendre en compte notre environnement pour permettre un partage optimal des informations en exploitant au mieux la technologie qui nous est accessible aujourd'hui. Avant de nous intéresser aux contraintes de l'environnement, nous posons quelques définitions qu'il nous semble important d'explicitier.

### 2.1 Définitions

Avant de commencer à expliquer nos contraintes et nos objectifs, il convient de bien définir ce que nous voulons proposer, c'est-à-dire le partage d'informations dans les réseaux de communication inter-véhiculaire.

Nous devons donc dans un premier temps bien comprendre ce qu'est le partage d'informations. Il s'agit de la mise à disposition d'une information par un utilisateur pour en faire bénéficier d'autres. Le partage d'informations requiert donc un certain nombre d'utilisateurs. Un utilisateur met à disposition une information. Les autres utilisateurs peuvent accéder à cette information et l'exploiter. Par exemple, à l'heure actuelle, lorsqu'un automobiliste active ses feux de détresse, il met à disposition une information concernant un danger. Les automobilistes qui voient cette information peuvent alors l'exploiter et agir de manière adaptée, en ralentissant par exemple. Malheureusement, pour l'instant, l'échange d'informations est très limité puisque d'une part, les données qu'il est possible de mettre à disposition ne sont pas nombreuses, et d'autre part, l'information n'est accessible et exploitable que très tard puisque les véhicules doivent déjà être très proches. Il est donc nécessaire de trouver un nouveau moyen d'échanger des informations. Pour cela, nous proposons d'utiliser un réseau de communication.

Un réseau de communication, c'est un réseau par lequel peuvent transiter des informations. C'est le support permettant l'échange d'informations. Actuellement, dans nos voitures, la communication s'opère par la vue ou par l'ouïe. Grâce à l'informatique, nous avons accès à de nouveaux moyens de communication. Les réseaux sans fil permettent à des terminaux en situation de mobilité d'échanger des informations et sans nécessiter de manipulation particulière. Aucune action de l'utilisateur n'est requise avec les réseaux sans fil, contrairement aux réseaux filaires qui nécessitent la mise en place d'un câble permettant la communication. La seule contrainte est la proximité. Deux terminaux peuvent communiquer s'ils sont assez proches. Notre réseau de communication est donc formé

par les véhicules suffisamment proches et par le moyen de communication sans fil. Les véhicules peuvent donc échanger des informations. De plus, une information mise à disposition sur le réseau de communication peut être exploitée et remise à disposition pour atteindre de nouveaux véhicules. Ce principe est déjà utilisé pour les embouteillages. Un automobiliste active ses feux de détresse et les automobilistes voyant l'information l'exploitent et activent à leur tour leurs feux de détresse pour atteindre de nouveaux véhicules.

Finalement, le partage d'informations dans un réseau de communication inter-véhiculaire, c'est la mise à disposition d'informations par les véhicules sur le réseau sans fil considéré pour permettre l'exploitation de ces informations par les autres véhicules. Il s'agit donc pour nous de proposer une solution permettant à des terminaux embarqués dans les véhicules de communiquer entre eux pour échanger des données.

### 2.2 Contraintes de l'environnement

Puisque nous nous plaçons dans le contexte routier, plusieurs éléments sont à prendre en considération dans notre solution. Tout d'abord, le réseau routier est un réseau fortement contraint où les automobilistes ont un champ d'action limité. Par exemple, il est impossible de faire un demi-tour n'importe où et n'importe quand pour éviter un embouteillage. Les véhicules suivent donc des chemins très contraints qu'il est parfois difficile de modifier. Ensuite, les véhicules se déplacent potentiellement à grande vitesse. Ainsi, entre deux instants, la topologie du réseau est différente. Des véhicules proches à un instant donné peuvent ne plus l'être l'instant d'après. Cela signifie donc que des véhicules qui peuvent communiquer à un instant donné ne le peuvent peut-être plus l'instant suivant. Les messages doivent donc être échangés dans un court laps de temps. Enfin, les véhicules ne se déplacent pas sans but. Chaque véhicule possède un point de départ et une destination. Pourtant, pour nous, cette destination est inconnue et l'itinéraire que le véhicule va emprunter nous est aussi inconnu. Bien qu'il soit possible de connaître l'itinéraire des véhicules, nous ne voulons pas proposer un système reposant sur ces informations. En effet, pour avoir ces informations, il est nécessaire dans un premier temps d'utiliser des cartes. Dans un second temps, cela impose que l'utilisateur fournisse au moins l'information sur sa destination. Or, pour nous, il ne s'agit pas de fournir une solution qui repose sur trop de pré-requis. L'un des défis est donc pour nous de ne pas reposer sur l'utilisation de données cartographiques ni même sur la collecte d'informations préalables auprès des conducteurs. Les principales caractéristiques de notre environnement sont donc que les véhicules suivent des routes inconnues à l'avance, que les automobilistes peuvent modifier leur itinéraire à condition d'avoir les informations suffisamment tôt pour pouvoir prévoir ces changements de trajet, et que les véhicules ne peuvent communiquer que pendant un temps très limité.

Toutes ces contraintes nous confrontent à un certain nombre de problèmes auxquels nous devons faire face pour proposer la solution la plus efficace possible.

#### 2.3 Différents problèmes à résoudre

Pour fournir le système que nous voulons, de nombreux problèmes sont à résoudre. Par rapport à la solution que nous voulons proposer, nous devons d'abord identifier quelles informations doivent être partagées. Dans l'environnement routier, il existe une quantité très importante d'informations. Les accidents, les embouteillages, les travaux, les freinages d'urgence, les conducteurs « dangereux », les places de stationnement disponibles, la présence de radar de police, les véhicules d'intervention d'urgence, les convois exceptionnels ne représentent qu'un ensemble limité d'informations qu'il est possible d'échanger. Il en existe bien d'autres. Pourtant, toutes les informations ne doivent pas être partagées. Seules certaines sont intéressantes et peuvent être exploitées. La première difficulté est donc d'identifier toutes ces informations.

Ensuite, nous devons déterminer la qualité d'une information. Nous devons être capable de différencier les données intéressantes et les données inintéressantes pour un véhicule. C'est ce que nous appelons la pertinence d'une information. Si les véhicules partagent des informations, elles ne sont pertinentes que pour un certain nombre d'entre eux. Il est donc très important pour chaque véhicule de savoir déterminer quelles sont les informations exploitables.

Un autre aspect à considérer est que l'utilisateur est un conducteur. Il ne s'agit donc pas de lui fournir un trop grand nombre d'informations. En conduisant, il ne peut en assimiler qu'un nombre limité. Nous devons trouver un moyen de répondre aux attentes spécifiques de chaque conducteur pour ne lui fournir que celles qui répondent à un besoin.

Les précédents problèmes en impliquent un autre. Le nombre d'informations que notre système doit traiter risque d'être très important. Il est donc nécessaire de proposer un mécanisme permettant à nos terminaux de traiter rapidement et efficacement les données. Notre système doit fournir aux conducteurs les informations le plus rapidement possible.

Enfin, nous devons proposer un mécanisme permettant l'échange d'informations en utilisant notre support de communication. La question est de savoir comment mettre à disposition une information pour les autres automobilistes. L'utilisation de réseaux sans fil est la base de l'échange mais nous devons proposer une solution adaptée à notre contexte pour que chaque véhicule puisse accéder aux informations dont il a besoin. Les temps de communication étant très courts, cet échange doit se faire aussi rapidement que possible.

En ayant connaissance de tous ces problèmes, cette thèse a pour but de remplir certains objectifs.

### 3 Objectifs de la thèse

L'objectif final de cette thèse est donc de proposer une solution permettant le partage d'informations dans un réseau de communication inter-véhiculaire. Nous voulons offrir aux conducteurs

un nouveau moyen de communication leur permettant de conduire mieux. Grâce à un échange de données adapté, nous voulons permettre à chacun d'agir au mieux pour lui-même et pour les autres. Cette thèse propose donc trois contributions majeures :

1. **Un mécanisme d'évaluation de la pertinence** : Ce mécanisme permet de déterminer quelles informations doivent être prises en compte par le véhicule. Il s'agit donc de déterminer quelles sont les informations qui peuvent intéresser le conducteur.
2. **Un mécanisme d'évaluation continue des informations reçues** : Tout au long de son déplacement, le véhicule va glaner des informations qui peuvent répondre à des besoins de l'automobiliste. De plus, puisque le véhicule se déplace, certaines informations peuvent être intéressantes à un moment donné puis ne plus l'être. Ce mécanisme sert donc à contrôler en permanence quelles informations doivent être présentées au conducteur pour répondre à ses besoins.
3. **Un protocole de dissémination des informations** : Il s'agit ici de définir comment un véhicule met à disposition une information pour les autres véhicules et comment une information est rendue exploitable par tout un ensemble de véhicules.

## 4 Plan de ce mémoire

Après avoir décrit les solutions existantes dans le chapitre 2 (page 7), nous exposerons nos contributions. Nous commencerons d'abord par détailler notre mécanisme d'évaluation de la pertinence dans le chapitre 3 (page 29). Nous verrons ensuite dans le chapitre 4 (page 47) comment nous évaluons de manière continue les différentes requêtes posées à notre système. Notre troisième contribution fera l'objet du chapitre 5 (page 63). Nous y expliciterons notre protocole de dissémination permettant à chaque véhicule d'avoir accès aux informations intéressantes pour lui. Le chapitre 6 (page 83) présentera le prototype et le simulateur que nous avons développés, ainsi que les résultats expérimentaux que nous avons obtenus. Finalement, nous terminerons ce mémoire par nos conclusions et nos perspectives dans le chapitre 7 (page 105).

# 2

---

## Un domaine de recherche récent

---

Le *pair-à-pair mobile*, cadre dans lequel s'inscrit la *communication inter-véhiculaire*, est un domaine de recherche récent. Depuis quelques années, les *systèmes de positionnement* permettent de proposer de nouveaux services aux utilisateurs en exploitant leur localisation géographique.

Les nouveaux services proposés aujourd'hui s'adressent principalement à des utilisateurs faiblement mobiles. La mise en œuvre de nouveaux services avec des contraintes de mobilité plus fortes, pour les automobilistes par exemple, représente un challenge très important. Les premiers systèmes d'information distribués s'appuyaient sur une connaissance de la topologie du réseau. En effet, les nœuds étaient immobiles et cela rendait possible l'utilisation de protocoles de routage exploitant la connaissance de la topologie pour définir la meilleure manière d'acheminer les informations. Avec l'apparition des terminaux mobiles, les systèmes d'information ont du prendre en compte la dynamique de la topologie. Les MANETs – *Mobile Ad hoc NETWORKS* – sont ces réseaux formés par un ensemble de nœuds mobiles. Ici, le routage des informations est plus compliqué puisqu'aucun chemin ne peut être déterminé avec certitude. Du fait de la mobilité des terminaux et des possibles déconnexions des terminaux, les chemins valides à un instant  $t$  ne le sont peut-être plus à l'instant  $t + 1$ .

En considérant une mobilité relativement faible, celle des piétons par exemple, il est possible de prendre en considération les déplacements des terminaux pour maintenir une connaissance suffisante de la topologie et ainsi, garantir la bonne livraison des informations. Aujourd'hui, avec les réseaux inter-véhiculaires, nous sommes dans les VANETs – *Vehicular Ad hoc NETworks* – qui sont donc une sous-classe des MANETs. Cette distinction peut sembler superflue et pourtant, elle est nécessaire, tant les challenges des VANETs sont importants. En effet, la forte mobilité des véhicules rend inutilisable les solutions proposées pour les MANETs, tant la topologie varie rapidement au cours du temps. Pourtant, les ITS – *Intelligent Transportation Systems* – sont en pleine expansion [FJM<sup>+</sup>01, YKLH07] et de nombreuses contributions ont été proposées afin de proposer de nouveaux systèmes d'assistance aux conducteurs. Dans [LpH04, LH06, BEH04], les auteurs proposent un aperçu de différentes directions de recherche.

La suite de ce chapitre est organisée de la manière suivante. Dans la section 1 (page 8), nous présentons *les différentes architectures* qu'il est possible d'utiliser pour mettre en place les nouveaux services pour les conducteurs et discutons les limites des solutions existantes. Dans la section 2 (page 13), *les différentes familles de protocoles* utilisées dans la communication inter-véhiculaire sont présentées. Nous nous concentrons dans la section 3 (page 19) sur *les techniques de gestion des données* pour finalement terminer sur *des travaux spécifiques* qui traitent d'aspects très intéressants dans la section 4 (page 24). Enfin, nous positionnons nos travaux de recherche par rapport aux travaux exposés dans ce chapitre dans la section 5 (page 26).

### 1 Les différentes architectures

Pour fournir de nouveaux services aux automobilistes, l'architecture utilisée pour construire l'application est très importante. Les systèmes d'assistance aux conducteurs les plus populaires aujourd'hui reposent sur *un appareil autonome sans aucun moyen de communication*. Il s'agit des *systèmes de navigation* qui utilisent une *base de données cartographiques* embarquée dans un terminal mobile. Cette base de données est ensuite exploitée pour fournir à l'utilisateur un itinéraire ou l'emplacement de POIs – *Points Of Interest* – tels que des stations services, des parkings, etc. Par contre, pour fournir de nouvelles informations aux conducteurs, il est nécessaire de mettre à jour la base de données embarquée, ce qui représente la principale contrainte de ces systèmes. Dans un monde où les données sont de plus en plus nombreuses et de plus en plus dynamiques, ce genre de système n'est plus suffisant. Il est nécessaire de fournir aux automobilistes *des données mises à jour en permanence*.

La technologie actuelle nous offre de nombreux moyens pour communiquer, en particulier des moyens de *communication sans fil*. Il est donc possible d'équiper nos voitures avec de tels terminaux pour accéder, pendant notre trajet, à des sources d'informations très complètes. Les différentes technologies existantes permettent d'envisager différentes manières d'accéder à l'information.

L'utilisation des *réseaux téléphoniques* permet d'interroger des sources de données distantes. Ces sources de données sont généralement des serveurs stockant des volumes de données beaucoup

plus importants que les bases de données embarquées des systèmes de navigation. De plus, ces sources peuvent être maintenues à jour de sorte que les véhicules qui les interrogent reçoivent toujours les informations les plus récentes. Pour les piétons, les applications comme *AroundMe* ou *Locly* disponibles sur l'iPhone utilisent les informations de localisation des terminaux pour interroger des serveurs distants. Les utilisateurs peuvent ainsi recevoir la liste des restaurants, des hôtels ou même des hôpitaux proches d'eux. Pour les automobilistes, certaines applications comme *Coyote*, *Eklaireur* ou *Wikango* utilisent le même principe pour fournir des alertes aux conducteurs. L'utilisation des réseaux de téléphonie mobile permet ainsi aux conducteurs d'être avertis de la présence de radars de police mobiles, d'incidents, d'embouteillages, de travaux ou encore d'accidents sur la route. L'application *Parking Dispo* fournit quant à elle des informations sur les parkings Parisien en s'appuyant sur les mêmes réseaux.

Une autre possibilité pour fournir de l'information aux automobilistes est d'utiliser les ondes radios de la bande FM. Le protocole TMC – *Traffic Message Channel* – définit notamment une liste de 1460 événements. Certaines applications utilisent ces données pour fournir de l'information sur le trafic en temps réel et ainsi, optimiser les itinéraires proposés aux conducteurs.

Finalement, la dernière possibilité est d'utiliser les moyens de communication sans fil de courte portée tel que le *protocole IEEE 802.11*. Avec ces technologies, un automobiliste ne peut pas communiquer directement avec un serveur distant mais il peut communiquer avec tous les appareils proches de lui. Il s'agit d'une communication de proximité où les terminaux peuvent échanger des messages s'ils sont suffisamment rapprochés.

Dans la suite de cette section, nous présentons *trois différents type d'architectures* [JMS07]. Dans la section 1.1 (page 9), nous considérons les architectures reposant sur une infrastructure. Ensuite, dans la section 1.2 (page 10), nous expliquons qu'il est possible d'avoir une approche totalement décentralisée sans aucun besoin d'infrastructure. Finalement, dans la section 1.3 (page 12), nous nous intéressons aux architectures hybrides qui exploitent les avantages des deux approches.

### 1.1 La robustesse d'une infrastructure

Les premières architectures proposées reposent sur un modèle où *une infrastructure* fournit toutes les données. Par exemple, dans les systèmes de navigation récents, un récepteur FM reçoit les informations TMC puis elles sont interprétées pour fournir le meilleur itinéraire. Au Japon, le système VICS – *Vehicle Information and Communication System* – utilise différents types de transmetteurs pour fournir des informations sur la route. Dans ces deux architectures, les véhicules ne communiquent pas avec l'infrastructure, c'est l'infrastructure qui envoie des informations. Les véhicules ne font qu'interpréter les informations pour fournir aux conducteurs les données dont il a besoin.

Le plus grand inconvénient de ces infrastructures est le *temps nécessaire à la mise à jour des données*. En effet, puisque les véhicules (ou leur conducteur) ne sont pas capables ici de fournir

eux-mêmes des informations. Les différents événements doivent donc être constatés, soit par les autorités locales, soit par les compagnies spécialisées, pour ensuite être ajoutés et diffusés dans le système. Une information n'est donc disponible qu'après un certain temps. Ces architectures ne peuvent donc pas servir pour véhiculer *les informations très dynamiques*. Par contre, l'avantage de ces architectures est *la robustesse*. Les données diffusées sont certifiées et ne peuvent pas être corrompues.

Pour permettre une plus grande réactivité, des architectures permettant aux véhicules d'envoyer des informations ont été proposées. La première utilise *les réseaux de téléphonie mobile*. Les véhicules peuvent ainsi accéder à des serveurs distants mais aussi envoyer des informations sur ces serveurs. Lorsqu'un véhicule constate un événement, il peut ainsi ajouter l'information dans la base de données distante. De cette manière, tous les véhicules peuvent accéder à cette information.

Une autre architecture utilise des points d'accès disséminés le long des routes. En utilisant les moyens de communication de courte portée, les véhicules peuvent communiquer avec les points d'accès lorsqu'ils passent à proximité de ces derniers. Ces points d'accès permettent d'accéder à des informations stockées sur des serveurs distants ou d'envoyer des données sur ces serveurs. Dans [GOM08], l'architecture utilisée est une architecture trois tiers. Un serveur central est connecté à des bornes disséminées le long des routes. Ces bornes communiquent avec les terminaux mobiles des usagers. Cela est possible ici car la zone de déploiement est relativement petite. Il s'agit d'un campus universitaire. Par contre, le déploiement d'une telle architecture, avec des bornes interconnectées à des serveurs centraux, n'est pas envisageable sur l'ensemble de nos routes.

Avec de telles solutions, la mise à disposition d'une information est plus facile que pour les architectures où les véhicules ne peuvent pas communiquer avec l'infrastructure. Cela permet également de gérer des données plus dynamiques. Cependant, ici encore, le temps entre la mise à disposition d'une donnée et son exploitation est toujours trop important pour pouvoir gérer des informations très éphémères telles qu'un freinage d'urgence par exemple. Le fait d'utiliser un intermédiaire induit des temps trop longs pour certaines applications. De plus, le coût nécessaire au déploiement d'une infrastructure est très important et ne peut être envisagé à court ou moyen terme à grande échelle. Quant aux solutions s'appuyant sur une infrastructure existante, les réseaux de téléphonie mobile par exemple, elles représentent un coût important pour les usagers et ne permettent pas de régler le problème de lenteur de mise à disposition. C'est pourquoi, des architectures plus dynamiques ont été imaginées.

### 1.2 La coopération entre véhicules

Certains projets s'appuient exclusivement sur la communication entre les véhicules pour former *des réseaux ad hoc véhiculaires*. Dans le projet CarTalk2000 [MCGM02, dBKvKN04], les communications entre véhicules ont été utilisées pour gérer trois situations pouvant survenir sur les

routes. La première est la diffusion d'alertes concernant un danger, par exemple lorsqu'un accident survient. Les véhicules accidentés peuvent ainsi diffuser des messages signalant leur position. Cela permet de prévenir les véhicules proches. Grâce à des techniques de *multi-hop*<sup>1</sup>, l'information peut être relayée assez loin, permettant d'éviter des sur-accidents. La deuxième situation gérée est la situation où la visibilité d'un automobiliste est réduite, à cause de la présence d'un camion juste devant lui par exemple. Dans cette situation, les solutions proposées dans le projet CarTalk2000 permettent de diffuser des informations permettant aux conducteurs de « voir à travers ». Enfin, la dernière situation traitée est la fusion de files. En particulier l'insertion de véhicules sur autoroute et la sortie d'autoroute. Là encore, les moyens de communication sans fil et les réseaux ad hoc sont utilisés pour permettre à chaque véhicule de communiquer et ainsi, de coopérer pour éviter les situations dangereuses. Pour former les réseaux ad hoc, l'utilisation des *standards UMTS* est préconisée en adaptant la technologie *UTRA-TDD* pour permettre l'utilisation en réseaux ad hoc. Le projet se focalise plus sur le développement technologique que sur le développement de services. Cependant, l'utilisation de cette architecture est très intéressante puisqu'elle ne requiert pas un coût trop important et qu'elle permet une très grande réactivité. Puisque les véhicules communiquent sans passer par un intermédiaire, l'échange de messages est en effet beaucoup plus rapide. La diffusion des informations ne concerne toutefois ici qu'une zone géographique réduite. Aucun message ne doit parcourir plus que quelques centaines de mètres. Il ne s'agit donc pas de construire un réseau inter-véhiculaire incluant tous les véhicules mais plutôt de créer de petites sphères de communication bien spécifiques à des scénarios prédéfinis.

D'autres solutions tentent d'utiliser la communication entre les véhicules pour créer des réseaux à l'échelle du réseau routier [WER<sup>+</sup>03]. En particulier dans [GION03], les auteurs présentent une architecture totalement décentralisée : *Grassroots*. Dans cette architecture, chaque nœud doit savoir réaliser trois opérations : interroger, surveiller et disséminer les informations. Les auteurs introduisent quatre types d'entités. Tout d'abord les *producteurs* d'informations sont les véhicules qui génèrent, grâce à divers capteurs, des informations. Les *consommateurs* d'informations exploitent les informations produites. Des *agrégateurs* collectent les informations produites et les agrègent pour résumer les données transmises sur le réseau afin d'économiser la bande passante. Enfin, la mission des *relayers* est de transmettre les messages pour qu'ils atteignent leur destination le plus rapidement possible. Chaque nœud peut prendre un de ces rôles à n'importe quel moment, en fonction du besoin du moment et de l'endroit où il se trouve. Ici, l'utilisation de la communication entre les véhicules n'est pas forcément limitée à une zone géographique très restreinte. C'est en fonction de l'information que la zone géographique d'intérêt est déterminée. Les messages peuvent parcourir plusieurs kilomètres pour atteindre leur destination et les messages diffusés peuvent être exploités par un nombre plus ou moins important de véhicules.

L'utilisation exclusive de communication entre les véhicules présente quelques inconvénients qui nécessitent de trouver des solutions. Tout d'abord, coordonner tous les véhicules pour qu'ils partagent les bonnes informations aux bons endroits n'est pas aisé. Il est de surcroît nécessaire

---

1. En français : Multi-sauts. Il s'agit d'utiliser chaque terminal comme relai pour faire « sauter » les données de terminal en terminal.

d'avoir un taux de véhicules équipés suffisant. En effet, si peu de véhicules sont équipés, la création des réseaux ad hoc est très difficile, voire impossible. C'est pourquoi, des architectures associant communication inter-véhiculaire et infrastructure ont été proposées.

### 1.3 Une architecture hybride

Une architecture qui est de plus en plus souvent utilisée est l'architecture hybride. C'est-à-dire une architecture où une infrastructure est déployée mais où la communication entre les véhicules est exploitée en parallèle. Ici, l'infrastructure peut être utilisée de deux manières différentes. Tout d'abord, comme passerelle vers des serveurs distants, il s'agit alors d'une infrastructure assez lourde interconnectée. La communication inter-véhiculaire peut alors être utilisée pour permettre à des véhicules éloignés de communiquer avec l'infrastructure, en utilisant d'autres véhicules comme relais. Le projet *FleetNet* [FFH<sup>+</sup>04, FHB01, BFW03, ERW<sup>+</sup>03, WER04, HFMF03] – *Internet on the Road* – utilisait une telle architecture. L'infrastructure servait avant tout à offrir un accès à *Internet* [BFW03]. Avec leur système, les véhicules proches des bornes peuvent donc accéder à des services distants présents sur *Internet*. L'objectif ici est plutôt d'offrir des services aux passagers qui pourront, par exemple, jouer en ligne. Par contre, l'infrastructure ne peut pas être implantée suffisamment massivement pour offrir un tel accès à chaque instant pour tous les véhicules. La communication entre les véhicules doit permettre à chacun de communiquer avec ces bornes en utilisant les véhicules voisins comme relais. Pour cela, ils utilisent une version modifiée de l'UTRA-TDD supportant les réseaux ad hoc [ERW<sup>+</sup>03]. En utilisant des techniques de multi-hop, chaque véhicule doit pouvoir accéder à *Internet*. Ensuite, la communication directe entre les véhicules permet d'envisager des services liés à la sécurité. Par exemple, lors d'un freinage d'urgence, le véhicule diffuse l'information pour prévenir les véhicules proches.

L'infrastructure peut aussi être moins lourde s'il ne s'agit pas de bornes interconnectées. Il s'agit alors plutôt d'une infrastructure qui aide la communication entre les véhicules en stockant les informations et en les disséminant. Dans [LCM09], les auteurs proposent un mécanisme de dissémination d'informations dans un tel environnement. Des bornes sont disséminées pour stocker et diffuser les informations aux véhicules proches. Pour atteindre des véhicules qui ne sont pas à proximité d'une borne, les véhicules communiquent entre eux pour permettre à chacun d'obtenir des informations de ces bornes. Les solutions utilisant l'infrastructure comme aide aux services sont les plus à même d'être utilisées puisqu'elles proposent un fonctionnement avec une infrastructure limitée.

L'utilisation d'une infrastructure permet d'assister la communication entre les véhicules et de limiter l'impact du taux d'équipement sur les systèmes. Cela est particulièrement intéressant au lancement de ces systèmes puisque c'est dans cette période que le taux d'équipement risque ne pas être suffisant. Par contre, le coût lié à l'implantation d'une infrastructure est très important et un tel déploiement n'est pas envisageable, notamment si l'on considère une infrastructure interconnectée. Avoir une infrastructure partielle qui ne soit pas interconnectée est le plus vraisemblable dans un

avenir proche. C'est aussi la vision adoptée dans [WXY04] où des points d'accès génèrent, stockent et diffusent des informations.

Pour nous, l'objectif est de proposer une solution ne s'appuyant au départ sur aucune architecture, de sorte que notre système puisse être utilisé sans aucune installation coûteuse, longue et impossible dans certains environnements. Par contre, cela ne signifie pas qu'une infrastructure n'est pas envisageable. Notre système ne doit donc pas être incompatible avec une infrastructure qui stockerait et aiderait à disséminer les informations. Selon nous, il est très important de ne pas différencier le rôle des véhicules et des bornes disséminées le long des routes. Les bornes doivent être considérées comme des véhicules immobiles.

Bien que l'architecture choisie décrive déjà partiellement le principe du système qui l'utilise, il est aussi très important de savoir comment les informations vont transiter à travers cette architecture. C'est pourquoi, dans la section suivante, nous étudions les différents protocoles de communication existants.

## 2 Protocoles de communication

Quelque soit l'architecture choisie, il est nécessaire de spécifier comment les informations vont transiter entre les différentes entités. En effet, il est important de savoir comment les différents véhicules vont communiquer entre eux et/ou avec l'infrastructure. Dans le cadre des protocoles de communication, de nombreuses possibilités s'offrent à nous. Le choix dépend principalement de l'objectif poursuivi. Quoiqu'il en soit, le contexte des VANETs offre des challenges très intéressants dûs à la forte mobilité des véhicules. En 1997, dans [NI97], les auteurs identifiaient déjà le besoin de localisation pour les futurs services. Ils proposaient alors un adressage et un routage géographique pour prendre en compte la localisation des terminaux. Ils devinaient l'émergence des nouveaux services liés à la localisation que nous connaissons aujourd'hui.

Aujourd'hui, les protocoles de routage ne prennent pas seulement en compte la localisation mais aussi la mobilité. Dans [LCL10], les auteurs offrent un aperçu des différents protocoles de communication dans les VANETs. Ils fournissent une taxonomie des réseaux ad hoc véhiculaires où il y a principalement trois familles de protocoles. La première famille concerne les *protocoles de routage unicast*. Il s'agit des protocoles dont l'objectif est de transmettre une donnée d'une source connue à une destination connue. La deuxième famille inclut les *protocoles multicast et geocast*. Il s'agit de protocoles dont l'objectif est de diffuser une information à tous les véhicules présents dans une certaine zone géographique spécifique. Enfin, la dernière famille est représentée par les *protocoles de broadcast* dont l'objectif est principalement de diffuser des messages d'urgences à tous les véhicules alentours. Bien que les applications soient différentes, les différents protocoles de communication ont tous un réel intérêt.

Dans la suite, nous nous intéressons dans la section 2.1 (page 14) aux protocoles de routage proprement dit. C'est-à-dire les protocoles dont le but est d'envoyer un message d'une source

connue à une destination connue. Nous verrons ensuite dans la section 2.2 (page 15) les protocoles geocast dont l'objectif est d'atteindre une zone géographique spécifique. Finalement, dans la section 2.3 (page 17), nous nous intéressons aux protocoles cherchant à diffuser l'information dans une zone géographique.

### 2.1 Protocoles de routage

Les protocoles de routage ont pour objectif de transmettre un message d'un expéditeur à un unique destinataire. Dans le contexte des MANETs, certaines solutions sont très intéressantes. Dans [KWZ02], les auteurs proposent un protocole de routage pour les MANETs. Leur protocole ne requiert que les informations sur les voisins directs. C'est-à-dire que chaque nœud n'a besoin de connaître que la position de ses proches voisins. En limitant la connaissance nécessaire au fonctionnement du protocole, ils permettent d'une part d'économiser les échanges d'informations relatifs à la topologie du réseau et d'autre part d'être moins sensibles aux variations de topologie. Cela permet à leur protocole d'envoyer un message jusqu'à une localisation avec des performances très intéressantes.

Malheureusement, les solutions actuelles fonctionnant pour les MANETs ne fonctionnent plus dans les VANETs tant la topologie varie fortement et de manière continue. Dans [WQQ08], les auteurs s'intéressent aux protocoles de routage comme AODV – *Ad hoc On demand Distance Vector*, DSDV – *Destination Sequenced Distance Vector* – ou DSR – *Dynamic Source Routing* – qui sont bien adaptés aux caractéristiques des MANETs. Ces protocoles s'appuient sur la construction d'une route dans le réseau et requièrent donc une certaine connaissance de la topologie. De ce fait, les performances se dégradent dès lors que l'on considère une mobilité plus forte. C'est pourquoi, dans les VANETs, ces protocoles sont inutilisables.

Une étude de différentes *stratégies de routage dans les réseaux ad hoc mobiles* est présentée dans [MWH01]. Les auteurs y séparent les solutions en fonction de la connaissance du graphe de chaque nœud. En effet, plus la topologie du réseau est connue par un grand nombre de nœuds, plus il est facile de trouver un chemin. Si, dans un environnement immobile, il est assez facile de permettre à chacun d'avoir connaissance de la topologie du réseau, ce n'est plus du tout le cas dans un environnement mobile. Pour maintenir un niveau de connaissance suffisant, un certain nombre de messages doivent être échangés. Dans un environnement comme les VANETs, le voisinage ainsi observé varie de plus très rapidement et ces solutions deviennent inutilisables. Il est alors nécessaire de trouver des solutions ne nécessitant aucune connaissance de la topologie. De cette manière, les coûts liés au maintien de cette connaissance sont inexistantes. Par contre, il devient extrêmement difficile de trouver un chemin valide à un instant donné. Trouver le meilleur chemin le plus rapidement possible représente donc le challenge le plus important.

Dans [FMH<sup>+</sup>03], les auteurs montrent que le routage basé sur la position peut fonctionner sur autoroute mais en adaptant les mécanismes actuels qui s'appuient sur la topologie, élément trop

dynamique dans les VANETs. Dans [FMH<sup>+</sup>02], les auteurs font une comparaison de différentes stratégies de routage dans les réseaux ad hoc véhiculaires. En particulier, ils comparent une stratégie basée sur le maintien d'un chemin entre les nœuds (DSR) avec des stratégies ne s'appuyant que sur la position des véhicules. Leur étude montre que les stratégies basées exclusivement sur la position du véhicule sont assez performantes mais qu'elles doivent encore être améliorées pour faire du routage efficace. Ils envisagent donc un routage basé non seulement sur la position mais aussi sur la vitesse et la direction des véhicules.

Puisque la topologie est un élément sur lequel les protocoles de routage ne peuvent pas s'appuyer dans les VANETs, certaines solutions tentent de reposer sur d'autres données. Par exemple, dans [LHT<sup>+</sup>03] ou dans [THRC03] les auteurs utilisent des cartes pour faire du routage dans les villes. En effet, puisque le réseau est composé des véhicules, il y a une certaine analogie entre la topologie du réseau de communication et la topologie du réseau routier. Autrement dit, savoir où il y a des véhicules est une indication intéressante pour savoir où il y a des terminaux. Ainsi, en s'appuyant sur des données cartographiques embarquées et incluant les flux de véhicules sur les différents tronçons de route, il est possible de déterminer, sans communication et sans trop de difficulté, où il peut y avoir des véhicules pour relayer une information. Par contre, même si ces informations donnent une indication, elles ne reflètent pas la réalité exacte à un instant  $t$ . Malgré cela, leur protocole permet de déterminer un chemin ayant une forte probabilité d'exister.

Dans le cadre des protocoles de routage, la question essentielle est de déterminer un chemin entre deux nœuds. Toutes les solutions reposent sur une connaissance plus ou moins importante de la topologie. Dans les réseaux immobiles, la topologie est connue à priori. Dans les réseaux faiblement mobiles, la connaissance de la topologie est maintenue en diffusant certains messages. Dans les réseaux fortement mobiles, la connaissance de la topologie est obtenue en analysant des données externes. Pour les réseaux de communication véhiculaire, des données cartographiques permettent de « deviner » où se trouvent les nœuds. Quelque soit la méthode utilisée, cela permet de transmettre un message à un destinataire unique. Cette famille de protocole est donc très utile lorsque l'on veut faire de la communication spécifique. Par exemple, lorsque l'on veut interroger un nœud particulier. Dans le cadre des services pour les conducteurs, il peut s'agir, par exemple, d'interroger un service fourni par une station service permettant de connaître ses tarifs. Cependant, il peut être intéressant d'interroger ou de diffuser une information sur tout un ensemble de nœuds dans une zone géographique particulière. Dans ce cas, l'utilisation des protocoles géocast est plus adaptée.

### 2.2 Protocoles géocast

Les protocoles géocast ont pour but de disséminer un message à un ensemble de nœuds se trouvant dans *une zone géographique particulière*. Ici, l'expéditeur d'un message est un nœud et le destinataire est une région. La difficulté avec de tels protocoles est de réussir à atteindre le plus rapidement la région visée tout en minimisant les coûts liés à la communication. Un aperçu de

plusieurs protocoles géocasts est proposé dans [Mai04]. Dans cet article, l'auteur décrit clairement quelques protocoles. L'objectif pour nous ici n'est pas de présenter tous les protocoles mais de voir comment ils se comportent et comment ils peuvent être utilisés.

Dans [KV99], les auteurs proposent un protocole utilisant une *zone de transmission*. Il s'agit de la zone où le message est transmis et permettant d'atteindre *la zone cible*. Cette zone de transmission doit être suffisamment importante pour permettre d'atteindre la zone cible. En effet, il doit exister un chemin entre l'expéditeur et la zone cible dans la zone de transmission. Comme les nœuds sont mobiles, une zone de transmission trop restreinte risque de ne pas offrir de chemin. Par contre, cette zone ne doit pas être trop importante puisque le message est diffusé dans cette zone de transmission. Autrement dit, plus la zone est importante, plus le nombre de messages échangés pour atteindre la zone cible va être important et par conséquent, la bande passante sera utilisée de manière trop importante. Il est donc nécessaire, pour ce type de protocole, de bien définir la zone de transmission. Les auteurs proposent donc deux moyens de définir la zone de transmission : de manière explicite ou implicite. La manière explicite suppose que le message transmis contient l'information sur la zone de transmission. A l'inverse, la manière implicite ne requiert pas d'information dans le message mais une opération géométrique sur chaque nœud permet de savoir s'ils en font partie. En effet, en connaissant la distance entre le précédent expéditeur et soi-même, un nœud peut savoir s'il rapproche le message transmis de la zone cible ou pas. De cette manière et de proche en proche, le message peut atteindre la zone cible.

Dans [MCF<sup>+</sup>03], les auteurs présentent trois façons de gérer les messages géocasts. Ces messages doivent être disponibles pour *tous les nœuds présents dans la zone* et pendant une certaine durée que l'on peut qualifier de *durée de vie*. Leur premier mécanisme repose sur l'utilisation d'un serveur. Les messages géocasts sont envoyés au serveur qui stocke l'ensemble des messages géocasts de toutes les zones. Ensuite, périodiquement, le serveur envoie chaque message vers sa zone cible. Cette solution est très coûteuse en terme d'échanges de données puisque les messages doivent parcourir de grandes distances pour atteindre leur zone cible. La deuxième approche qu'ils proposent est basée sur l'élection d'un nœud comme responsable d'un message dans une zone cible. C'est-à-dire que pour chaque message géocast diffusé, il est d'abord diffusé vers la zone cible. Pour maintenir le message dans cette zone, un responsable est désigné. C'est lui qui stocke et conserve le message. Si ce responsable se déplace hors de la zone cible du message dont il est en charge, un nouveau responsable est élu et le message est transféré de l'un à l'autre. De cette manière, à chaque instant, il existe une instance du message géocast dans la zone cible. Pour accéder à l'information, il suffit d'interroger le bon nœud. Finalement, la dernière approche proposée s'appuie sur la duplication du message. Cette fois-ci, le message est acheminé vers la zone cible mais il n'y a plus de responsable ici. Tous les nœuds appartenant à la zone cible stockent le message. Lorsqu'un nœud entre dans la zone cible d'un message, il est détecté par les nœuds déjà présents. Il suffit de lui transférer le message, ce qui garantit qu'à chaque instant, tous les nœuds d'une zone cible possèdent le message géocast associé. Assez simplement, lorsqu'un nœud sort d'une zone cible, il lui suffit d'effacer le message géocast correspondant.

Les protocoles géocast permettent d'envoyer un message dans une certaine zone géographique.

Cela est particulièrement intéressant lorsque cette zone peut-être définie et que l'origine du message n'est pas dans cette zone. Par exemple, un véhicule pourrait utiliser un protocole géocast pour envoyer une demande des prix des carburants des stations services présentes sur sa route. Par contre, pour ce faire, il est nécessaire de connaître l'itinéraire du véhicule. De plus, pour qu'il puisse récupérer les différents tarifs, la réponse doit pouvoir lui revenir, ce qui n'est pas évident et nécessite des mécanismes supplémentaires. Une autre manière de faire est de disséminer les informations autour de l'endroit où elles sont intéressantes. C'est-à-dire que chaque station service diffuse ses tarifs de sorte que tous les véhicules proches d'une station service en connaissent les prix. Dans ce schéma, l'information est générée directement dans la zone où elle est intéressante et nous utiliserons plutôt des protocoles de dissémination.

### 2.3 Protocoles de dissémination

Les protocoles de dissémination permettent de diffuser une information autour de l'endroit où elle est générée. Dans les réseaux de capteurs, cela est très intéressant. En effet, en supposant qu'un capteur détecte un phénomène, la diffusion de l'information permet d'informer les proches terminaux. Par exemple, si un capteur dans une voiture détecte un accident, la diffusion de cette information peut permettre aux autres conducteurs d'être plus prudents et de ralentir pour éviter un sur-accident.

Dans [HKB99], les auteurs proposent SPIN – *Sensor Protocols for Information via Negotiation* – une famille de protocoles adaptatifs pour la dissémination d'informations dans les réseaux sans fil de capteurs. Ce type de réseau présente certaines contraintes. Tout d'abord, *l'énergie* est une ressource critique puisque chaque capteur fonctionne sur batterie et qu'il n'est pas toujours évident de les remplacer. Ensuite, les capteurs ont *des processeurs limités*. Finalement, comme pour tous les réseaux sans fil, *la bande passante* doit être économisée. Pour répondre aux contraintes d'économie d'énergie et de bande passante, ils proposent de définir les données avec des métadonnées. Ces métadonnées donnent une description des données contenues sur chaque capteur. De cette manière, l'échange des métadonnées est moins coûteux, aussi bien en terme de communication que d'énergie, que l'échange des données complètes. Toutefois, les auteurs ne se placent pas dans un contexte mobile. C'est-à-dire que chaque capteur est immobile et la topologie du réseau est donc fixe.

Dans le domaine de la communication inter-véhiculaire, d'autres travaux s'intéressent à la dissémination des informations. Dans [GION03], le mécanisme de dissémination proposé repose sur trois décisions. Quand, quoi et où diffuser une information? Pour savoir quelles informations diffuser, les auteurs proposent de trier les informations selon un certain critère. Cela peut être vu comme un mécanisme de pertinence permettant de déterminer les informations les plus intéressantes à trier. Pour savoir quand disséminer une information, ils proposent une méthode naïve qui consiste à diffuser périodiquement des informations, en utilisant au maximum la bande passante. Une méthode plus intelligente consiste à ne diffuser que lorsqu'il y a quelque chose de pertinent

à disséminer, en respectant un seuil de transmission maximum correspondant à la limite autorisée par la bande passante.

Dans RANDI – *RANK-based Dissemination* – [WXT07], les auteurs prennent en compte les limitations des ressources pour déterminer la quantité à échanger, à quel moment l'échanger et comment l'échanger. Les ressources qu'ils prennent en compte sont l'énergie, la bande passante et la capacité de stockage qui sont les ressources les plus critiques pour nos terminaux mobiles tels que nos téléphones.

Dans [ZWXS08, ZXW08a], les auteurs présentent TrafficInfo, un algorithme de dissémination d'informations sur le trafic. Pour cela, ils utilisent des données cartographiques. Grâce à elles, chaque véhicule compare le temps espéré pour parcourir un segment de route avec le temps réel qu'il a mis pour effectuer ce parcours. Il peut ainsi créer un *rapport* et le disséminer aux véhicules proches pour les informer du temps réel de parcours d'un segment de route. Chaque véhicule se crée ainsi une base de données avec les différents rapports reçus ou générés. Grâce à des techniques d'agrégation et à un mécanisme de pertinence, la quantité d'informations stockées sur chaque véhicule est limitée. Les rapports les plus anciens ou concernant un segment trop éloigné sont supprimés. Ils limitent ainsi la consommation de la bande passante, ressource critique dans notre contexte, tout en permettant à chaque véhicule d'avoir une connaissance des temps de parcours des segments proches. Ainsi, ils permettent d'optimiser le système de navigation. Leur solution est très intéressante dans le cadre de la dissémination d'informations sur le trafic. Cependant, elle est spécifique à ces données et ne peut pas être appliquée, par exemple, à la dissémination d'informations sur les places de stationnement disponibles ou sur un véhicule d'urgence en intervention. De plus, ils considèrent que, chaque fois qu'un véhicule parcourt un segment, il génère un rapport. Cela signifie que partout et à chaque instant, des rapports sont disséminés sur l'ensemble des routes. Même s'ils proposent un mécanisme pour limiter la consommation de la bande passante, la bande passante restante n'est pas suffisante pour que TrafficInfo soit couplé avec un autre système gérant d'autres informations.

Dans [LCM09, LM07], les auteurs proposent un mécanisme de dissémination d'informations dans un environnement hybride (voir section 1.3 (page 12)). Pour chaque information, une zone de persistance est définie. C'est dans cette zone que l'information doit être maintenue. Ensuite, dans cette zone, des points de répliqua sont déterminés grâce à une base de données cartographiques. En effet, pour un embouteillage par exemple, des informations cartographiques peuvent permettre de définir les endroits clés où l'information doit se trouver pour permettre aux véhicules d'être informés, cela afin de leur offrir la possibilité de choisir une route alternative. L'infrastructure partielle considérée permet de stocker et de garder les informations sur ces points de répliqua. Finalement, chaque véhicule souscrit à des informations en diffusant ses requêtes et son itinéraire périodiquement. L'itinéraire est obtenu grâce au système de navigation. De cette manière, chaque détenteur d'une information peut avertir les véhicules ayant manifesté leur intérêt. Les auteurs proposent une adaptation de leur solution pour le cas où il n'y a pas d'infrastructure. De cette manière, leur solution s'adapte à la disponibilité ou non d'une infrastructure. Les auteurs mentionnent que leur solution repose sur l'utilisation d'un système de navigation et que cela peut être

gênant. En effet, aujourd'hui, bien que ces systèmes soient de plus en plus présents, ils ne sont pas utilisés quotidiennement. Il est donc nécessaire de proposer une solution qui ne dépend pas de cette hypothèse.

Pour le système *TrafficView* [NSI06, NDLI04], les auteurs considèrent un ensemble de véhicules roulant sur plusieurs voies de circulation et dans chaque direction. Ils présentent différents protocoles de dissémination permettant de diffuser périodiquement toutes les données stockées sur un véhicule en utilisant un seul paquet réseau, grâce à des techniques d'agrégation.

Dans [RYA<sup>+</sup>09], les auteurs proposent un mécanisme d'ordonnancement des messages à diffuser. Ils expliquent que le fait qu'une information soit pertinente n'est pas suffisant dans le cas où beaucoup de messages doivent être diffusés. En effet, il est nécessaire de choisir les messages à diffuser en priorité. En utilisant des files d'attente, ils permettent de donner la priorité aux informations les plus importantes.

Il y a donc plusieurs questions à se poser lorsque l'on veut faire de la dissémination. Tout d'abord, il est important de garder à l'esprit que la bande passante est la ressource à économiser. Il faut donc disséminer le plus efficacement possible. Les solutions naïves comme l'inondation permettent de présenter le concept mais trouvent rapidement leur limite. Dans les réseaux de communication inter-véhiculaire, le nombre de véhicules et d'informations à transmettre peut rapidement être très important. Par contre, il est aussi nécessaire d'identifier les informations qui doivent être disséminées. Pour cela, un mécanisme de gestion des informations est nécessaire.

### 3 Gestion des informations

Dans cette section, nous nous intéressons à différentes formes de gestion des informations. En effet, derrière ce terme assez générique, nous incluons plusieurs éléments qui nous paraissent importants dans le cadre de la communication inter-véhiculaire. Tout d'abord, nous allons décrire différents mécanismes d'évaluation de la pertinence des informations. Ces mécanismes permettent de déterminer l'intérêt d'une information pour un utilisateur donné. Ensuite, nous nous intéressons à la gestion des ressources. En effet, même si la plupart des informations peuvent être exploitées par tout un ensemble de véhicules, ce n'est pas le cas des ressources. La gestion des informations est donc différente. Dans le cas d'un embouteillage par exemple, il est intéressant de prévenir le plus de véhicules possible pour permettre à tous les conducteurs d'éviter l'embouteillage. Par contre, dans le cas d'une place de stationnement, prévenir tout un ensemble de véhicules de la présence d'une place de stationnement peut s'avérer très gênant puisqu'un seul peut l'obtenir. Certains travaux se sont intéressés au problème et ont permis d'envisager des mécanismes de gestion des ressources très intéressants. Finalement, nous nous focaliserons sur l'interrogation des données. Jusqu'à présent, nous avons vu les architectures et les protocoles permettant à chaque véhicule de transmettre des informations. Un autre aspect à prendre en considération est l'interrogation de ces informations. Etant donné qu'elles peuvent être nombreuses, il est en effet nécessaire de proposer des techniques efficaces d'interrogation afin de les exploiter au mieux.

### 3.1 Evaluation de la pertinence

L'évaluation de la pertinence consiste en un mécanisme permettant de déterminer si une information est intéressante ou non à un moment donné pour un véhicule donné. Ce genre de mécanisme est absolument nécessaire dans les systèmes d'information véhiculaires puisque c'est ce mécanisme qui permet de ne communiquer aux conducteurs que les informations qui les concernent.

Dans [GION03], les auteurs s'intéressent à la dissémination d'informations sur le trafic. Pour cela, chaque véhicule possède une base de données cartographiques contenant l'ensemble des routes avec les temps de parcours optimaux. C'est-à-dire lorsque la route peut être parcourue en roulant au maximum de la vitesse autorisée. Chaque véhicule circulant sur un tronçon peut alors comparer son temps de parcours avec le temps de parcours idéal. De cette manière, chaque véhicule met à jour sa base de données avec les temps constatés. Pour prévenir les autres véhicules des temps réels de parcours, il serait intéressant de disséminer toutes les informations sur toutes les routes parcourues. Pourtant, la contrainte de bande passante impose de faire des choix et de ne diffuser que les informations les plus pertinentes. Les auteurs proposent un mécanisme permettant de ne disséminer que les informations les plus utiles. Tout d'abord, lorsqu'un véhicule s'apprête à disséminer une information, un certain nombre de données sont présélectionnées. Il s'agit des informations concernant les routes proches. En effet, lorsqu'un véhicule diffuse une information, elle sera reçue par les véhicules proches. Or, ces véhicules sont intéressés en priorité par les informations des routes qu'ils risquent d'emprunter. Ensuite, les auteurs utilisent une fonction de pertinence permettant de ne sélectionner qu'un sous-ensemble de ces informations pour diminuer encore le nombre de données échangées. Pour cela, ils calculent la différence entre le temps de parcours estimé et le temps de parcours réel. De cette manière, pour un tronçon donné, plus la différence entre temps de parcours constaté et le temps de parcours idéal est importante, plus l'information sera considérée comme pertinente. En effet, il n'est pas utile de diffuser les informations sur des temps de parcours qui sont identiques aux temps de parcours idéaux puisque chaque véhicule possède déjà ces informations. Le plus important étant de mettre à jour les informations, il est plus intéressant de diffuser les informations qui auront un réel impact sur la base de données embarquée.

Certains travaux proposent des fonctions de pertinence très intéressantes. Dans [SWX05, WXYR05, WX04] par exemple, les auteurs évaluent la pertinence des informations en se basant sur des critères spatio-temporels. L'équation 2.1 (page 20) et l'équation 2.2 (page 20) nous donnent chacune une fonction de pertinence spatio-temporelle pour une information où  $\alpha$  et  $\beta$  sont des coefficients de pénalité respectivement sur l'âge ( $t$ ) et sur la distance ( $d$ ) par rapport à l'information.

$$R(e) = e^{-\alpha \times t - \beta \times d} \quad (\alpha, \beta \geq 0) \quad (2.1)$$

$$R(e) = -\alpha \times t - \beta \times d \quad (\alpha, \beta \geq 0) \quad (2.2)$$

Les mécanismes d'évaluation de la pertinence permettent donc de *trier les informations*. Grâce à cela, un traitement spécifique peut être réalisé en fonction de cette pertinence. Par exemple, les

informations les plus pertinentes vont être retransmises pour être exploitées par d'autres véhicules alors que les informations moins pertinentes vont être supprimées pour libérer de l'espace, en attendant de nouvelles informations.

#### 3.2 La gestion des ressources

Dans le cadre de la gestion des ressources, les places de stationnement font l'objet de recherches bien spécifiques. En effet, dans un système d'information destiné aux conducteurs, alerter les conducteurs des différents événements qui arrivent sur la route est très important. Alors que prévenir de la présence d'un accident permet d'éviter les sur-accidents, que détecter la présence d'un embouteillage permet à chacun de l'éviter, les places de stationnement ne doivent pas être fournies comme une information globale puisqu'elles ne sont exploitables que par un certain nombre de véhicules. Une ressource est limitée. En plus de fournir l'information, il est nécessaire d'optimiser son utilisation et d'éviter, autant que possible, la compétition entre les véhicules.

Dans [GOM08], les auteurs proposent une approche multi-agents pour allouer des places de stationnement dans un campus. Lorsqu'un véhicule arrive sur le campus, son terminal (PDA, ordinateur portable ou téléphone mobile) communique avec l'infrastructure disséminée sur le campus. Grâce à un serveur central et à des capteurs sur les places de stationnement, l'infrastructure alloue alors une place de stationnement proche de la destination de l'utilisateur. Bien que l'architecture utilisée soit peu exploitable dans des environnements plus importants (comme une ville entière par exemple), les auteurs ont su offrir un service efficace. Non seulement ils ont proposé un mécanisme permettant d'allouer des places de stationnement mais en plus, ils ont considéré les rôles de différents utilisateurs et le prix des places de stationnement. C'est-à-dire que sur le campus, certaines places de stationnement sont réservées au personnel. Dans leur approche, ils ont considéré le profil utilisateur pour permettre de choisir la bonne place pour la bonne personne. De plus, les places étant payantes, ils ont considéré le paiement grâce aux capteurs des places de stationnement. Ainsi, dès qu'un utilisateur quitte sa place, le prix est directement calculé. Leur solution est particulièrement adaptée à l'environnement qu'ils ont choisi, c'est-à-dire un campus universitaire. Bien que leur approche ne soit pas généralisable à l'échelle d'une ville, leur proposition fournit des pistes très intéressantes en terme de gestion de ressources. En particulier, la prise en compte du profil utilisateur et la gestion des prix des places de stationnement.

Pour aider dans la recherche de places de stationnement, l'utilisation des parcmètres est parfois suggérée. En effet, en équipant les parcmètres avec des modules de communication sans fil, les automobilistes pourraient recevoir des informations sur les places disponibles en centre ville par exemple. Dans [BL04], les auteurs proposent donc de construire des « réseaux de parcmètres ». De cette manière, une infrastructure permet de faciliter la coordination entre les véhicules.

Dans [CBSM07, CGM06], les auteurs prédisent l'occupation de places de stationnement en utilisant des chaînes de Markov. De cette manière, chaque véhicule peut estimer ses chances de

prendre possession de la ressource. Cette manière de faire n'élimine pas la compétition entre les véhicules puisqu'il ne s'agit pas d'affecter chaque place de stationnement à un véhicule mais elle permet de ne pas nécessiter de coordination entre les véhicules et chaque véhicule reste donc autonome.

Pour gérer les ressources, il est nécessaire de prendre en considération les besoins de chaque véhicule individuellement. Cela signifie que chaque ressource doit être allouée à un véhicule particulier. Par contre, il est aussi nécessaire de trouver un moyen de rendre équitable cette allocation. Par exemple, dans le cadre des places de stationnement, l'allocation doit permettre à chacun de trouver sa ressource aussi rapidement. Il faut, autant que possible, éviter de privilégier certains véhicules au détriment d'autres qui risqueraient de chercher pendant beaucoup plus longtemps.

### 3.3 Interrogation des données

Avec l'intégration de la mobilité, les bases de données deviennent de plus en plus complexes. Les solutions des bases de données traditionnelles ne sont alors généralement plus applicables. Dans [Wol02], Ouri Wolfson identifie les défis liés à la gestion des informations sur des objets mobiles. Il présente, entre autres, la représentation des données, l'interrogation des données, la distribution des informations de positionnement pour les MANETs, l'incertitude ou encore la confidentialité.

Par rapport à la représentation des données, certaines propositions permettent d'optimiser les processus de mise à jour en adoptant une représentation spécifique. Dans [SWCD97, SWCD98], les auteurs proposent le modèle de données MOST – *Moving Objects Spatio-Temporal*. Il s'agit d'un modèle où les attributs des objets peuvent être des fonctions du temps. De cette manière, ils offrent la possibilité d'avoir des données dont les mises à jour sont implicites. C'est-à-dire des données dont les attributs n'ont pas besoin d'être mis à jour pour prendre de nouvelles valeurs au cours du temps. Grâce à cette représentation, la base de données peut être interrogée sur son état futur. Par contre, l'interrogation sur les valeurs futures des attributs des données rend incertains les résultats. En effet, il n'est pas garanti que ces valeurs estimées se vérifient. Dans [WCD<sup>+</sup>98], les auteurs modélisent l'imprécision pour permettre d'interroger les données avec une certaine tolérance.

Dans [WSCY99, WSX<sup>+</sup>99, WJS<sup>+</sup>99], les auteurs s'intéressent à la mise à jour et à l'interrogation de bases de données traquant des objets mobiles. Étant donnée une telle base de données, il est nécessaire, en effet, de mettre à jour régulièrement les positions des objets qu'elle contient. Pour cela, les auteurs proposent de mettre à jour la position d'un objet lorsque celui-ci s'est déplacé d'une certaine distance. Le principe n'est donc pas de garder une information aussi fraîche que possible mais de garantir à chaque instant que la position des objets traqués est suffisamment proche de celle qui est inscrite dans la base de données. Ils proposent ensuite un mécanisme permettant d'interroger ces données. Dans de telles bases de données, les requêtes les plus utiles

sont celles qui portent sur la présence d'objets dans une certaine zone. Comme les objets sont mobiles et que les données de la base ne sont pas forcément à jour, cela signifie que le résultat d'une telle requête n'est pas certain. C'est pourquoi, ils introduisent une probabilité de présence permettant de répondre au mieux à ce genre de requête.

Dans [XW03], les auteurs proposent d'intégrer la prédiction des temps de parcours pour répondre à des requêtes. Autrement dit, chaque objet mobile est représenté avec un itinéraire. De plus, chaque tronçon de route possède une information sur le temps de parcours estimé. Pour répondre aux requêtes, des prédictions sont utilisées pour récupérer les résultats des requêtes.

Dans [VW01], les auteurs présentent *un modèle et un langage spatio-temporels pour les objets mobiles dans les réseaux routiers*. Pour cela, ils considèrent l'utilisation de données cartographiques permettant de spécifier précisément l'itinéraire d'un véhicule. Cela permet de ne nécessiter des mises à jour que lorsqu'un événement survient. Par exemple, lorsqu'un véhicule roule moins vite que prévu ou lorsqu'il change d'itinéraire. Dans le cas où l'itinéraire de départ est respecté, il n'y a plus besoin de faire de mise à jour.

Au delà de la représentation, les requêtes sont très importantes dans les environnements mobiles. Traditionnellement, les requêtes les plus difficiles à traiter sont les requêtes spatio-temporelles. C'est-à-dire les requêtes dont les résultats dépendent de la position des données interrogées. En effet, il est fréquemment intéressant de connaître les données présentes dans *une certaine zone* ou bien *les n données les plus proches* d'une certaine position à un certain moment. Pourtant, dans [ZXW08b], les auteurs proposent, pour les environnements mobiles non connectés, un nouveau type de requête : *les requêtes d'existence spatiale*. Il s'agit de requêtes dont le résultat est une seule donnée, présente dans un intervalle de temps donné et dans une zone géographique spécifiée. Les auteurs considèrent que les requêtes de zone ou des *n* plus proches voisins sont trop difficiles à traiter dans ce type d'environnements et que de toute façon, le plus important pour l'utilisateur n'est pas d'avoir tous les résultats mais que la connaissance de l'existence d'une seule donnée est suffisante. Ils proposent donc un mécanisme reposant sur *la dissémination de la requête* pour interroger les nœuds du réseaux puis sur *la dissémination du résultat* pour atteindre le générateur de la requête.

Dans [LWX08, LW08], les auteurs s'intéressent à l'interrogation de données dans *les bases de données pair-à-pair mobile*. Ces bases de données sont caractérisées par *une topologie fortement dynamique, des ressources limitées (mémoire, énergie, bande passante)* et par *une absence de coordination globale*. MOBI-DIK – *MOBIle Discovery of local resource Knowledge* – [CWXY05] est le nom du système qu'ils proposent. Ils considèrent qu'une *ressource réelle* est identifiée dans le monde virtuel par *un rapport*. Par exemple, une place de parking libre est identifiée par un rapport indiquant ses caractéristiques. Leur mécanisme repose sur la dissémination des rapports. Autrement dit, plutôt que de disséminer les requêtes puis d'attendre les résultats, ils proposent de disséminer les rapports pour que chaque nœud puisse se constituer une base de données locale contenant un grand nombre d'informations. C'est ensuite cette base de données qui est interrogée pour répondre aux requêtes de l'utilisateur. Cette base de données se construit donc petit à petit et

chaque véhicule reçoit donc un flux discontinu de rapports. Dans le domaine des flux de données, l'algorithme SOLE – *Scalable On-Line Execution* – [MA08] traite les requêtes continues sur des flux de données spatio-temporels. Cette solution permet d'évaluer plusieurs requêtes de manière concurrentes. En utilisant les tuples positifs et négatifs, les requêtes peuvent être mises à jour en fonction des flux de données reçus. De manière similaire mais pour les bases de données spatio-temporelles, les auteurs présentent SINA – *Scalable INcremental hash-based Algorithm* – qui permet le traitement de requêtes continues dans [MXA04].

D'autres travaux se sont intéressés à la modélisation et à l'interrogation d'objets mobiles dans les réseaux. En particulier dans [GDAD06] où toute une représentation et un système d'interrogation sont présentés.

L'interrogation des données dans les environnements mobiles n'est pas évidente en soi. La mobilité des informations et des nœuds rend les résultats des requêtes beaucoup plus difficiles à obtenir. De plus, certaines requêtes doivent être évaluées en continu. En effet, une évaluation traditionnelle permet de récupérer une réponse valide à un instant donné. Pourtant, ici, il est souvent intéressant de traquer les mises à jour des résultats. Autrement dit, avoir une réponse valide à un instant n'est pas toujours suffisant, surtout si cette réponse devient invalide très peu de temps après. Entre le moment à l'interrogation et le moment à l'exploitation, les changements doivent parfois être pris en compte. Pour prendre en compte ces changements, l'utilisation de « triggers » est parfois utilisée [TSWN04]. De plus, l'un des aspects particulier de ces nouvelles bases de données et qu'elles sont fortement distribuées. C'est-à-dire que chaque terminal possède ses informations. Si l'utilisation de serveurs offre des ressources suffisantes pour traiter des requêtes coûteuses en terme de temps de traitement, des terminaux embarqués n'ont pas les mêmes capacités. Permettre à chacun d'évaluer ces requêtes est très important et loin d'être évident. C'est pourquoi, il est toujours nécessaire de proposer un mécanisme d'interrogation des données qui soit adapté au contexte.

## 4 Autres travaux

Dans le cadre de la communication inter-véhiculaire, certains auteurs ont proposé des solutions répondant à des problèmes autres que ceux mentionnés précédemment. Dans la suite, nous proposons un survol de ces différents travaux, qui doivent être considérés afin de fournir un système efficace. Par exemple, certains travaux proposent de nouveaux protocoles de communication au niveau de la couche MAC ou de la couche physique. Dans [BCCF02, KGMRR<sup>+</sup>03, XSJ03, SG07], les auteurs proposent un protocole MAC pour assurer la bonne diffusion des données. En effet, lorsque l'on utilise des communications sans fil, il peut y avoir des *collisions* qui peuvent engendrer des pertes de données. Il est donc important d'utiliser un moyen de communication garantissant la bonne dissémination des informations. Dans [STC02], les auteurs s'intéressent à l'optimisation de CDMA – *Code division multiple access* – pour améliorer les communications, au niveau de la couche physique. Dans [ERW<sup>+</sup>03], les auteurs évaluent leur version modifiée de l'*UTRA-TDD*<sup>2</sup>, un

---

2. UMTS Terrestrial Radio Access – Time Division Duplexing.

moyen de communication basé sur l'UMTS mais adapté pour permettre une utilisation en réseau ad hoc.

L'impact du nombre de véhicules équipés sur les performances des systèmes n'est pas à démontrer. En effet, pour qu'un système de communication entre les véhicules fonctionne, il faut suffisamment de véhicules qui communiquent entre eux. Dans [ARP04], les auteurs étudient la connectivité des véhicules sur autoroute. Ils montrent ainsi que sur autoroute, plus la densité de véhicules est importante, plus la connectivité est importante, ce qui est logique. Par contre, ils montrent que le nombre de voies joue un rôle dans la connectivité et que plus le nombre de voies est important, plus la connectivité est forte. Un autre aspect lié à l'échange d'informations dans les réseaux ad hoc mobiles est la motivation à participer. En effet, même s'il y a suffisamment de véhicules équipés et que tous les acteurs veulent profiter des données, tous ne sont pas forcément très motivés pour fournir des données. Dans [ZCY03], les auteurs présentent un mécanisme purement logiciel basé sur un système de paiement. De cette manière, les différents utilisateurs ont un intérêt à participer. Dans [XWR06], un système de paiement au niveau matériel a pour but de motiver les utilisateurs à participer. Par contre, si certains conducteurs acceptent d'échanger des données, les questions de sécurité doivent être posées. Dans [LLZ<sup>+</sup>08, RH07, FFBA07, ABD<sup>+</sup>06, RH05b, Döt05, RH05a, GGS04, HČL04], les différents auteurs s'intéressent à la sécurité dans les réseaux de communication inter-véhiculaire. En effet, il est nécessaire de garantir l'intégrité des données et de prévenir les attaques qui pourraient survenir dans de tels réseaux. En même temps, il est aussi nécessaire de préserver l'anonymat des différents utilisateurs.

Afin de valider les différentes propositions faites, la simulation est souvent utilisée. En effet, il est difficile d'implémenter un prototype et de le déployer à grande échelle pour pouvoir l'évaluer. Dans le domaine des réseaux, le principal simulateur utilisé est NS – *Network Simulator* – [PEBS00]. L'une des difficultés avec les simulateurs est le modèle de mobilité. En effet, pour évaluer efficacement les solutions, il est important de les simuler dans des conditions aussi proches de la réalité que possible. Certains auteurs se sont intéressés aux modèles de mobilité [BHM07, FHFB07, KML05] et ont tenté d'offrir de nouveaux outils pour les simulations. Pourtant, certains auteurs ont démontré la faisabilité et l'utilité de la communication entre les véhicules à travers des démonstrateurs. Des plateformes d'expérimentations ont été développées en utilisant des robots [SYG06]. Dans [KTT<sup>+</sup>02], les auteurs utilisent 5 véhicules équipés de DGPS, de radar et de module de communication pour montrer qu'il est possible d'envisager des véhicules autonomes. Grâce aux DGPS, la précision du système de localisation est de l'ordre du centimètre. Les radars permettent d'éviter les collisions tandis que la communication entre les véhicules leur permet de coopérer pour effectuer les manœuvres d'insertion dans une file par exemple. Les auteurs montrent donc que la communication inter-véhiculaire permet à chaque véhicule de prendre les bonnes décisions pour effectuer ces manœuvres. Il ne s'agit donc pas de partager une information sur une longue distance mais d'utiliser une communication de proximité pour mettre tous les véhicules d'accord quant aux décisions à prendre. Dans [HFMF03], les auteurs présentent un scénario de démonstration de *FleetNet* avec 6 véhicules et un routeur.

### 5 Conclusion

La recherche dans le domaine de la communication inter-véhiculaire est très active et couvre de nombreux aspects. Nous avons tout d'abord vu les différentes architectures qu'il est possible d'utiliser. L'utilisation seule d'une infrastructure ne permet pas d'envisager de fournir des informations suffisamment à jour. Si, pour les applications visant à fournir de l'information trafic en temps réel, un délai est acceptable, cela n'est pas envisageable pour les applications liées à la sécurité. Il est donc nécessaire de proposer un système exploitant la communication sans fil entre les véhicules. Une infrastructure adaptée peut permettre d'assister les échanges entre les véhicules de diverses manières. La meilleure architecture est donc probablement une architecture hybride pair-à-pair. De cette manière, chaque véhicule reste autonome et peut communiquer avec les autres en même temps qu'une infrastructure peut être exploitée si elle est présente mais sans être absolument nécessaire.

Concernant les protocoles de communication, chaque famille possède ses avantages et ses inconvénients. Pour les applications où il s'agit d'interroger des serveurs fixes pour recueillir des informations, les protocoles de routage sont très utiles. Ils permettent d'envoyer une information ou une requête vers ces serveurs. Par contre, s'il s'agit d'interroger les nœuds d'une certaine zone géographique, les protocoles géocasts sont meilleurs. En exploitant les informations de localisation, ils permettent d'atteindre les zones cibles et d'informer un ensemble de nœuds efficacement. Pour nous, l'application visée est le partage d'informations. Dans ce contexte, les protocoles de dissémination sont plus adaptés. Lorsqu'une information est générée ou captée par un nœud, les protocoles de dissémination permettent de « contaminer » un ensemble de véhicules d'une certaine région spatio-temporelle. Il s'agit donc de dupliquer l'information sur un ensemble de véhicules pour que chacun puisse exploiter cette information localement.

Le fait de reposer sur un tel mécanisme impose une gestion particulière des informations. En effet, puisque chaque véhicule reçoit des données en permanence, cela signifie que les données reçues ne sont pas demandées. L'exploitation locale représente donc un challenge important aussi. Tout d'abord, les mécanismes de pertinence permettent de réaliser un filtrage des informations. En proposant une fonction d'évaluation adaptée, chaque véhicule peut limiter le nombre d'informations stockées et ne garder que celles qui le concernent. Ce premier filtrage permet de préserver la capacité de stockage de chaque nœud mais ne répond pas complètement aux besoins des utilisateurs. Si les informations concernent des événements, il est nécessaire de proposer un mécanisme d'interrogation permettant de répondre aux requêtes posées par le conducteur. Dans le cas des ressources, non seulement il est nécessaire de répondre aux requêtes du conducteur mais il faut aussi prendre en compte l'aspect compétitif lié à l'acquisition des ressources. Que ce soit par des mécanismes d'allocation ou par des techniques d'évaluation de la disponibilité des ressources, la prise en compte des autres conducteurs ne peut pas être négligée. Ainsi, là encore, même si de nombreux travaux ont été réalisés, certaines contributions restent à faire.

Dans le chapitre 3 (page 29), nous détaillons notre mécanisme d'évaluation de la pertinence. Notre mécanisme d'évaluation continue est expliqué dans le chapitre 4 (page 47). Notre dernière

contribution fait l'objet du chapitre 5 (page 63) où nous exposons notre manière de disséminer les informations.



# 3

---

## Evaluation de la pertinence

---

Dans ce chapitre, il s'agit pour nous de présenter notre mécanisme permettant d'évaluer la pertinence d'une information pour un véhicule et pour un conducteur. Ce mécanisme est primordial dans un système d'information visant à assister les conducteurs puisque c'est lui qui va déterminer quelles informations vont être exploitées. En effet, chaque véhicule reçoit, au cours de son parcours, des informations sur son environnement. Ces informations doivent être évaluées avant d'être éventuellement transmises au conducteur. Un tel mécanisme doit permettre de sélectionner, pour chaque information, les véhicules qui doivent la prendre en compte et ceux qui doivent l'ignorer. Le challenge est donc que cette sélection soit aussi précise que possible. L'objectif est que l'information soit prise en compte par *tous et seulement* les véhicules dont les conducteurs sont potentiellement intéressés. La difficulté est double :

1. Fournir l'information à un minimum de véhicules par erreur. C'est-à-dire éviter qu'un véhicule considère pertinente une information qui ne peut pas l'être pour son conducteur.
2. N'oublier aucun véhicule.

Cela implique donc un filtrage permettant de limiter le nombre de véhicules prenant en compte une information. Mais ce filtrage ne doit pas être trop restrictif car le plus important reste que les véhicules ne doivent pas filtrer une information qui doit être considérée pertinente. Dans le domaine routier, l'évaluation de la pertinence fait intervenir des critères spatio-temporels puisque les informations ne sont exploitables que par certains véhicules appartenant à une zone spatio-temporelle particulière. Afin d'évaluer au mieux la pertinence d'une information, il convient d'identifier les informations que nous considérons. Dans un premier temps, nous proposons une classification des événements. Cela nous permettra d'identifier les éléments importants à prendre en compte et nous verrons alors comment nous représentons les données. A partir des données que nous considérons et de leur représentation, nous pourrons exposer notre mécanisme d'évaluation de la pertinence. En s'appuyant sur la définition de vecteurs, nous expliquerons *la probabilité de rencontre* qui est, pour nous, la fonction définissant la pertinence d'une information. La conclusion permettra d'identifier les points forts et les points faibles de notre proposition.

### 1 Représentation des informations

A l'heure actuelle, les solutions existantes dans le domaine de la communication V2V – *Vehicle To Vehicle* – n'ont considéré qu'un petit sous-ensemble de types d'événements, s'intéressant surtout aux événements immobiles. Pourtant, il existe d'autres types d'événements dans le cadre de la communication inter-véhiculaire puisque les conducteurs peuvent être intéressés par de nombreuses informations. Nous pouvons citer quelques exemples : accidents, embouteillages, situations de freinage d'urgence, informations sur les prix des stations services, véhicules d'intervention d'urgence tels que les ambulances, obstacles sur la chaussée, ou encore le comportement dangereux de certains conducteurs. Dans ce genre de situation, la collaboration entre les véhicules peut permettre de rendre plus confortable la conduite. Les événements étant très diversifiés, nous allons présenter maintenant *une classification des différents types d'événements*. Cela nous permettra ensuite d'explicitier notre manière de les représenter.

#### 1.1 Classification des événements

Nous ne voulons pas uniquement supporter les événements immobiles, comme la présence d'un accident, mais aussi les événements mobiles, comme un véhicule d'intervention d'urgence demandant aux véhicules le précédant de lui laisser la priorité. Considérer des événements mobiles est un réel challenge puisque l'ensemble des véhicules pour lesquels l'information est pertinente évolue non seulement en fonction du déplacement des véhicules impliqués (dans notre exemple, les véhicules qui reçoivent l'information) mais aussi en fonction du déplacement du véhicule porteur de l'événement (dans notre exemple, il s'agit du véhicule d'intervention d'urgence). A l'heure actuelle, aucune solution existante ne supporte de tels événements. En plus de la prise en compte de la mobilité, il convient de tenir compte du sens de circulation pour établir la pertinence des

informations, même pour certains événements immobiles. Si l'on considère un embouteillage, seuls les véhicules circulant dans le même sens de circulation sont concernés. Chaque événement peut donc être dépendant de la direction s'il n'est pas pertinent pour tous les véhicules proches mais seulement pour ceux circulant dans une direction particulière. Et chaque événement peut être mobile si sa position change au cours du temps.

A partir de ces constats, nous avons classifié les événements dans les réseaux de communication inter-véhiculaire en quatre catégories que nous illustrons en donnant quelques exemples :

- **Événement immobile, indépendant de la direction** : une place de stationnement disponible correspond à un événement immobile indépendant de la direction puisqu'elle est statique et peut intéresser tous les proches véhicules quelque soit leur sens de circulation.
- **Événement immobile, dépendant de la direction** : un accident sur une autoroute est un événement immobile dépendant de la direction puisque la position de l'accident est figée mais seuls les véhicules risquant d'être gênés par l'accident sont intéressés par l'information. Pour les véhicules proches mais roulant dans le sens opposé de circulation, cette information risque d'être perturbante plutôt que pertinente.
- **Événement mobile, indépendant de la direction** : un conducteur dangereux en centre ville est un événement mobile indépendant de la direction. En effet, ce véhicule se déplace dans un sens de circulation mais risque d'être dangereux, peut-être même plus dangereux, pour les véhicules circulant sur la voie opposée de circulation. L'information doit donc être transmise à tous les véhicules proches de l'événement en prenant en considération la mobilité de ce véhicule mais pas sa direction.
- **Événement mobile, dépendant de la direction** : pour finir, le meilleur exemple d'événement mobile dépendant de la direction est sans doute l'exemple du véhicule d'intervention d'urgence. Ce véhicule se déplace très rapidement et doit prévenir l'ensemble des véhicules qu'il croise de son arrivée pour qu'ils le laissent passer. Dans notre vie courante, cela est déjà réalisé par l'utilisation des sirènes qui permettent d'attirer l'attention des conducteurs et ainsi, permettent de libérer le passage. Pourtant, l'utilisation de ces sirènes représente une nuisance sonore et n'est pas efficace à 100%. Il arrive que des conducteurs ne l'entendent pas (musique dans la voiture...). De plus, tous les conducteurs qui entendent une sirène ne sont pas nécessairement concernés. L'utilisation de moyen de communication inter-véhiculaire est l'alternative la plus efficace puisqu'elle permettrait de ne pas provoquer de nuisance sonore, de ne prévenir que les véhicules concernés et de les prévenir plus rapidement.

Cette classification est assez simple tout en étant complète. L'intérêt ici n'était pas de fournir une liste exhaustive de tous les événements qui peuvent survenir sur nos routes mais bien de catégoriser les événements de manière à pouvoir les regrouper dans des ensembles. Lister tous les événements permettrait d'être plus précis pour les événements spécifiquement choisis mais nous ne pourrions pas proposer un système capable de gérer toutes les situations et tous les événements, y compris les événements pour lesquels notre solution n'aurait pas été développée au départ. C'est pourquoi notre classification se veut aussi générique que possible. Grâce à cela,

tous les événements peuvent être traités par notre solution. Il s'agit maintenant de représenter les informations de manière à être capable de les exploiter au mieux.

### 1.2 Représentation des données

Les quatre types d'événements identifiés précédemment sont utilisés pour représenter tous les événements pouvant survenir sur les routes. Dans la suite, nous décrivons comment ces différents événements sont représentés lors de leur génération afin qu'ils puissent être exploités par les véhicules :

- ◊ Tout d'abord, chaque événement doit être caractérisé par une Clef unique générée par la concaténation d'un identifiant unique de véhicule (par exemple, son adresse MAC) avec un identifiant local d'événement unique.
- ◊ Une Version est aussi attachée à chaque événement afin de distinguer différentes mises à jour du même événement. Une fois généré, un événement est disséminé le long d'un ensemble de véhicules potentiellement intéressés. Pour mettre à jour l'information transmise aux autres véhicules, par exemple pour mettre à jour la position d'un événement mobile, le véhicule générateur de l'événement peut produire une nouvelle version du même événement. Cette Version permet aussi de gérer la durée de vie d'une information. En effet, il est impossible de générer des messages d'invalidation. Lorsqu'une information sur un événement est disséminée, il est impossible de déterminer quels véhicules vont la prendre en compte puisque c'est chaque véhicule, localement et individuellement, qui va déterminer sa pertinence. Autrement dit, le générateur d'une information n'a aucun moyen de prévenir chaque véhicule que l'événement qu'il a généré n'est plus valide. Chaque événement est donc considéré comme invalide par chaque véhicule, localement, au bout d'un certain temps. La version permet de réactualiser un événement dont la validité est toujours en cours.
- ◊ Pour déterminer si un événement doit être présenté de manière urgente au conducteur, chaque événement possède une Importance. Par exemple, si un véhicule reçoit un message indiquant un accident quelques centaines de mètres plus loin, il est essentiel d'avertir le conducteur pour qu'il puisse agir de manière adéquate. L'importance associée à cet événement devrait donc être fixée au maximum. A l'inverse, les événements ayant une importance faible ne devraient pas nécessairement être fournis au conducteur.
- ◊ Chaque événement possède aussi la date et la position lors de la génération. Ces informations constituent un vecteur à 4 dimensions représenté par l'attribut Position courante.
- ◊ Deux positions précédentes dites « de références » ainsi que leur date sont également stockées. Il s'agit de la Position de référence pour la direction et de la Position de référence pour la mobilité. Ces deux positions fournissent aux véhicules recevant l'événement les informations nécessaires pour estimer la pertinence de l'événement. Cette estimation est détaillée dans la section 2.2 (page 40)
- ◊ La Position du dernier diffuseur contient la position du dernier véhicule qui a relayé l'information ou plus précisément, la position du véhicule par lequel l'information a été reçue au

moment où l'information a été diffusée. Cette information est utilisée pour le protocole de dissémination, expliqué dans le chapitre 5 (page 63).

- ◊ L'attribut Numéro de saut indique le nombre de rediffusions qu'il a été nécessaire d'effectuer pour atteindre le véhicule. Cette information est spécifique au protocole de dissémination (voir le chapitre 5 (page 63)).
- ◊ Pour terminer, une Description permet de décrire plus précisément l'événement représenté (e.g. un accident, un freinage d'urgence, une place de stationnement disponible, ...). Cet attribut permet de définir concrètement l'information qui doit être transmise au conducteur au moment où il doit être prévenu.

Les attributs utilisés pour décrire les événements sont résumés dans le tableau 3.1 (page 33). Les positions présentées dans le tableau (de type PositionAndTime) correspondent à des relevés GPS incluant la position en 3 dimensions ainsi que l'information sur le temps GPS du relevé. L'utilisation du temps GPS nous évite les problèmes liés à la synchronisation des horloges de chaque véhicule.

Nom de l'attribut	Abréviation	Type
Clef unique	Key	string
Version	V	int
Importance	I	int
Position courante	CurrentPos	PositionAndTime
Position de référence pour la direction	DRP	PositionAndTime
Position de référence pour la mobilité	MRP	PositionAndTime
Position du dernier diffuseur	LastDiffPos	PositionAndTime
Numéro de saut	H	int
Description	Desc	EventDescription

TABLE 3.1 – Représentation des événements

Notons que le type de l'événement (mobile ou immobile, dépendant ou non de la direction) n'est pas représenté explicitement comme étant un attribut dans notre représentation. En effet, le type peut être facilement déduit à partir des positions de références fournies par notre représentation. Ces positions sont utilisées pour déterminer des vecteurs représentant la direction et la mobilité de l'événement. Ainsi, si la position de référence pour la direction (DirectionRefPosition) est nulle, l'événement n'est pas dépendant de la direction. De manière analogue, si la position de référence pour la mobilité (MobilityRefPosition) est nulle, l'événement n'est pas mobile. C'est donc lors de la génération de l'événement que le type doit être connu puisque c'est lors de la génération que les valeurs de ces attributs seront attribuées. Par contre, lorsqu'un véhicule reçoit une information, le traitement sera le même quelque soit le type de l'événement.

Nous avons choisi une représentation très simple. En particulier, nous avons considéré ici une description représentée par un simple message fourni au conducteur. Cela permet à chaque message

d'être transmis dans un seul paquet. Notons qu'il est envisageable d'utiliser une représentation plus complexe permettant de fournir des informations plus précises sur les événements. En particulier, l'utilisation de contenu multimédia, comme des photos ou du contenu audio, permettrait de fournir des informations additionnelles très intéressantes dans certains cas [CDI09]. Par exemple, une libération de place de stationnement pourrait être accompagnée d'une photo de l'emplacement qui montrerait si c'est une grande ou une petite place.

### 1.3 Exemples de messages échangés

Dans cette section, nous présentons quelques exemples de messages générés correspondant à différents types d'événements. Le tableau 3.2 (page 34) décrit un événement immobile indépendant de la direction correspondant à une place de parking libérée. La première ligne montre le message émis par un véhicule libérant sa place de stationnement (ou par un capteur sur la place de parking détectant que le véhicule part). Aucun des attributs correspondant aux positions de références n'est rempli ici puisqu'il s'agit d'un événement de type immobile indépendant de la direction (DirRefPos et MobRefPos sont nulles pour ce type d'événements). La seconde ligne montre le message émis par un véhicule ayant reçu l'information et la relayant. Les seuls attributs qui sont modifiés ici sont le Numéro de saut et la Position du dernier diffuseur. Le Numéro de saut est incrémenté puisqu'un nouveau saut va être effectué. Pour la Position du dernier diffuseur, elle est fixée à la position du véhicule lui-même puisqu'il va émettre l'événement et donc, pour ceux qui vont la recevoir, sa position courante (sa position au moment où il va émettre) sera la Position du dernier diffuseur. La troisième ligne montre le message émis par le véhicule suivant. Pour des raisons de simplicité, seules la longitude et la latitude sont représentées pour les positions GPS. Nous n'avons pas montré l'altitude dans un souci de lisibilité.

Key	V	I	CurrentPos	DRP	MRP	LastDiffPos	H	Desc
VaM1	1	1	50°19'15.91 N 3°30'51.11 E 10h25m17s			50°19'15.91 N 3°30'51.11 E 10h25m17s	0	Available parking space
VaM1	1	1	50°19'15.91 N 3°30'51.11 E 10h25m17s			50°19'17.51 N 3°30'54.71 E 10h25m18s	1	Available parking space
VaM1	1	1	50°19'15.91 N 3°30'51.11 E 10h25m17s			50°19'18.85 N 3°30'58.04 E 10h25m20s	2	Available parking space

TABLE 3.2 – Exemple de messages diffusés pour un événement immobile indépendant de la direction

Le tableau 3.3 (page 35) décrit l'information sur un accident qui est un événement immobile dépendant de la direction. Puisqu'il s'agit d'un accident, l'importance du message est supérieure à celle de la place de stationnement. Nous considérons pour le moment que l'importance est une valeur comprise entre 1 et 10. Nous supposons donc que pour chaque événement, une correspondance avec une table d'importance permet d'affecter la valeur. La première ligne montre le message

initial. Cette fois-ci, il y a une position de référence pour la direction puisque les véhicules recevant cette information doivent estimer s'ils vont être gênés ou non. La seconde ligne montre le message après un saut.

Key	V	I	CurrentPos	DRP	MRP	LastDiffPos	H	Desc
XbC3	1	10	50°19'20.50 N 3°30'53.93 E 10h32m35s	50°19'20.81 N 3°30'53.04 E 10h32m34s		50°19'20.50 N 3°30'53.93 E 10h32m35s	0	Accident
XbC3	1	10	50°19'20.50 N 3°30'53.93 E 10h32m35s	50°19'20.81 N 3°30'53.04 E 10h32m34s	-	50°19'23.48 N 3°30'52.48 E 10h32m36s	1	Accident

TABLE 3.3 – Exemple de messages diffusés pour un événement immobile dépendant de la direction

## 2 Estimation de la pertinence des événements

L'un des problèmes principaux dans un système d'information visant à assister les conducteurs est de déterminer la pertinence d'un événement pour un véhicule qui reçoit l'information. Dans cette section, nous montrons d'abord comment les données décrivant un événement sont exploitées pour calculer un vecteur de mobilité à quatre dimensions (trois dimensions pour l'espace et une dimension temporelle). Ensuite, nous introduisons la notion de probabilité de rencontre. Cette probabilité de rencontre est utilisée pour estimer si oui ou non un véhicule a des chances de rencontrer un événement et donc, si l'événement doit être considéré ou pas.

### 2.1 Vecteur mobilité & direction

Pour estimer la direction et la mobilité d'un objet mobile (un événement mobile ou un véhicule), nous utilisons des vecteurs obtenus à partir d'une position précédente (appelé « position de référence ») et la position courante de l'objet. Ces vecteurs s'appliquent à tous les terminaux équipés d'un système de positionnement. Dans notre système, nous considérons que tous les véhicules et tous les générateurs d'événements<sup>1</sup> en sont équipés. Notons que même si ce principe implique l'utilisation d'un système de positionnement pour obtenir les positions nécessaires, nous n'utilisons aucun système de navigation ni aucune carte. L'utilisation de cartes ou d'un système de navigation permettrait d'estimer les positions futures (qui sont nécessaires pour le calcul de la probabilité de rencontre) des objets mobiles plus précisément. En effet, nous pourrions corriger nos vecteurs en fonction des cartes et dans le cas où nous aurions la destination de l'objet mobile, nous pourrions même estimer sa trajectoire jusqu'à cette destination. Pourtant, notre proposition ne repose pas sur l'utilisation de données cartographiques. En effet, nous voulons que notre proposition puisse être

1. Les générateurs d'événements sont principalement les véhicules mais il peut aussi s'agir de bornes ou de capteurs disséminés le long des routes.

utilisée partout et pour cela, nous devons aussi considérer les lieux où les utilisateurs n'ont pas de données cartographiques. Aujourd'hui, les systèmes de navigation embarquent une carte mais cette carte est parfois limitée à un seul pays. Si notre système reposait sur l'utilisation de ces cartes, cela signifierait que nous limiterions son utilisation aux seuls pays dont l'utilisateur possède la carte. Les services web permettant d'accéder aux cartes du monde entier sont très intéressants mais là encore, le coût lié à la connexion est très important, d'autant plus dans les pays étrangers. En plus de cela, les cartes ne contiennent pas toutes les informations. Par exemple, les entrées des parkings et les routes dans ces parkings ne sont pas incluses dans les bases de données cartographiques. Autrement dit, l'utilisation de cartes restreint l'utilisation du système. De plus, même si les systèmes de navigation se sont démocratisés et que nous sommes de plus en plus nombreux à les utiliser, nous ne les utilisons que pour certains trajets. Nous n'utilisons pas les systèmes de navigation pour les trajets habituels. Pourtant, un système de partage d'informations permettant aux conducteurs de prendre les bonnes décisions, a pour vocation d'être utilisé quotidiennement. Ainsi, même si des cartes sont disponibles, il n'est pas envisageable de supposer la destination des véhicules connue à priori. Pour toutes ces raisons, notre proposition n'exploite pas de données cartographiques, bien que nous savons que l'utilisation de cartes permettrait d'améliorer nos prédictions.

Dans la suite, nous exposons comment les vecteurs mobilité et direction sont calculés. Pour faciliter la compréhension, nous utilisons un système de coordonnées cartésien. Ces coordonnées cartésiennes sont obtenues par projection des coordonnées obtenues par le GPS<sup>2</sup>. Tout d'abord, la position de l'objet  $A$  à l'instant  $t$  est exprimée comme suit :

$$P_A(t) = \begin{pmatrix} x_{A_t} \\ y_{A_t} \\ z_{A_t} \\ t \end{pmatrix} \quad (3.1)$$

où  $x_{A_t}$ ,  $y_{A_t}$  et  $z_{A_t}$  sont les coordonnées cartésiennes géographiques de l'objet  $A$  à l'instant  $t$ . Nous exprimons le vecteur de mobilité de l'objet  $A$  entre  $t_1$  et  $t_2$  comme :

$$V_A(t_1, t_2) = P_A(t_2) - P_A(t_1) \quad (3.2)$$

$$V_A(t_1, t_2) = \begin{pmatrix} x_{A_{t_2}} - x_{A_{t_1}} \\ y_{A_{t_2}} - y_{A_{t_1}} \\ z_{A_{t_2}} - z_{A_{t_1}} \\ t_2 - t_1 \end{pmatrix} \quad (3.3)$$

Chaque véhicule peut donc calculer son propre vecteur de mobilité. En appliquant ce vecteur à la position courante du véhicule, une estimation de sa position future est obtenue :

---

2. Les projections utilisées dépendent de la région de la Terre considérée

$$P_A(t_{n+i}) = P_A(t_n) + V_A(t_{n-i}, t_n) \quad (3.4)$$

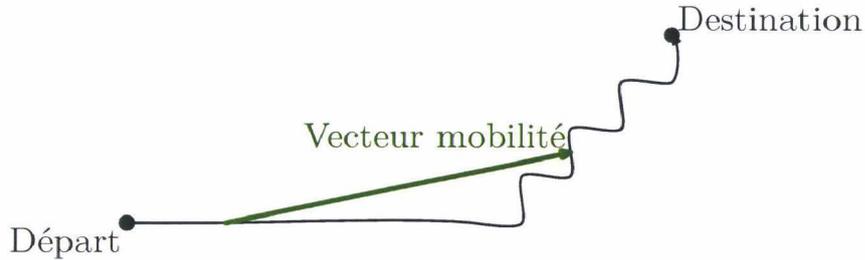


FIGURE 3.1 – Vecteur mobilité calculé avec un grand intervalle  $[t_{n-i}, t_n]$

La position future estimée est extrêmement dépendante de l'intervalle de temps choisi entre les deux positions composant le vecteur de mobilité. En effet, si  $t_n$  et  $t_{n-i}$  sont très éloignés (voir la figure 3.1 (page 37)), l'estimation de la position future n'est pas très précise mais fournit une bonne vue globale du déplacement de l'objet. A l'inverse, si l'intervalle de temps  $[t_{n-i}, t_n]$  utilisé entre les deux positions est très court, l'estimation de la position future est bien plus précise dans l'immédiat mais aucune information sur le déplacement global de l'objet peut être obtenue (voir la figure 3.2 (page 37)).



FIGURE 3.2 – Vecteur mobilité calculé avec un petit intervalle  $[t_{n-i}, t_n]$

En fonction de l'intervalle de temps entre les deux positions de référence, deux vecteurs sont établis :

- ◇ Le vecteur de mobilité dont le rôle est de fournir une impression générale sur le mouvement de l'objet et une bonne estimation de la position future. Pour avoir un bon compromis entre les deux précédents cas (figure 3.1 (page 37) et figure 3.2 (page 37)), un intervalle « moyen »  $[t_{n-i}, t_n]$  doit être utilisé pour ce vecteur de mobilité (voir la figure 3.3 (page 38)).
- ◇ Le vecteur de direction, qui est obtenu avec un intervalle court  $[t_{n-i}, t_n]$  donne une position future estimée très précise mais seulement dans un futur très proche.

En plus de définir leurs propres vecteur de mobilité et vecteur de direction, chaque véhicule peut calculer ces vecteurs pour les événements reçus. Il suffit d'utiliser les données associées aux

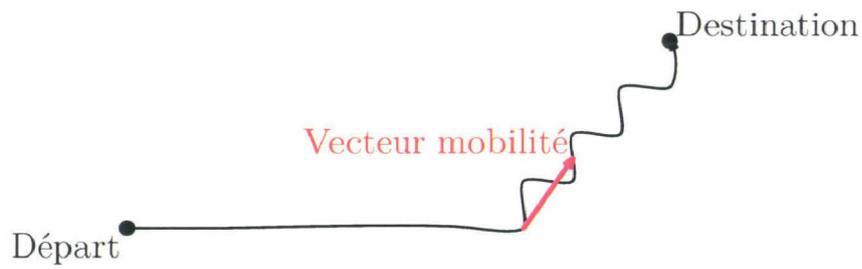


FIGURE 3.3 – Vecteur mobilité calculé avec un intervalle moyen

événements, et plus précisément la position courante avec la position de référence pour la mobilité pour le vecteur de mobilité ou la position de référence pour la direction pour le vecteur de direction.

Pour chaque événement, un vecteur de mobilité du véhicule par rapport à l'événement est calculé. Il s'agit ici de déterminer le vecteur de mobilité du véhicule en prenant l'événement comme référentiel. L'objectif est « d'annuler » la mobilité de l'événement en l'intégrant dans le vecteur de mobilité du véhicule. De cette manière, nous ne manipulons qu'un seul vecteur de mobilité par couple véhicule/événement. Les événements mobiles pourront être considérés comme immobiles dans le calcul de la pertinence. La figure 3.4 (page 38) montre le changement de référentiel. A gauche, nous voyons le vecteur de mobilité d'un véhicule et celui d'un événement. Il s'agit donc d'un événement mobile. A droite, nous voyons le résultat du changement de référentiel. En appliquant l'inverse du vecteur de mobilité de l'événement à notre référentiel, nous obtenons que l'événement est bien immobile par rapport à ce nouveau référentiel et nous avons un nouveau vecteur pour le véhicule. De cette manière, le calcul de la probabilité de rencontre (voir la section 2.2 (page 40)) est simplifié puisque nous avons un seul vecteur pour chaque couple (véhicule, événement), même si l'événement est mobile. Naturellement, il apparaît que ce changement de référentiel peut aussi être appliqué pour un événement immobile. Ce changement de référentiel ne modifiera pas le vecteur de mobilité du véhicule de base mais cela permet de montrer que la technique utilisée est vraiment indépendante du type d'événement considéré.

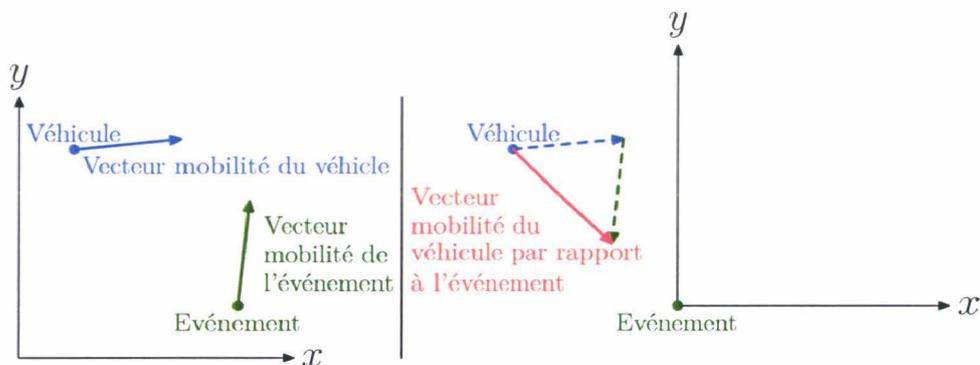


FIGURE 3.4 – Illustration d'un changement de référentiel

Pour expliquer comment ce changement de référentiel est réalisé, considérons le vecteur de mobilité d'un véhicule  $A$  calculé avec un intervalle  $[t_{A_1}, t_{A_2}]$  et le vecteur de mobilité d'un événement  $B$  calculé avec un intervalle  $[t_{B_1}, t_{B_2}]$  :

$$V_A(t_{A_1}, t_{A_2}) = \begin{pmatrix} x_A \\ y_A \\ z_A \\ \Delta t_A \end{pmatrix}, V_B(t_{B_1}, t_{B_2}) = \begin{pmatrix} x_B \\ y_B \\ z_B \\ \Delta t_B \end{pmatrix} \quad (3.5)$$

La première étape consiste à modifier les vecteurs pour qu'ils aient la même base de temps. Nous obtenons les vecteurs suivants :

$$\begin{cases} V'_A(t_{A_1}, t_{A_2}) = V_A(t_{A_1}, t_{A_2}) \times \Delta t_B \\ V'_B(t_{B_1}, t_{B_2}) = V_B(t_{B_1}, t_{B_2}) \times \Delta t_A \end{cases} \quad (3.6)$$

Une fois que nous avons nos deux vecteurs sur la même base de temps, la soustraction des deux vecteurs est réalisée. Le vecteur de mobilité du véhicule  $A$  par rapport à l'événement  $B$  est donc calculé comme cela :

$$V_{AB}(t_{A_1}, t_{A_2}, t_{B_1}, t_{B_2}) = \begin{pmatrix} (x_A \times \Delta t_B) - (x_B \times \Delta t_A) \\ (y_A \times \Delta t_B) - (y_B \times \Delta t_A) \\ (z_A \times \Delta t_B) - (z_B \times \Delta t_A) \\ \Delta t_A \times \Delta t_B \end{pmatrix} \quad (3.7)$$

Dans le cas d'un événement immobile  $C$ , le vecteur de mobilité est nul, quelque soit la base de temps considérée. L'application des précédentes formules nous donne donc le vecteur de mobilité du véhicule  $A$  de base comme vecteur de mobilité du véhicule  $A$  par rapport à l'événement  $C$ .

$$V_{AC}(t_{A_1}, t_{A_2}, t_{C_1}, t_{C_2}) = \begin{pmatrix} x_A - 0 \\ y_A - 0 \\ z_A - 0 \\ \Delta t_A \end{pmatrix} = V_A(t_{A_1}, t_{A_2}) \quad (3.8)$$

Comme nous le montrerons dans la suite, ces vecteurs nous aident à déterminer une probabilité de rencontre pour estimer si un véhicule a de fortes chances ou non de rencontrer un événement. Le calcul de cette probabilité de rencontre est décrit dans la section suivante.

### 2.2 Probabilité de rencontre

Dans cette section, nous introduisons d'abord le concept de probabilité de rencontre. Ensuite, nous montrons comment les différents paramètres dont elle dépend peuvent être choisis.

#### Définition et calcul

En utilisant le vecteur de mobilité du véhicule par rapport à l'événement, la position courante du véhicule et la position courante de l'événement, nous obtenons quatre valeurs dont dépend la probabilité de rencontre :

- ◇ La distance géographique minimale entre le véhicule et l'événement au cours du temps ( $\Delta d$ ).
- ◇ Le temps nécessaire au véhicule pour être le plus proche de l'événement ( $\Delta t$ ).
- ◇ L'âge de l'événement au moment où le véhicule sera le plus proche de l'événement ( $\Delta g$ ).
- ◇ L'angle entre le vecteur direction du véhicule et le vecteur direction de l'événement (représenté par un coefficient de colinéarité  $c$ ).

La représentation géométrique de  $\Delta d$  et  $\Delta t$  est donnée pour exemple dans la figure 3.5 (page 42). Dans cet exemple, nous avons pris un événement immobile pour une meilleure lisibilité mais la construction est identique avec un événement mobile après application d'un changement de référentiel (voir la figure 3.4 (page 38)).

Dans la figure 3.5 (page 42),  $B$  représente la position courante du véhicule,  $C$  la position courante de l'événement et  $\vec{AB}$  est le vecteur de mobilité du véhicule par rapport à l'événement. Avec ces informations qui sont soit internes au véhicule, soit incluses dans l'événement, nous pouvons déterminer le point  $D$  par construction de deux triangles rectangles, l'un d'hypoténuse  $[BC]$  et l'autre d'hypoténuse  $[AC]$ . En effet, nous avons le système d'équation suivant :

$$\begin{cases} |BC|^2 = |BD|^2 + |DC|^2 \\ |AC|^2 = |AD|^2 + |DC|^2 \end{cases} \quad (3.9)$$

Et nous pouvons le résoudre pour obtenir  $D$ . Pour cela, supposons que nous ayons :

$$|AD| = |AB| + |BD| \quad (3.10)$$

Cela signifie simplement que  $B$  est situé entre  $A$  et  $D$ . C'est-à-dire que le véhicule se dirige vers l'événement. Nous pouvons alors déduire :

$$\begin{cases} |BC|^2 = |BD|^2 + |DC|^2 \\ |AC|^2 = (|AB| + |BD|)^2 + |DC|^2 \end{cases} \quad (3.11)$$

$$\begin{cases} |BC|^2 - |BD|^2 = |DC|^2 \\ |AC|^2 - (|AB| + |BD|)^2 = |DC|^2 \end{cases} \quad (3.12)$$

$$|BC|^2 - |BD|^2 = |AC|^2 - (|AB| + |BD|)^2 \quad (3.13)$$

$$|BC|^2 - |BD|^2 = |AC|^2 - (|AB|^2 + 2 \times |AB| \times |BD| + |BD|^2) \quad (3.14)$$

$$|BC|^2 - |BD|^2 = |AC|^2 - |AB|^2 - 2 \times |AB| \times |BD| - |BD|^2 \quad (3.15)$$

$$|BC|^2 = |AC|^2 - |AB|^2 - 2 \times |AB| \times |BD| \quad (3.16)$$

$$2 \times |AB| \times |BD| = |AC|^2 - |AB|^2 - |BC|^2 \quad (3.17)$$

$$|BD| = \frac{|AC|^2 - |AB|^2 - |BC|^2}{2 \times |AB|} \quad (3.18)$$

Comme nous connaissons  $|AC|$ ,  $|AB|$  et  $|BC|$ , nous pouvons donc déduire  $|BD|$  et donc  $|DC|$  grâce à la précédente équation :

$$|BC|^2 = |BD|^2 + |DC|^2 \quad (3.19)$$

$$|DC| = \sqrt{|BC|^2 - |BD|^2} \quad (3.20)$$

$D$  est donc le point le plus proche de  $C$  sur la ligne  $(AB)$  et  $|DC|$  représente  $\Delta d$ , la distance géographique minimale entre le véhicule et l'événement au cours du temps.  $|BD|$  est la distance entre le véhicule et le point  $D$ . Puisque le vecteur de mobilité  $\vec{AB}$  contient une composante temporelle,  $|BD|$  peut être convertie en un temps pour obtenir  $\Delta t$ . Notons que dans le cas où le véhicule s'éloigne de l'événement, nous n'aurons pas l'hypothèse 3.10 (page 40) mais :

$$|AD| = |BD| - |AB| \quad (3.21)$$

Cela a pour seule conséquence de changer le signe de  $\Delta t$ . Autrement dit, dans le cas où le véhicule s'approche de l'événement, nous avons  $\Delta t > 0$  alors que dans le cas où il s'en éloigne, nous avons  $\Delta t < 0$ .

L'âge de l'événement ( $\Delta g$ ) est obtenu en utilisant le temps de génération contenu dans la position courante de l'événement. Cette valeur correspond à la durée écoulée entre le moment où

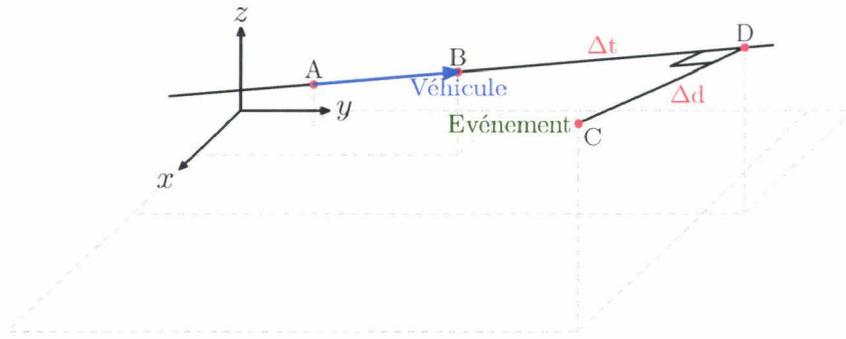


FIGURE 3.5 – Représentation géométrique de  $\Delta d$  et  $\Delta t$

l'événement a été généré et le moment où le véhicule sera le plus proche de l'événement (le point  $D$  dans la figure 3.5 (page 42)). Elle est donc calculée grâce aux informations liées à l'événement et à  $\Delta t$  calculé précédemment.

Finalement, comme mentionné précédemment, les véhicules sont capables d'estimer leur vecteur direction ainsi que le vecteur direction de l'événement. Ces deux vecteurs produisent un coefficient de colinéarité ( $c$ ) qui est une mesure de l'angle formé par ces vecteurs<sup>3</sup>. Pour les événements dépendants de la direction, cela permet de déterminer si les vecteurs directions du véhicule et de l'événement correspondent. Pour les événements indépendants de la direction (identifiés parce que la position de référence pour la direction est *nulle*),  $c$  est positionné à 0.

Une fois que les valeurs  $\Delta d$ ,  $\Delta t$ ,  $\Delta g$ , et  $c$  ont été calculées, elles sont utilisées pour estimer une « probabilité de rencontre » entre le véhicule et l'événement. La probabilité de rencontre (PR) est une valeur comprise entre 0% et 100%. Elle est obtenue en utilisant la fonction suivante, qui dépend des quatre valeurs précédentes :

$$PR = \frac{1}{\alpha \times \Delta d + \beta \times \Delta t + \gamma \times \Delta g + \zeta \times c + 1} \quad (3.22)$$

$$\alpha \geq 0, \beta \geq 0, \gamma \geq 0, \zeta \geq 0$$

où  $\alpha$ ,  $\beta$ ,  $\gamma$  et  $\zeta$  sont des coefficients de pénalité avec des valeurs  $\geq 0$ . Ils sont utilisés pour équilibrer l'importance relative des valeurs  $\Delta d$ ,  $\Delta t$ ,  $\Delta g$ , et  $c$ . Plus un coefficient est important, plus la valeur associée est pénalisante lors du calcul de la probabilité de rencontre. Par exemple, plus le coefficient  $\alpha$  est grand, plus la zone spatiale où l'événement est pertinent sera petite.  $\beta$  et  $\gamma$  sont utilisés pour que les informations les plus récentes et celles qui vont être rencontrées le plus tôt soient considérées en premier. Finalement,  $\zeta$  est utilisé pour pondérer l'importance du coefficient de colinéarité. Notons que si le véhicule s'éloigne du véhicule, nous avons  $\Delta t < 0$ . Lorsque ce cas

3. Nous obtenons ce coefficient en utilisant le théorème d'Al-Kashi dont la formule est :  $\widehat{ABC} = \arccos\left(\frac{AB^2 + BC^2 - CA^2}{2 \times AB \times BC}\right)$

est détecté, nous positionnons  $\Delta t$  à 0 et  $\Delta d$  prend alors la valeur de la distance actuelle. Au lieu que  $\Delta d$  soit la distance minimale au cours du temps, il s'agit alors de la distance à vol d'oiseau. De cette manière, le calcul de la probabilité de rencontre garde son sens, même pour les événements se trouvant derrière le véhicule.

Un véhicule réagit aux événements qu'il reçoit en fonction de la comparaison de la probabilité de rencontre avec deux seuils :

- ◊ Un seuil de pertinence indique la valeur minimale que doit avoir la probabilité de rencontre pour qu'un événement soit considéré comme pertinent. C'est-à-dire que la probabilité de rencontre est suffisante pour considérer que le véhicule va rencontrer l'événement.
- ◊ Un seuil de stockage indique la valeur minimale de la probabilité de rencontre pour laquelle l'événement est potentiellement pertinent pour le véhicule. L'événement est alors enregistré pour pouvoir être surveillé. L'évolution de la pertinence au cours du temps est surveillée par notre mécanisme d'évaluation continue, présenté dans le chapitre 4 (page 47).

Le seuil de stockage est donc plus petit que le seuil de pertinence. En définissant un seuil de stockage plus petit que le seuil de pertinence, le véhicule va enregistrer des événements qui ne sont pas pertinents au moment où ils ont été reçus mais qui pourraient devenir pertinents dans un futur proche (par exemple si le véhicule change sa direction).

### Choix des coefficients de pénalité

Les coefficients de pénalité  $\alpha$ ,  $\beta$ ,  $\gamma$  et  $\zeta$  sont utilisés pour pondérer l'importance relative des différentes valeurs. Considérés individuellement, ils nous permettent de définir des bornes sur la pertinence des événements. Par exemple, si le seuil de pertinence est positionné à 75%, prendre :

$$\alpha = \frac{1}{300} \quad (3.23)$$

implique que si la distance géographique minimale entre le véhicule et l'événement au cours du temps ( $\Delta d$ ) est plus grande que  $100m$ , alors l'événement ne sera pas considéré comme pertinent, quelques soient les valeurs des autres paramètres ( $\Delta d$ ,  $\Delta t$ ,  $\Delta g$  et  $c$ ).

En effet, nous pouvons déterminer la valeur des coefficients à partir d'une tolérance. En posant

$MaxD$ ,  $MaxT$ ,  $MaxA$  et  $MaxC$  comme étant les tolérances pour chacune des valeurs, nous avons :

$$0.75 = \frac{1}{(\alpha \times MaxD + 1)} \Rightarrow \alpha = \frac{1}{3 \times MaxD} \quad (3.24)$$

$$0.75 = \frac{1}{(\beta \times MaxT + 1)} \Rightarrow \beta = \frac{1}{3 \times MaxT} \quad (3.25)$$

$$0.75 = \frac{1}{(\gamma \times MaxA + 1)} \Rightarrow \gamma = \frac{1}{3 \times MaxA} \quad (3.26)$$

$$0.75 = \frac{1}{(\zeta \times MaxC + 1)} \Rightarrow \zeta = \frac{1}{3 \times MaxC} \quad (3.27)$$

De cette manière, une tolérance de 100 mètres pour  $\Delta d$  nous donne bien :

$$0.75 = \frac{1}{(\alpha \times 100 + 1)} \Rightarrow \alpha = \frac{1}{3 \times 100} \Rightarrow \alpha = \frac{1}{300} \quad (3.28)$$

De la même manière,  $\beta$  permet de fixer l'intervalle de temps maximum entre le temps actuel et le temps lorsque le véhicule sera le plus proche de l'événement. Si cet intervalle de temps est dépassé, l'événement n'est pas considéré comme pertinent. Par exemple, pour une valeur de  $\beta = \frac{1}{900}$ , un événement ne sera pas considéré comme pertinent s'il faut plus de 5 minutes au véhicule pour atteindre l'événement. De manière analogue,  $\gamma$  est utilisé pour pénaliser la pertinence en fonction de l'âge de l'événement. En pratique,  $\gamma$  doit être fixé en fonction de la fréquence à laquelle une nouvelle version des événements de longue durée est générée. Si cette période est de 5 minutes, alors nous pouvons fixer  $\gamma = \frac{1}{900}$ . Enfin,  $\zeta$  permet de fixer la tolérance maximale pour l'angle formé par les vecteurs directions des événements dépendant de la direction.

Naturellement, l'importance des paramètres  $\Delta d$ ,  $\Delta t$ ,  $\Delta g$  et  $c$  dépend de l'événement considéré. Par exemple, un message décrivant un embouteillage devrait être diffusé sur plusieurs kilomètres pour que les conducteurs puissent changer leur route. La pénalité sur  $\Delta t$  devrait donc être très petite. D'un autre côté, lorsqu'il s'agit d'une place de stationnement, la pénalité sur  $\Delta t$  devrait être beaucoup plus importante puisque les conducteurs cherchent des places de stationnement aussi proches que possible.

De plus, pour un même type d'événement, les coefficients de pénalité peuvent avoir à être modifiés en fonction de l'heure et de la date courante. En effet, si l'on considère des places de stationnement en environnement urbain, la pénalité sur l'âge ( $\gamma$ ) devrait être plus importante le samedi après-midi que pendant la nuit de lundi à mardi. La prise en compte du contexte pourrait donc permettre d'adapter les coefficients de pénalité. Bien que nous ne proposons pas de gestion du contexte pour faire cette adaptation, nous avons réalisé des simulations avec différentes configurations. Cela nous a permis de trouver des valeurs de coefficients pour lesquelles les résultats sont assez bons. Ces valeurs sont présentées dans le tableau 3.4 (page 45).

Configuration	$\alpha$	$\beta$	$\gamma$	$\zeta$
Accident sur autoroute	$\alpha = \frac{1}{3 \times 100}$	$\beta = \frac{1}{3 \times 300}$	$\gamma = 0$	$\zeta = \frac{1}{3 \times 50}$
Accident en centre ville	$\alpha = \frac{1}{3 \times 75}$	$\beta = \frac{1}{3 \times 50}$	$\gamma = 0$	$\zeta = \frac{1}{3 \times 100}$
Place de stationnement en centre ville	$\alpha = \frac{1}{3 \times 100}$	$\beta = \frac{1}{3 \times 150}$	$\gamma = \frac{1}{3 \times 200}$	$\zeta = 0$

TABLE 3.4 – Différentes valeurs des coefficients en fonction des configurations testées.

### 3 Conclusion

Nous avons présenté dans ce chapitre un mécanisme d'évaluation de la pertinence d'un événement pour un véhicule qui a été validé dans [DCI08, DCI10]. Notre solution permet de filtrer les messages pour ne garder que ceux qui sont intéressants pour les conducteurs. De plus, elle ne repose pas sur l'utilisation d'une carte, ce qui la rend beaucoup plus générale. En effet, l'utilisation d'une carte limiterait la solution aux seuls endroits connus. Bien qu'elle ne soit pas exacte, notre approche vectorielle reste précise puisque nous tenons compte non seulement de la direction mais aussi de la mobilité globale du véhicule. Contrairement aux autres solutions qui ne se basent que sur la distance et l'âge des événements [GION03, SWX05, WXYR05, WX04], notre solution peut traiter les événements mobiles et/ou dépendants de la direction. La représentation des événements que nous avons proposée est très simpliste mais elle permet de définir un ensemble d'attributs minimal permettant au système de fonctionner. Il est tout à fait possible d'enrichir cette représentation pour fournir, par exemple, du contenu multimédia [CDI09].

L'évaluation de la pertinence permet donc à chaque véhicule de se constituer une base de données locale avec les informations sur un minimum d'événements. Il est maintenant nécessaire d'interroger cette base de données pour pouvoir répondre aux demandes explicites et implicites des conducteurs. Le chapitre suivant présente notre mécanisme d'évaluation continue.



# 4

---

## Evaluation continue de requêtes

---

Dans ce chapitre, il s'agit pour nous d'exposer notre mécanisme permettant de traiter en continu les données reçues. En effet, au cours de son déplacement, le véhicule va recevoir des événements et les stocker localement. Ainsi, chaque véhicule se construit une base de données locale contenant un ensemble d'événements qu'il peut vraisemblablement rencontrer (voir la chapitre 3 (page 29). Etant donné que le véhicule se déplace et que le temps passe, la pertinence de chaque information reçue varie au cours du temps. Il est donc nécessaire de maintenir, au fil du temps, une base de données cohérente, c'est-à-dire dont les informations sont toujours à jour. Il faut donc que des informations qui étaient pertinentes dans le passé et qui ne le sont plus soit effacées. De plus, des informations qui n'étaient pas pertinentes peuvent le devenir. A cela s'ajoute les nouvelles demandes des conducteurs. Par exemple, pour un conducteur cherchant une place de stationnement, le système doit lui proposer les places de stationnement les plus pertinentes jusqu'à ce qu'il ait pu ranger son véhicule. Il faut donc, là encore, à chaque instant, chercher la meilleure place de stationnement dans la base de données locale. Il s'agit d'une évaluation continue d'une requête portant sur la proximité des places de stationnement précédemment reçues et stockées localement. Ainsi, chaque véhicule embarque un évaluateur de requête dont la tâche est d'évaluer de manière continue un

ensemble de requêtes permettant le bon fonctionnement du système en offrant au conducteur les meilleures informations. Nous proposons ici un mécanisme d'évaluation continue de requêtes adapté au contexte inter-véhiculaire. Les solutions actuelles se concentrant sur l'évaluation continue de requêtes s'appuient sur une architecture centralisée [MA08, MXA04]. Il s'agit d'un serveur contenant une base de données très importante et dont la tâche est de répondre à toutes les requêtes. Pour nous, il s'agit ici de proposer un mécanisme d'évaluation continue de requêtes embarqué dans les véhicules. Cette fois-ci, l'évaluateur se déplace et c'est là toute la particularité de notre contexte.

### 1 Définitions

Tout d'abord, quelques définitions s'imposent. Lorsque nous parlons d'évaluation continue, nous sommes dans le domaine des bases de données. Il s'agit donc ici de gérer un certain nombre de données. En effet, les informations partagées dans le réseau de communication inter-véhiculaire sont stockées par chaque véhicule localement dans une base de données. Cette base de données est composée de tables. Ces tables possèdent un certain nombre d'attributs. Les informations stockées dans ces tables doivent donc respecter le schéma de la table. C'est en s'appuyant sur ce schéma que les requêtes peuvent être posées. L'évaluation d'une requête, c'est le processus qui permet d'obtenir les résultats de cette requête. L'évaluation continue d'une requête, c'est le processus qui permet d'obtenir les résultats de cette requête et de mettre à jour ses résultats au cours du temps. Dans le cas où la base de données ne varie pas (les données qu'elle contient ne changent pas au cours du temps) et où le contexte de la requête ne change pas, il n'est pas nécessaire de faire de l'évaluation continue. En effet, les résultats de l'évaluation de la requête ne peuvent pas varier au cours du temps dans ce cas là. Par contre, dans le cas où la base de données varie au cours du temps, il est nécessaire de faire une évaluation continue. La base de données peut être modifiée par diverses opérations. Des données peuvent être ajoutées, supprimées ou mises à jour. Il est aussi nécessaire de faire de l'évaluation continue dans le cas où le contexte de la requête varie. Par contexte de la requête, nous entendons : les informations de l'évaluateur dont dépend la requête. En effet, une requête repose sur des informations qui ne sont pas forcément liées à la base de données. Prenons, par exemple, la requête « Quels sont les événements qui sont à moins de 200 mètres de moi ? » :

```
SELECT * FROM events WHERE distance() $<$ 200;
```

Cette requête dépend de ma position. Ma position n'est pas un événement et n'est pas stockée dans la base de données. C'est une information de l'évaluateur. L'évaluateur peut être vu comme la boîte noire qui s'occupe de répondre aux requêtes. Dans le cas où l'évaluateur subit des modifications au cours du temps, il est nécessaire de faire de l'évaluation continue si la requête considérée repose sur une donnée modifiable de l'évaluateur. Ces définitions étant posées, voyons maintenant nos besoins.

## 2 Besoins

La première question à se poser lorsque l'on doit évaluer des requêtes est de définir nos besoins. Pour pouvoir définir nos besoins, voyons d'abord les hypothèses de notre système.

### 2.1 Hypothèses

Pour le système d'information que nous proposons ici, les données sont reçues en continu au cours du déplacement du véhicule. De plus, les données sont éphémères et ne sont donc valides que pendant une période plus ou moins courte. C'est pourquoi, nous pouvons considérer que la base de données embarquée dans les véhicules ne contient aucune donnée au démarrage du système. Concernant les requêtes, un certain nombre sont prédéfinies au départ mais d'autres peuvent s'ajouter ensuite. Par exemple, dès le départ, il est intéressant pour le conducteur d'avoir une requête permettant d'être informé de tous les dangers présents sur la route et qu'il risque de rencontrer :

```
SELECT * FROM events WHERE EP()>70 AND priority>=10;
```

Par contre, concernant les places de stationnement, un conducteur ne voudra poser cette requête que lorsqu'il sera proche de sa destination :

```
SELECT * FROM events WHERE EventDescription="Parking" AND distance()<200;
```

Il s'agit donc pour notre évaluateur de requêtes de traiter ce genre de requêtes en partant des hypothèses que :

- ◇ La base de données interrogée est vide au départ ;
- ◇ Les ajouts et les mises à jour des données arrivent ponctuellement ;
- ◇ Des requêtes sont posées dès le départ et les résultats doivent rester valides tout au long de l'utilisation du système ;
- ◇ Des requêtes peuvent apparaître ponctuellement pour répondre à un besoin spécifique à un moment spécifique.

### 2.2 Besoins

En partant de ces hypothèses, nous voyons ce dont nous avons besoin. Dans un premier temps, il est nécessaire de prendre en considération toutes les modifications de la base de données. En considérant un certain nombre de requêtes posées, les résultats de ces requêtes doivent prendre en considération les ajouts, les suppressions et les mises à jour de données dans la base. Ensuite, compte tenu du déplacement du véhicule et donc, de l'évaluateur qu'il porte, ainsi que du temps qui passe, l'ensemble des résultats de l'évaluation d'une requête à un instant  $t$  perd de sa validité au cours du temps. Il est donc nécessaire de vérifier périodiquement la validité des résultats de chaque requête. Ainsi, il est nécessaire de faire de l'évaluation continue. Voyons maintenant les difficultés liées à ces besoins.

## 3 Difficultés

La manière la plus naïve de répondre à nos besoins est d'évaluer l'ensemble des requêtes sur l'ensemble des données avec une certaine fréquence. Dans l'idéal, cette fréquence devrait être aussi grande que possible. En effet, pour garantir des résultats cohérents à chaque instant, il faut que le temps entre deux évaluations successives soit aussi court que possible. Malheureusement, cette solution n'est pas réalisable puisqu'elle demande une utilisation du processeur beaucoup trop importante dès lors qu'il y a beaucoup de requêtes et beaucoup de données. Le processus d'évaluation continue constitue un vrai défi.

L'évaluation des requêtes ne pose pas de difficulté en soi. C'est le maintien de la validité des résultats qui est problématique. En effet, nous devons proposer à chaque instant les bons résultats en tenant compte des modifications de la base de données. Ces modifications interviennent plus ou moins fréquemment et il est donc nécessaire de proposer un mécanisme permettant de gérer ces modifications. Les véhicules se déplacent potentiellement à grande vitesse. Par conséquent, les requêtes dépendantes de la position du véhicule (et donc, de l'évaluateur qu'il porte) doivent être évaluées continûment en prenant en compte la mobilité de l'évaluateur. Les ressources étant limitées, nous devons donc proposer un mécanisme d'évaluation des requêtes qui puisse traiter nos requêtes continues avec un minimum de ressources. C'est pourquoi il est nécessaire de proposer des optimisations permettant de répondre aux problèmes posés.

## 4 Optimisations

Etant donné qu'une évaluation périodique des requêtes sur l'ensemble des données n'est pas envisageable, nous devons proposer une solution alternative permettant de garantir des résultats cohérents tout au long de l'exécution. Pour cela nous avons décomposé le problème en observant les différents aspects qui varient au cours du temps.

Le premier élément à prendre en compte est la modification de la base de données. En effet, la base de données est vide au départ et c'est pendant son déplacement que le véhicule va glaner des données. La première optimisation concerne donc l'évaluation incrémentale. Le principe n'est pas ici de réévaluer les requêtes sur l'ensemble des données mais de ne considérer que les modifications de la base de données. La solution naïve repose sur deux processus distincts :

- ◊ L'un gérant le contenu de la base de données ;
- ◊ L'autre gérant l'évaluation des requêtes.

Nous voulons, maintenant, considérer une interaction entre ces deux processus. Autrement dit, l'évaluation des requêtes fournit des résultats. Ces résultats sont considérés comme valides pendant toute la durée d'exécution des requêtes continues. Les modifications du contenu de la base de données activent, directement, une mise à jour des résultats des requêtes actives. De cette manière, toutes les modifications dans la base de données sont répercutées dans les résultats des requêtes continues. Ce mécanisme est explicité en détail dans la section 4.1 (page 52).

Le deuxième élément à prendre en compte est la modification de l'évaluateur de requête. Les requêtes sont posées sur un objet mobile et donc, sur un évaluateur mobile. Nos requêtes continues sont donc mobiles puisqu'elles dépendent de la position de l'évaluateur. Autrement dit, une même requête, évaluée sur deux véhicules différents, fournira deux ensembles de résultats différents. De même, la même requête évaluée à deux instants différents sur le même évaluateur fournira deux ensembles de résultats différents. Cela est simplement dû au fait que les requêtes que nous considérons sont de la forme :

```
SELECT * FROM TABLE [WHERE CONDITIONS AND/OR SPATIO_TEMPORAL_CONDITIONS]
[CONTINUOUSLY];
```

Dans cette requête, nous remarquons en particulier :

- **CONDITIONS** : représente les prédicats traditionnels. C'est-à-dire les prédicats usuellement utilisés dans les bases de données et portant sur les attributs des tables interrogées. Ces prédicats permettent d'identifier les informations présentes dans la base. Ils sont utilisés pour savoir, par exemple, si un événement possède une forte priorité ou encore si un événement concerne une place de stationnement ou un accident. Ces prédicats n'ont pas besoin de subir un traitement particulier puisque les informations sur lesquelles ils portent ne varient pas au cours du temps. L'évaluation continue ne concerne donc pas ces prédicats puisque leur évaluation donnera toujours le même résultat quelque soit le moment de l'évaluation et quelque soit l'évaluateur qui réalise l'opération.
- **SPATIO\_TEMPORAL\_CONDITIONS** : représente les prédicats particuliers concernant les informations spatio-temporelles des données. Ces prédicats permettent de définir des conditions sur les caractéristiques spatio-temporelles des données à l'instant où ils sont évalués et en prenant en considération les informations spatio-temporelles de l'évaluateur

qui les évalue. Ils sont donc utilisés pour connaître, par exemple, la proximité de l'événement par rapport à l'évaluateur où encore, pour nous, la probabilité que l'évaluateur rencontre l'événement. Le résultat de l'évaluation de ces prédicats est donc très fortement lié à l'évaluateur qui exécute l'opération ainsi qu'à l'instant où l'opération est réalisée.

- **CONTINUOUSLY** : représente le mot clef qui active l'évaluation continue.

Nous avons donc pour chaque requête un ensemble de prédicats dont l'évaluation dépend du véhicule qui l'évalue et du moment où cette requête est évaluée. Les résultats d'une même requête seront donc différents en fonction du véhicule qui l'évalue et du moment où il l'évalue. Pour gérer cela, nous utilisons un mécanisme qui prend en compte les modifications de l'évaluateur pour déclencher une réévaluation de certaines données. Ce mécanisme est présenté dans la section 4.2.

Pour la suite de ce chapitre, nous utiliserons la table des événements `Events` représentée par le tableau 4.1 pour donner des exemples.

Key	I	CurrentPos	DRP	MRP	Desc
<i>p1</i>	1	(153, 268, 75, 751)	-	-	Parking
<i>p2</i>	1	(207, 241, 73, 776)	-	-	Parking
<i>p3</i>	1	(214, 304, 73, 782)	-	-	Parking
<i>p4</i>	1	(126, 297, 74, 799)	-	-	Parking
<i>p5</i>	1	(268, 275, 78, 825)	-	-	Parking
<i>p6</i>	1	(242, 358, 75, 841)	-	-	Parking
<i>a1</i>	10	(338, 108, 70, 325)	(324, 87, 70, 323)	-	Accident
<i>d1</i>	10	(193, 273, 74, 804)	-	(232, 264, 73, 784)	Dangerous driver
<i>e1</i>	10	(148, 261, 75, 836)	(140, 250, 75, 834)	(91, 184, 75, 816)	Emergency vehicle

TABLE 4.1 – Exemple de table contenant des événements.

#### 4.1 Evaluation incrémentale

Les contraintes de mobilité dans le cadre des réseaux inter-véhiculaires imposent de considérer une évaluation de requêtes continue tant l'ensemble des résultats pour une requête est susceptible de changer rapidement. Avec une évaluation continue, les résultats d'une requête sont mis à jour en temps réel de manière à ce que l'utilisateur puisse toujours consulter l'ensemble des résultats mis à jour. Cela implique que toutes les modifications de la base d'informations (ajout, suppression, mise à jour) doivent être prises en compte dans le résultat de la requête continue. La difficulté réside ici dans la gestion des mises à jour des résultats afin de ne pas réévaluer systématiquement la totalité de la requête de manière périodique. Pour permettre cela, nous procédons en deux étapes. La première est l'évaluation initiale. La seconde consiste en le maintien de la cohérence des résultats.

### Evaluation initiale

Soit une requête  $r$ , par exemple « Quelles sont les places de parking libres que je peux rencontrer ? », nous pouvons l'exprimer de la manière suivante :

```
SELECT * FROM events WHERE EventDescription="Parking" AND EP()>70;
```

Elle correspond à la forme :

```
SELECT * FROM TABLE WHERE CONDITIONS AND SPATIO_TEMPORAL_CONDITIONS;
```

où CONDITIONS correspond à EventDescription="Parking"

et où SPATIO\_TEMPORAL\_CONDITIONS correspond à EP()<200.

Nous extrayons, dans un premier temps, une requête simplifiée  $r_s$  représentant la même requête que  $r$  mais en enlevant les critères spatio-temporels. Dans notre exemple, nous obtenons ainsi « Quelles sont les places de parking libres ? » :

```
SELECT * FROM events WHERE EventDescription="Parking";
```

Key	I	CurrentPos	DRP	MRP	Desc
$p_1$	1	(153, 268, 75, 751)	-	-	Parking
$p_2$	1	(207, 241, 73, 776)	-	-	Parking
$p_3$	1	(214, 304, 73, 782)	-	-	Parking
$p_4$	1	(126, 297, 74, 799)	-	-	Parking
$p_5$	1	(268, 275, 78, 825)	-	-	Parking
$p_6$	1	(242, 358, 75, 841)	-	-	Parking

TABLE 4.2 – Résultat de la requête : « Quelles sont les places de parking libres ? »

En évaluant  $r_s$ , nous récupérons les références (un ensemble d'identifiants) sur les résultats pouvant répondre à  $r$ . Soit  $S_{r_s}$  cet ensemble, le tableau 4.2 montre cet ensemble pour l'exemple que nous avons choisi.  $S_{r_s}$  contient bien tous les résultats de la requête  $r$  puisque  $r_s$  est moins sélectif que  $r$ . De cette manière, pour avoir les résultats de  $r$ , il n'est pas nécessaire d'évaluer l'ensemble des données mais seulement celles appartenant à  $S_{r_s}$ . Ce mécanisme permet de n'évaluer les critères spatio-temporels que sur les données pouvant effectivement répondre à la requête :

```
SELECT * FROM  $S_{r_s}$  WHERE SPATIO_TEMPORAL_CONDITIONS;
```

Key	I	CurrentPos	DRP	MRP	Desc
p4	1	(126, 297, 74, 799)	-	-	Parking
p6	1	(242, 358, 75, 841)	-	-	Parking

TABLE 4.3 – Résultat de la requête : « Quelles sont les places de parking libres que je peux rencontrer ? »

Dans notre exemple, il s’agit donc d’évaluer la requête :

SELECT \* FROM  $S_{r_s}$  WHERE EP() $>70$ ;

Cela permet d’avoir les résultats initiaux :  $S_r$  (représenté dans le tableau 4.3 pour notre exemple). Pour maintenir la cohérence des résultats, il suffit maintenant de prendre en compte les modifications de la base de données en les intégrant dans les ensembles  $S_{r_s}$ .

#### Maintien de la cohérence

Lorsque la base de données est modifiée, les résultats de la requête  $r$  doivent prendre en compte cette modification. En effet, au cours de son parcours, le véhicule peut recevoir de nouvelles informations, d’autres peuvent devenir obsolètes (par exemple lorsqu’une place de parking libre devient occupée). Pour les informations dynamiques, les nouvelles positions peuvent remettre en cause les résultats des requêtes. Le mécanisme permettant la prise en compte de ces modifications consiste à d’abord vérifier l’impact de la modification sur  $S_{r_s}$  puis sur  $S_r$ . Voyons ce qui se passe dans les différents cas possibles.

- **Ajout de données :** Dans le cas où la modification consiste en l’ajout d’une nouvelle donnée (par exemple l’apparition d’une nouvelle place de stationnement libre), l’évaluation de  $r_s$  sur cette donnée nous permet de savoir s’il y a effectivement un impact  $S_{r_s}$ . Les tableaux 4.4 et 4.7 montrent deux données qui peuvent être ajoutées au cours du déplacement du véhicule. L’évaluation de  $r_s$  sur la première donnée ajoutée permet de déterminer que  $S_{r_s}$  est modifiée. Le tableau 4.5 montre ce nouvel ensemble. En effet, il s’agit bien d’une place de parking. Nous évaluons alors les prédicats spatio-temporel de  $r$  sur la donnée ajoutée pour savoir si elle fait partie des résultats. Nous mettons ainsi à jour les résultats de la requête. Le tableau 4.6 montre le nouvel ensemble résultat.

Key	I	CurrentPos	DRP	MRP	Desc
p7	1	(212, 334, 76, 878)	-	-	Parking

TABLE 4.4 – Exemple 1 de donnée ajoutée pouvant modifier les résultats de la requête.

Key	I	CurrentPos	DRP	MRP	Desc
p1	1	(153, 268, 75, 751)	-	-	Parking
p2	1	(207, 241, 73, 776)	-	-	Parking
p3	1	(214, 304, 73, 782)	-	-	Parking
p4	1	(126, 297, 74, 799)	-	-	Parking
p5	1	(268, 275, 78, 825)	-	-	Parking
p6	1	(242, 358, 75, 841)	-	-	Parking
p7	1	(212, 334, 76, 878)	-	-	Parking

TABLE 4.5 – Nouvel ensemble  $S_{r_s}$ 

Key	I	CurrentPos	DRP	MRP	Desc
p4	1	(126, 297, 74, 799)	-	-	Parking
p6	1	(242, 358, 75, 841)	-	-	Parking
p7	1	(212, 334, 76, 878)	-	-	Parking

TABLE 4.6 – Résultat mis à jour de la requête : « Quelles sont les places de parking libres que je peux rencontrer ? »

Key	I	CurrentPos	DRP	MRP	Desc
a2	10	(539, 278, 72, 468)	(524, 287, 70, 464)	-	Accident

TABLE 4.7 – Exemple 2 de donnée ajoutée pouvant modifier les résultats de la requête.

Par contre, pour la donnée du tableau 4.7, l'information ne peut pas répondre à la requête puisqu'elle ne répond pas à  $r_s$ . Il n'y a donc aucune mise à jour de l'ensemble  $S_{r_s}$  et le résultat de la requête n'est pas modifié.

- **Suppression de données :** Dans le cas d'une suppression d'une information, aucune évaluation n'est nécessaire. Nous vérifions si  $S_{r_s}$  possède une référence sur cette information, si c'est le cas, alors, elle est supprimée de  $S_{r_s}$ . Elle est également retirée de l'ensemble des résultats de  $r$ . Le tableau 4.8 montre une information devant être supprimée de la base. L'ensemble  $S_{r_s}$  est alors mis à jour en conséquence (voir le tableau 4.9) et finalement, le tableau 4.10 présente le nouveau résultat de la requête.

Key	I	CurrentPos	DRP	MRP	Desc
p4	1	(126, 297, 74, 799)	-	-	Parking

TABLE 4.8 – Exemple de donnée supprimée de la base de données.

- **Mise à jour de données :** Par contre, dans le cas où il s'agit de la mise à jour d'une position d'un objet mobile, le problème est différent. La modification de l'ensemble des résultats d'une requête repose ici sur les critères spatio-temporels. Pour filtrer les critères spatio-temporels des requêtes à évaluer sur cette donnée mise à jour, nous vérifions dans un premier temps si l'ensemble  $S_{r_s}$  possède une référence sur la donnée qui est modifiée. Si tel est le cas, alors

Key	I	CurrentPos	DRP	MRP	Desc
<i>p1</i>	1	(153, 268, 75, 751)	-	-	Parking
<i>p2</i>	1	(207, 241, 73, 776)	-	-	Parking
<i>p3</i>	1	(214, 304, 73, 782)	-	-	Parking
<i>p5</i>	1	(268, 275, 78, 825)	-	-	Parking
<i>p6</i>	1	(242, 358, 75, 841)	-	-	Parking
<i>p7</i>	1	(212, 334, 76, 878)	-	-	Parking

TABLE 4.9 – Nouvel ensemble  $S_{r_s}$  après suppression de la donnée.

Key	I	CurrentPos	DRP	MRP	Desc
<i>p6</i>	1	(242, 358, 75, 841)	-	-	Parking
<i>p7</i>	1	(212, 334, 76, 878)	-	-	Parking

TABLE 4.10 – Résultat mis à jour de la requête : « Quelles sont les places de parking libres que je peux rencontrer ? » après suppression de la donnée.

la précédente évaluation des critères spatio-temporels sur cette donnée doit être remise en cause. Nous réévaluons donc les critères spatio-temporels sur la donnée mise à jour.

Supposons pour notre exemple la requête : « Quels sont les événements dangereux que je risque de rencontrer sur ma route ? »

```
SELECT * FROM events WHERE EP(>)>70 AND priority>=10;
```

Dans un premier temps, la première évaluation de cette requête nous permet d'obtenir l'ensemble  $S_{r_s}$  présenté dans le tableau 4.11.

Key	I	CurrentPos	DRP	MRP	Desc
<i>a1</i>	10	(338, 108, 70, 325)	(324, 87, 70, 323)	-	Accident
<i>d1</i>	10	(193, 273, 74, 804)	-	(232, 264, 73, 784)	Dangerous driver
<i>e1</i>	10	(148, 261, 75, 836)	(140, 250, 75, 834)	(91, 184, 75, 816)	Emergency vehicle

TABLE 4.11 – Ensemble  $S_{r_s}$  pour la requête « Quels sont les événements dangereux que je risque de rencontrer sur ma route ? »

De cet ensemble, nous avons extrait l'ensemble initial de résultat répondant à la requête (voir tableau 4.12).

Key	I	CurrentPos	DRP	MRP	Desc
<i>a1</i>	10	(338, 108, 70, 325)	(324, 87, 70, 323)	-	Accident
<i>e1</i>	10	(148, 261, 75, 836)	(140, 250, 75, 834)	(91, 184, 75, 816)	Emergency vehicle

TABLE 4.12 – Résultats initiaux de la requête « Quels sont les événements dangereux que je risque de rencontrer sur ma route ? »

Le tableau 4.13 montre une information que peut recevoir le véhicule pendant son dépla-

cement. Comme une information contenant déjà la même clef est présente dans la base de données, il s'agit bien d'une mise à jour. La nouvelle donnée reçue doit écraser la précédente.

Key	I	CurrentPos	DRP	MRP	Desc
<i>d1</i>	10	(291, 207, 74, 824)	-	(193, 273, 74, 804)	Dangerous driver

TABLE 4.13 – Exemple d'une information à mettre à jour

Pour savoir si la donnée doit être réévaluée, il s'agit donc dans un premier temps de vérifier si la clef de la donnée mise à jour est présente dans l'ensemble  $S_{r_s}$  de la requête. Dans notre exemple, c'est bien le cas. La mise à jour de la donnée risque donc de modifier l'ensemble des résultats de la requête. Nous réévaluons alors les critères spatio-temporels de la requête sur cette donnée mise à jour pour maintenir la cohérence de la base de données. Le tableau 4.14 donne l'ensemble de résultats après cette mise à jour dans le cas où la nouvelle évaluation ne rend pas l'information valide pour la requête.

Key	I	CurrentPos	DRP	MRP	Desc
<i>a1</i>	10	(338, 108, 70, 325)	(324, 87, 70, 323)	-	Accident
<i>e1</i>	10	(148, 261, 75, 836)	(140, 250, 75, 834)	(91, 184, 75, 816)	Emergency vehicle

TABLE 4.14 – Résultats mis à jour de la requête « Quels sont les événements dangereux que je risque de rencontrer sur ma route ? »

Grâce à ce mécanisme de prise en compte des modifications de la base de données, nous évitons de réévaluer toutes les requêtes sur toutes les données à chaque fois qu'une modification intervient dans la base de données. Cela permet de faire une évaluation continue plus efficace mais ce n'est pas encore optimal puisqu'il reste à prendre en compte le déplacement des véhicules. C'est-à-dire éviter de réévaluer l'ensemble des requêtes complètement à chaque fois que le véhicule se déplace. Nous présentons dans la suite le mécanisme permettant d'optimiser la prise en compte de la mobilité du véhicule.

## 4.2 Monitoring de l'évaluateur

La mobilité des véhicules rend l'évaluation des requêtes dépendantes de la localisation à un instant  $t$  de moins en moins pertinente au cours du temps. En effet, en se déplaçant, le véhicule va se rapprocher de certains événements et s'éloigner de certains autres rendant l'évaluation des requêtes de moins en moins cohérente avec la réalité. Pour palier à ce problème, les requêtes doivent être réévaluées. Cependant, réévaluer à chaque déplacement l'ensemble des requêtes sur l'ensemble des données n'est pas réalisable. La forte mobilité des véhicules a pour conséquence que le temps séparant deux évaluations successives n'est pas suffisant pour réaliser une réévaluation complète, même si l'on considère un faible nombre de requêtes et de données. C'est pourquoi, nous

avons mis en place un mécanisme permettant de ne réévaluer que certaines données pour chaque requête. Notre mécanisme utilise le déplacement du véhicule pour sélectionner ces données. Nous allons dans un premier temps présenter la représentation des résultats d'une requête dans notre système. Nous montrerons ensuite comment le déplacement du véhicule est pris en compte pour choisir les données qui doivent être réévaluées. Il s'agit donc bien ici de prendre en considération la mobilité du véhicule. Considérons donc la requête : « Quelles sont les places de parking libres qui sont à moins de 200 mètres ? » et supposons qu'un certain nombre de places de parking libres soit stocké dans la base. L'objectif est donc d'évaluer de manière continue cette requête en minimisant le nombre de données réévaluées à chaque instant.

L'évaluation des critères spatio-temporels des requêtes utilise l'ensemble  $S_{r_s}$  précédemment présenté. C'est l'évaluation des critères spatio-temporels sur cet ensemble qui permet de récupérer l'ensemble des résultats de la requête. Ce qui est problématique, c'est que cette évaluation doit être continue. En effet, la forte mobilité du véhicule remet en cause l'évaluation des critères spatio-temporels à chaque instant. Pour pouvoir limiter de manière pertinente le nombre d'éléments pour lesquels il faut réévaluer les critères spatio-temporels, certaines informations sont nécessaires. En effet, puisque nous voulons limiter le nombre d'éléments à réévaluer, il est nécessaire de bien choisir ces éléments. Pour cela, nous associons deux nouvelles valeurs à chaque donnée.

$\Delta_p$  : La distance entre la donnée et le prédicat spatio-temporel

Cette donnée permet de garder en mémoire le résultat de l'évaluation du prédicat spatio-temporel. Dans notre exemple, elle est calculée comme suit :

$$\Delta_p = 200 - distance() \quad (4.1)$$

De cette manière, les données dont  $\Delta_p$  est positive sont celles qui répondent à la requête alors que celles pour lesquelles la valeur est négative sont celles qui ne répondent pas à la requête. Cette valeur permet déjà de savoir que les données les plus sensibles sont celles dont la valeur est la plus proche de 0. En effet, pour ces données, un petit déplacement du véhicule risque de modifier le signe de cette valeur et donc l'appartenance ou non aux résultats de la requête. Pourtant, cette valeur n'est pas suffisante pour pouvoir sélectionner efficacement les données qui doivent être réévaluées. En effet, si l'on suppose une donnée très éloignée au départ, elle va donc être considérée comme n'étant pas très sensible aux déplacements du véhicule. Pourtant, au bout d'un certain temps et après plusieurs déplacements successifs, il est possible que le véhicule s'en rapproche suffisamment pour qu'elle rentre dans les résultats de la requête. Pour cette raison, nous utilisons une deuxième valeur qui va nous permettre de sélectionner les données à réévaluer plus efficacement.

$\Delta_e$  : La distance entre la donnée et le prédicat spatio-temporel en intégrant le déplacement du véhicule

Cette valeur a donc pour vocation de nous permettre de sélectionner les données à réévaluer. Il s'agit donc ici de garder une trace du déplacement du véhicule depuis le départ de la requête. Dans notre exemple, cette valeur est donc calculée comme suit :

$$\Delta_e = \text{distanceParcourue}() + \|\Delta_p\| \quad (4.2)$$

Cette valeur est mise à jour pour une donnée chaque fois que celle-ci est réévaluée de sorte que la distance parcourue associée soit bien celle depuis la dernière évaluation. De cette manière, plus le véhicule va se déplacer et réévaluer des données, plus ces données vont avoir une grande valeur pour  $\Delta_p$  alors que celles qui ne sont pas réévaluées vont garder leur  $\Delta_p$  et donc, être poussées en tête de la file d'attente pour la réévaluation.

#### Déroulement du mécanisme et exemple

Au départ, toutes les données sont donc évaluées et leurs  $\Delta_p$  et  $\Delta_e$  sont calculées. Le tableau 4.15 montre le résultat au départ. En appliquant l'équation 4.1 (page 58) pour chaque donnée, nous distinguons deux cas :

- $\Delta_p \leq 0$  : cela signifie que la distance entre l'évaluateur (le véhicule) et la donnée n'est pas conforme au prédicat de la requête.
- $\Delta_p > 0$  : cela signifie que la distance entre l'évaluateur (le véhicule) et la donnée est conforme au prédicat de la requête.

Grâce à un index sur  $\Delta_p$ , les résultats de la requête sont obtenus rapidement. Ils sont grisés dans le tableau 4.15.

Key	$\Delta_p$	$\Delta_e$
p1	45	45
p2	-168	168
p3	18	18
p4	96	96
p5	-12	12
p6	103	103
p7	-459	459

TABLE 4.15 – Résultat de l'évaluation initiale des  $\Delta_p$  et  $\Delta_e$  de toutes les données. Les lignes grisées représentent les données répondant à la requête ( $\Delta_p > 0$ ).

Ensuite, lorsque le véhicule se déplace, nous comparons ce déplacement avec  $\Delta_e$ . En effet,  $\Delta_e$  représente le déplacement minimal nécessaire pour que la conformité de la donnée par rapport à la requête puisse être remise en cause. C'est-à-dire que tant que le véhicule ne se déplace pas suffisamment, il n'est pas utile de réévaluer certaines données. Par exemple, si une donnée se trouve à 368 mètres du véhicule, cela signifie qu'elle est trop éloignée de 168 mètres pour correspondre à la requête. Tant que le véhicule ne s'est pas déplacé d'au moins 168 mètres, il est inutile de réévaluer la donnée. Grâce à  $\Delta_e$ , qui mémorise les distances minimales nécessaires à parcourir pour chaque donnée, nous pouvons sélectionner les données dont la distance parcourue depuis le lancement de la requête est supérieure à  $\Delta_e$ . Nous mettons alors à jour les données réévaluées. Le tableau 4.16 montre les données sélectionnées si le véhicule se déplace de 25 mètres. Grâce à un index sur  $\Delta_e$ , ces données sont obtenues dans un temps très court.

Key	$\Delta_p$	$\Delta_e$
p1	45	45
p2	-168	168
p3	18	18
p4	96	96
p5	-12	12
p6	103	103
p7	-459	459

TABLE 4.16 – Les cellules grisées correspondent aux données sélectionnées pour la réévaluation après un déplacement de 25 mètres.

Les données sont alors réévaluées et mises à jour. En appliquant l'équation 4.1 (page 58), nous obtenons les nouvelles valeurs de  $\Delta_p$ . Ces nouvelles valeurs sont calculées avec la nouvelle position du véhicule. Rien ne permet de prédire la valeur exacte de ces valeurs mais nous pouvons déterminer des bornes. En effet, si le véhicule s'est déplacé de  $x$  mètres, alors la variation de sa distance ( $\Delta_d$ ) par rapport à une donnée a varié d'une valeur comprise entre :

$$-x < \Delta_d < x$$

Il en va de même pour  $\Delta_p$  qui subit la même variation. Ainsi, pour l'élément p3 par exemple, nous avons :  $\Delta_p = 18$ . Après un déplacement de 25 mètres, nous pouvons dire que nous aurons :

$$18 - 25 < \Delta_p < 18 + 25$$

$$-7 < \Delta_p < 43$$

Nous voyons donc la possibilité que le signe de  $\Delta_p$  change. C'est pour cette raison que la donnée doit être réévaluée puisque c'est la positivité de  $\Delta_p$  qui détermine la validité de la donnée pour la requête. En appliquant l'équation 4.2 (page 59), p3 et p5 sont les seules données à avoir leur  $\Delta_e$  mis à jour. Le tableau 4.17 donne le résultat après l'évaluation de p3 et p5. Les  $\Delta_p$  sont bien mis à jour et nous pouvons retrouver les résultats de la requête (grisés dans le tableau).

Key	$\Delta_p$	$\Delta_e$
p1	45	45
p2	-168	168
p3	-3	$25 +   -3   = 28$
p4	96	96
p5	4	$25 +   4   = 29$
p6	103	103
p7	-459	459

TABLE 4.17 – Résultat de la requête après la réévaluation suite au déplacement de 25 mètres.

Lors d'un nouveau déplacement, nous prenons toujours la distance parcourue depuis le lancement de la requête pour la comparer avec  $\Delta_e$ . Ainsi, si le véhicule se déplace encore de 75 mètres, nous sélectionnons les données dont  $\Delta_e \leq 25 + 75$  mètres. Cela permet de prendre en compte tous les précédents déplacements. De plus, comme  $\Delta_e$  intègre la distance parcourue, dès qu'une donnée est réévaluée, son  $\Delta_e$  reste valide pour la comparaison avec la distance parcourue depuis le lancement de la requête. Le tableau 4.18 montre ces données. Elles sont alors réévaluées et nous pouvons obtenir les nouveaux résultats rapidement (tableau 4.19).

Key	$\Delta_p$	$\Delta_e$
p1	45	45
p2	-168	168
p3	-3	28
p4	96	96
p5	4	29
p6	103	103
p7	-459	459

TABLE 4.18 – Les cellules grisées correspondent aux données sélectionnées pour la réévaluation après un déplacement de 25+75 mètres.

Key	$\Delta_p$	$\Delta_e$
p1	104	$100 +   104   = 204$
p2	-168	168
p3	62	$100 +   62   = 162$
p4	157	$100 +   157   = 257$
p5	-52	$100 +   -52   = 152$
p6	103	103
p7	-459	459

TABLE 4.19 – Résultat de la requête après la réévaluation suite au déplacement de 25+75 mètres.

Chaque nouveau déplacement est ainsi incorporé et nous parvenons à maintenir les résultats valides au cours des déplacements du véhicule tout en minimisant le nombre d'éléments réévalués. Nous parvenons à assurer que toutes les données qui doivent être réévaluées le sont bien quand

elles le doivent. En effet, dans notre exemple, nous voyons que les données les plus à même d'être réévaluées sont, dans l'ordre, les données  $p_6$ ,  $p_5$ ,  $p_3$  et  $p_2$ . Alors que  $p_6$  était une donnée très éloignée par rapport aux autres au départ, elle devient maintenant celle qui va devoir être réévaluée en priorité. Notre mécanisme permet donc bien de réévaluer en priorité les données qui sont les plus proches de la limite du prédicat spatio-temporel.

## 5 Conclusion

Dans ce chapitre, il s'agissait pour nous de répondre à un besoin en terme d'interrogation des données de notre système. Le but ici était de permettre l'interrogation des données accumulées durant le trajet en continu pendant l'utilisation du système et donc, pendant cette accumulation de données.

Le fait que la base de données soit mise à jour pendant son interrogation nous imposait de proposer un mécanisme efficace d'évaluation continue des requêtes. Pour permettre cela, nous avons donc présenté un moyen de prendre en compte les ajouts, suppressions, mises à jour de données dans la base de données sans reposer sur une évaluation complète trop coûteuse en terme d'utilisation processeur [CD07]. Notre mécanisme repose sur l'utilisation d'une structure intermédiaire permettant de filtrer les résultats potentiels de chaque requête. Ainsi, nous avons une structure « pivot » qui permet de prendre en compte plus rapidement les modifications des données dans la base de données.

Nous nous sommes ensuite intéressés au second défi qui est la mobilité du véhicule. En effet, le véhicule se déplaçant constamment, les résultats des requêtes doivent être mis à jour régulièrement pour offrir une information pertinente. Là encore, nous nous sommes appuyés sur notre structure « pivot » en y intégrant des informations utiles, dépendantes de la requête considérée. Ces données nous permettent de déterminer rapidement quelles données sont susceptibles d'avoir changé d'état par rapport aux nouvelles informations de l'évaluateur.

Ces deux mécanismes combinés nous permettent donc de répondre aux contraintes du contexte en traitant aussi bien la dynamique des informations que la dynamique de l'évaluateur.

Après nous être intéressé à un mécanisme d'évaluation de la pertinence des informations et à un mécanisme d'interrogation des données, la dernière chose très importante à faire est de savoir comment les données sont échangées. Dans le chapitre suivant, nous examinons les contraintes en terme de réseau de notre contexte pour proposer un protocole permettant un échange efficace des informations.

# 5

---

## Dissémination de l'information

---

Jusqu'à présent, nous avons présenté les outils nécessaires à la gestion des informations dans notre système d'information. Nous avons donc présenté un mécanisme permettant de déterminer la pertinence d'une information. Nous nous sommes ensuite intéressés aux mécanismes permettant l'interrogation des données dans ce contexte. Autrement dit, nous avons décrit les processus de gestion des données à partir du moment où elles sont reçues jusqu'au moment où elles sont interrogées puis présentées au conducteur.

Il s'agit pour nous maintenant de considérer l'aspect communication de l'information et de déterminer comment les informations peuvent être reçues. En effet, pour que notre système puisse accomplir sa tâche, les informations relatives aux événements, une fois produites, doivent se répandre sur un ensemble de véhicules. Nous nous intéressons donc ici au protocole de dissémination. L'objectif est de faire en sorte que chaque information soit bien présente sur tout un ensemble de véhicules tout en minimisant le nombre de données échangées, cela pour limiter l'utilisation des ressources réseaux.

Nous proposons ici un mécanisme permettant de décider si une information doit être diffusée et si c'est le cas : dans quelle zone et par quels véhicules. La suite de ce chapitre est donc divisée

en plusieurs sections. Après avoir présenté l'intérêt d'un tel protocole, nous décrivons les différents points qui permettent de répondre à nos contraintes. La dernière section de ce chapitre est un peu particulière puisqu'elle traite du cas des ressources. En effet, pour les ressources, l'objectif est un peu différent puisqu'elles ne doivent pas être transmises à tous les automobilistes mais à un nombre limité de conducteurs. Une ressource ne pouvant être exploitée que par un seul véhicule, il est nécessaire de proposer une alternative au protocole de dissémination pour traiter ce cas.

### 1 Objectifs liés à la dissémination

L'objectif principal est de disséminer différents événements dans le réseau de communication inter-véhiculaire. Les événements que nous voulons traiter peuvent concerner des accidents, des places de stationnement libres, des véhicules d'intervention d'urgence, etc... Chaque événement possède sa propre zone de validité. Cette zone dépend donc de l'événement considéré mais aussi de l'endroit où l'événement a lieu. Le protocole de dissémination doit permettre de disséminer dans cette zone de validité. Même si le type d'événement donne une information intéressante pour définir la zone de validité de l'événement, cela n'est pas suffisant pour la définir suffisamment précisément. En effet, prenons quelques exemples :

- ◇ L'information sur un événement immobile, dépendant de la direction doit être propagée dans le sens de circulation de l'événement, derrière lui. Par exemple, l'information sur un accident sur autoroute doit permettre d'informer les conducteurs qui arrivent sur le lieu de l'accident. Cette information doit remonter suffisamment loin pour donner l'opportunité à ces conducteurs de prendre une route alternative et ainsi éviter de provoquer un embouteillage. Par contre, pour un freinage d'urgence, le type d'information est le même mais la zone de validité est différente. Il n'est pas nécessaire de remonter l'information aussi loin. Ici, il s'agit surtout d'éviter un sur-accident et donc, seuls les véhicules qui sont à quelques centaines de mètres doivent recevoir l'information.
- ◇ L'information sur un événement immobile, indépendant de la direction doit être propagée non plus dans un sens de circulation particulier mais tout autour de l'événement. La zone de validité d'une place de stationnement libre est un cercle autour de la place considérée. Tous les véhicules proches doivent posséder l'information puisqu'elle peut intéresser les conducteurs cherchant à se garer. Par contre, il n'est pas forcément nécessaire de communiquer l'information à des véhicules très éloignés puisque d'une part, la place pourra être trop loin pour intéresser les conducteurs et ensuite parce qu'elle risque d'être occupée avant d'être atteinte.
- ◇ Concernant les événements mobiles, les mêmes distinctions se font. Chaque événement nécessite d'être relayé plus ou moins loin de l'événement et dans des sens de circulation spécifiques. Par exemple, une ambulance est un événement mobile. Les véhicules à prévenir sont ceux qui se trouvent devant elle et qui peuvent la gêner. La zone de validité se trouve

donc devant l'ambulance et se déplace avec elle. Par contre, un convoi exceptionnel particulièrement lent et encombrant sur la route est un événement mobile mais les véhicules qui doivent être informés sont ceux qui se trouvent derrière et qui risquent de le rattraper.

Ainsi, pour chaque événement la zone de dissémination est spécifique. Chaque véhicule entrant dans une zone de dissémination d'un événement doit le recevoir et chaque véhicule sortant d'une zone de dissémination d'un événement doit le supprimer.

En plus de la zone de validité des événements, il est important de considérer les mises à jour des événements. En effet, les événements mobiles sont des événements dont la position change constamment au cours du temps. Il est donc nécessaire de mettre à jour ces événements sur l'ensemble des véhicules les possédant. Nous utilisons aussi des mises à jour d'événements pour les événements dont la durée de validité est très longue. En effet, les événements ont des durées de validité différentes. Dans l'idéal, la durée de validité d'un événement devrait être communiquée avec l'événement mais cela suppose que cette durée est connue a priori ce qui n'est évidemment pas le cas. Pour remédier à cela, l'une des possibilités est d'utiliser des messages d'invalidation. Chaque événement serait alors considéré comme valide jusqu'à la réception d'un message d'invalidation.

Malheureusement, l'environnement dans lequel nous nous trouvons nous empêche de mettre en place une telle solution. En effet, la forte mobilité des véhicules ne permet pas de garantir que chaque véhicule possédant une information va effectivement recevoir le message d'invalidation. Pour répondre à ce problème, nous utilisons donc les mises à jour d'événements. Chaque événement possède une durée de validité assez courte. Dans le cas où aucune mise à jour n'est reçue, l'événement est invalidé à la fin de cette période. Les mises à jour permettent de prolonger la durée de validité des événements qui le nécessitent.

Ainsi, notre mécanisme de dissémination doit donc permettre de diffuser les messages dans une zone spécifique et pendant un certain temps. Notre solution peut-être vue comme de l'épidémiologie. Il faut contaminer un certain nombre de véhicules dans une certaine zone et pendant un certain temps. L'objectif est de contrôler la duplication des données en limitant le nombre de véhicules possédant l'information. Les moyens disponibles sont les communications sans fil tels que les réseaux 802.11. La question ici est de savoir comment utiliser ces réseaux pour contaminer un ensemble spécifique de véhicules en prenant en compte les contraintes de ces réseaux.

En effet, une fois cette zone de dissémination définie, nous devons réaliser cette dissémination. Là encore, de nombreuses contraintes compliquent la tâche. Tout d'abord, les portées de communication sont limitées. Nous devons donc prendre en considération le fait que chaque véhicule ne peut communiquer qu'avec un petit ensemble de véhicules. Ensuite, la quantité d'informations transportable sur ces réseaux n'est pas illimitée. Nous devons bien choisir les données à échanger tant les contraintes en bande passante sont importantes. Bien que ces contraintes soient purement technologiques, nous ne pouvons nier leur existence.

Finalement, l'utilisation des réseaux sans fil implique une incertitude dans la réception des messages. La propagation des ondes est fortement influencée par l'environnement. Un certain nombre de messages ne sont donc pas reçus. Les communications sont incertaines et il est donc nécessaire de garantir, autant que possible, que les messages diffusés soient bien réceptionnés. La perte des messages est d'autant plus importante que nous sommes dans un environnement fortement mobile. Certains phénomènes physiques dégradent la qualité des messages émis. Tout d'abord, si plusieurs messages sont émis dans une même zone, chaque message provoque un bruit pour les autres. Ensuite, notre environnement est sujet à l'effet Doppler. En effet, lorsque la distance entre l'émetteur et le récepteur d'une onde varie au cours du temps, la fréquence de l'onde à la réception est différente de la fréquence lors de l'émission. Autrement dit, si l'on considère des véhicules circulant en sens contraire, leur distance varie grandement au cours du temps. Si l'on prend un véhicule comme référentiel, la vitesse de l'autre est la somme des deux vitesses. Et par conséquent, le message émis par un véhicule sera reçu avec une fréquence différente par le véhicule récepteur. Cet effet peut donc empêcher la bonne réception de certain message, surtout dans le cas de forte mobilité en sens contraire. Ainsi, bien que les réseaux sans fil permettent d'imaginer la communication entre les véhicules, cette communication n'est pas sans poser de réels défis.

## 2 Protocole de dissémination

C'est pourquoi, nous avons travaillé sur un protocole de dissémination. Ce protocole doit prendre en compte toutes les contraintes liées à l'utilisation de réseaux sans fil dans le domaine inter-véhiculaire pour permettre un échange efficace des données à travers les véhicules. Notre protocole repose sur la mise en place de différentes techniques permettant le transfert des informations en utilisant les ressources réseaux mises à notre disposition. Pour cela, nous nous sommes d'abord intéressés aux différents modes de communication. Cela est discuté en premier lieu dans la suite. Nous avons ensuite déterminé les zones de validité des événements en utilisant la probabilité de rencontre. Enfin, nous avons défini un mécanisme de dissémination à l'intérieur des zones de validité des événements permettant de maintenir l'information dans cette zone spatio-temporelle.

### 2.1 Unicast, Multicast ou Broadcast ?

Pour disséminer les messages sur le réseau, plusieurs possibilités s'offrent à nous. En effet, en utilisant les moyens de communication sans fil des terminaux mobiles que nous considérons (PDA, SmartPhone, ...), plusieurs modes de communication existent. Nous reposons donc sur l'utilisation du WiFi et sur le protocole IP pour le transport des données. Dès lors que cela est posé, nous devons choisir le mode de communication.

Le premier mode que nous pouvions utiliser est la communication point-à-point : l'unicast. C'est-à-dire échanger des messages entre les véhicules deux à deux. Cela suppose donc la création

d'un canal de communication pour chaque couple de véhicules. Ensuite, les messages peuvent transiter d'un véhicule à l'autre en utilisant les canaux ouverts. Cette méthode repose donc sur une communication spécifique entre véhicules. Il faut donc connaître le ou les véhicules destinataires des messages et envoyer chaque message à chaque véhicule. Cette solution permet donc de limiter le nombre de véhicules recevant l'information. De plus, l'utilisation du protocole TCP permet de garantir la bonne réception des messages. Ce moyen de communication est donc très intéressant pour créer un dialogue entre deux véhicules. Malheureusement, un grand nombre de messages ont besoin d'être échangés si l'on veut informer un grand nombre de véhicules sur un événement et les temps nécessaires à l'établissement de la communication sont très pénalisants. De plus, pour nous, l'information sur un événement ne doit pas être communiquée à des véhicules spécifiques n'importe où mais au contraire à tous les véhicules dans une certaine zone. Cette solution n'est donc pas vraiment adaptée à notre contexte.

Le deuxième mode de communication que nous pouvions utiliser est le mode multicast. Avec ce mode, les terminaux s'inscrivent à des groupes représentés par des adresses IP spécifiques. Ensuite, les messages sont envoyés à une adresse pour atteindre tous les terminaux s'étant inscrits au groupe. Ce mode de communication permet donc d'atteindre un grand nombre de véhicules avec un nombre limité de messages. Le nombre de messages envoyés est donc réduit puisqu'un seul message est reçu par tout un ensemble de véhicules. Par contre, il n'est plus possible ici d'utiliser le protocole TCP. Nous devons utiliser le protocole UDP qui consiste à construire des paquets et à les envoyer simplement. Il n'y a donc aucune garantie que tous les véhicules du groupe aient bien reçu le message. Cela peut être gênant, surtout si l'on considère des messages concernant des événements sur des freinages d'urgences par exemple. L'utilisation de ce mode de communication pose donc de nouveaux challenges concernant la garantie que les messages soient bien reçus. De plus, même si l'utilisation des groupes est très utile pour envoyer des messages à plusieurs véhicules, la constitution de ces groupes n'est pas évidente. Nous pourrions définir un groupe pour chaque événement. Chaque groupe serait défini en fonction de la zone de validité de l'événement pour lequel il a été créé. Par contre, cela est très compliqué puisque cela suppose, pour chaque véhicule, la connaissance de l'existence des groupes et donc des événements. Autrement dit, pour rejoindre un groupe, il faudrait connaître l'existence de l'événement mais pour connaître l'existence d'un événement, il faudrait avoir rejoint le groupe. La force de cette solution est donc la possibilité de communiquer une information à un ensemble de véhicules. Les difficultés sont donc l'établissement des groupes et la garantie que les messages soient bien reçus.

Enfin, nous pouvions utiliser le mode broadcast. Comme pour le mode multicast, il s'agit ici d'envoyer des paquets sur le réseau. Par contre, ici, les paquets ne sont pas reçus par un groupe de véhicules mais par tous les véhicules dans le rayon de communication de l'émetteur et faisant partie du même réseau. Il ne s'agit donc plus ici de créer des groupes mais simplement d'émettre des messages sur le réseau. Comme pour le mode multicast, c'est le protocole UDP qui est utilisé. Il n'y a donc aucun moyen de s'assurer que les messages sont reçus. L'utilisation de ce mode de communication est donc un moyen de déplacer le problème. Cela ne permet pas de répondre au problème de définition des zones de validité. Cela ne permet pas de garantir que les messages sont bien reçus. Cela permet simplement de diffuser des messages autour du véhicule.

Nous avons choisi d'utiliser le troisième mode de communication. En effet, c'est ce mode qui correspond le mieux au principe de dissémination dont nous avons besoin. D'une part parce qu'il permet à chaque véhicule d'atteindre les véhicules voisins simplement. D'autre part parce qu'il ne nécessite pas la mise en place complexe de groupe de communication. Autrement dit, il permet de déplacer la problématique de création des zones de dissémination et la garantie que les messages soient reçus. En effet, en utilisant ce mode de communication, la zone de validité n'est plus dépendante de l'aspect communication mais repose plutôt sur une analyse des événements. La suite de cette partie expose comment cette zone de validité est déterminée. Ensuite, pour s'assurer que les messages soient bien reçus dans cette zone de validité, ce n'est pas non plus l'aspect réseau qui s'en occupe mais une bonne gestion des messages. Nous exposerons notre principe de dissémination permettant de garantir, autant que possible, la bonne réception des messages.

### 2.2 Zone de dissémination des événements

La zone de dissémination d'un événement correspond à sa zone de la validité. Il est important de diffuser l'événement là où il est valide et il est inutile de diffuser l'événement là où il ne l'est pas. Pourtant, déterminer cette zone est très complexe puisque cela dépend de l'événement mais aussi de l'environnement<sup>1</sup>. Cette zone pourrait donc être déterminée en analysant des données cartographiques. Cependant, n'ayant pas considéré l'utilisation de telles données, cette solution n'est pas possible pour nous. De plus, bien que l'utilisation de données cartographiques peut sembler, au premier abord, la solution la plus évidente, elle est loin d'être triviale. L'analyse des cartes n'est pas une chose aisée et il resterait toujours très délicat de définir cette zone. De plus, si cette zone était définie de cette façon, elle serait déterminée une fois pour toute et les variations seraient difficilement envisageables, particulièrement pour les événements mobiles.

Pourtant, la principale hypothèse sur la zone de dissémination nous donne le premier pas vers la solution. Puisque la zone de dissémination d'un événement correspond à sa zone de validité, pourquoi ne pas utiliser le mécanisme d'évaluation de la validité d'un événement pour définir cette zone ? Pour nous, il s'agit donc d'utiliser la probabilité de rencontre pour diffuser l'information sur un événement dans la bonne zone.

En effet, l'utilisation de la probabilité de rencontre pour déterminer quels véhicules doivent rediffuser une information qu'ils ont reçue permet de diffuser les messages dans la bonne direction. C'est-à-dire dans la direction où les événements peuvent être pertinents. Le principe est de dire que si une information est pertinente pour un véhicule, elle a de fortes chances de l'être pour ses voisins. Si l'on considère le moment où l'événement est généré, les véhicules proches de l'événement vont recevoir l'information du générateur de l'événement. Ensuite, parmi ces véhicules, certains vont considérer l'événement comme étant pertinent. Par exemple, dans le cas d'un accident, les véhicules arrivant sur les lieux de l'accident et dans le sens de circulation vont considérer l'information comme

---

1. L'environnement regroupe un grand nombre d'éléments dont la configuration des routes, la densité de véhicules, la vitesse, etc. . .

pertinente. Ils vont alors pouvoir rediffuser l'information et atteindre des véhicules plus éloignés. A l'inverse, les véhicules se trouvant devant l'accident ne vont pas considérer l'information comme pertinente. L'accident se trouvant derrière eux, la probabilité de rencontre sera faible pour eux. Ils ne vont donc pas rediffuser l'information et elle ne va pas avancer devant l'accident, là où elle n'a aucune chance d'être pertinente pour un véhicule.

L'utilisation de la probabilité de rencontre permet aussi d'assurer qu'une information est maintenue, durant la phase de dissémination, suffisamment proche de l'événement pour que ce soit pertinent. Par exemple, une information concernant une place de stationnement disponible ne peut pas être intéressante pour les personnes roulant quelques kilomètres plus loin. C'est pourquoi, chaque fois qu'un véhicule reçoit un événement, il calcule la probabilité de rencontrer cet événement. Si cette valeur est supérieure à un certain seuil de diffusion, le véhicule doit rediffuser l'information pour atteindre des nouveaux véhicules. Sinon, il ne considère pas le message. De cette manière, tant qu'un événement est considéré comme pertinent par un véhicule, le message est relayé aux véhicules voisins et ainsi de suite.

$$PR = \frac{1}{\alpha \times \Delta d + \beta \times \Delta t + \gamma \times \Delta g + \zeta \times c + 1} \quad (5.1)$$

De plus, la probabilité de rencontre (voir la fonction 5.1 (page 69)) permet d'éviter la dissémination des événements obsolètes. L'information diffusée pour un événement tel qu'une place de stationnement ou un freinage d'urgence n'est pertinente que pendant une courte durée. Pour les événements ayant une durée de vie plus longue (un accident par exemple), il est possible d'adapter la valeur du coefficient de pénalité correspondant ( $\gamma$ ) de manière à ne pas trop pénaliser l'âge de l'événement. De plus, pour les événements de longue durée, de nouvelles versions sont générées pour continuer à prévenir les véhicules arrivant. Ainsi, le mécanisme de dissémination proposé ne nécessite pas de message indiquant la fin d'un événement. La zone de dissémination, qui est une zone spatio-temporelle, est donc définie dynamiquement grâce à la probabilité de rencontre. En fonction du temps et de la position de l'information, elle sera ou non relayée. Cela permet ainsi de maintenir aussi longtemps que nécessaire une information dans une zone spatiale qui diminue ou se déplace au cours du temps, en fonction de l'événement.

### 2.3 Economie de la bande passante

Maintenant que la zone de dissémination est définie, la question est de savoir comment diffuser l'information dans cette zone. La solution naïve serait de faire de l'inondation. Si l'on considère que chaque véhicule pour lequel la probabilité de rencontre est assez élevée doit diffuser l'information, alors en théorie, l'information est diffusée dans une zone et de proche en proche, grâce au multi-saut, l'information devrait atteindre l'ensemble de véhicules voulu. Malheureusement, cette solution consomme beaucoup de bande passante puisque si de nombreux véhicules sont proches les uns des autres, alors de nombreux messages vont être échangés dans une même portée de communication.

Comme les communications sans fil sont sensibles aux bruits, la perte de messages risque d'être si importante que l'information sera perdue. De plus, même si l'on ne considère aucune perte de messages à cause de problème de réseau, il pourra y avoir des pertes à cause des différentes densités de véhicules. En effet, si à un moment donné, il n'y a pas de véhicule pour recevoir l'information des véhicules précédents, alors l'information sera perdue. Pour éviter cela, il est donc nécessaire non pas de faire de l'inondation mais de l'inondation périodique. C'est-à-dire imposer à chaque véhicule de diffuser l'information avec une certaine fréquence. Cela afin d'éviter qu'à un moment donné, aucun véhicule ne reçoit l'information et donc, qu'aucun véhicule ne la relaye ensuite. Evidemment, si l'inondation provoque une congestion au niveau réseau, ce phénomène est encore pire avec l'inondation périodique. Autrement dit, nous devons trouver un autre moyen de disséminer une information dans une zone.

Pour éviter de faire de l'inondation, nous proposons un mécanisme qui désynchronise les rediffusions des différents véhicules. Lorsqu'un véhicule diffuse une information, les véhicules suffisamment proches vont recevoir l'information. Avec la solution naïve, tous ces véhicules vont diffuser l'information immédiatement après l'avoir reçue. Nous voulons désynchroniser toutes ces rediffusions pour éviter que tous rediffusent au même moment dans un espace réduit. En effet, puisque la valeur de la probabilité de rencontre peut être élevée pour tous les véhicules ayant reçu une information, tous vont vouloir la diffuser à leur tour. Or, il est nécessaire de limiter le nombre de diffusions d'un même message. C'est pourquoi, chaque véhicule recevant une information et considérant qu'il doit la rediffuser (c'est-à-dire que la probabilité de rencontre est supérieure au seuil de diffusion) va attendre un certain temps  $t$  avant de la diffuser effectivement. Pour déterminer cette période, nous prenons en compte la distance entre le véhicule par lequel l'information a été reçue et la distance du véhicule qui veut la rediffuser. Cette période est donc différente pour tous les véhicules recevant l'information puisque tous ont une distance différente par rapport au dernier émetteur de la donnée. La question à laquelle nous devons répondre est alors de savoir quels sont les véhicules qui doivent diffuser l'information en premier. Autrement dit, comment utiliser cette distance pour déterminer le temps d'attente. Les deux possibilités étaient soit de considérer le temps d'attente comme étant proportionnel à la distance, soit de le considérer inversement proportionnel. Dans le premier cas, cela signifie que plus la distance est grande, plus le temps d'attente doit être important. Dans le second cas, cela signifie que plus la distance est grande, plus le temps d'attente doit être petit. Intuitivement, le choix doit se faire en fonction du nombre de nouveaux véhicules que chacun peut contaminer. Puisque la dissémination se fait de manière omnidirectionnelle, la zone de réception des messages est une sphère autour du véhicule émetteur. Il faut donc choisir le véhicule dont cette sphère possède le moins d'intersection avec la sphère de l'émission précédente. Il est évident qu'il s'agit du véhicule le plus éloigné. Plus un véhicule est proche de l'émetteur d'un message, plus ses voisins risquent d'être les mêmes. Le meilleur candidat pour la rediffusion est donc le véhicule le plus éloigné du précédent émetteur. Le temps d'attente avant rediffusion doit donc être inversement proportionnel à la distance par rapport au véhicule émetteur précédent. Nous utilisons pour cela la formule suivante :

$$t = D \times \left(1 - \frac{d}{r}\right) \quad (5.2)$$

où  $D$  est le temps maximum d'attente avant rediffusion,  $r$  est la portée de communication (déterminée en fonction du moyen de communication utilisé, entre 200 et 400 mètres pour les communications inter-véhiculaires), et  $d$  correspond à la distance entre le véhicule émetteur et le véhicule receveur. La distance  $d$  est une valeur comprise entre 0 et  $r$  puisque, théoriquement, la distance entre les deux véhicules ne peut être supérieure à la portée de communication. Dans le cas où la distance  $d$  serait supérieure à la portée estimée  $r$ , nous considérons que  $d = r$ . De cette manière,  $\frac{d}{r}$  nous donne une valeur comprise entre 0 et 1. Cette valeur est proportionnelle à la distance entre les deux véhicules mais,  $1 - \frac{d}{r}$  nous donne bien une valeur comprise entre 0 et 1 inversement proportionnelle à la distance. Ainsi,  $D \times (1 - \frac{d}{r})$  nous donne bien une valeur comprise entre 0 et  $D$ . Cette valeur est bien inversement proportionnelle par rapport à la distance. Plus un véhicule est éloigné, plus il va diffuser rapidement.

Cette approche permet de propager un message loin de son origine si cela est nécessaire rapidement. Notons ici qu'une alternative pour choisir le véhicule qui doit rediffuser l'information serait de mettre en place une élection. Par exemple, les véhicules qui reçoivent l'information en informent l'émetteur puis il choisit celui qui doit rediffuser. Ce choix pourrait être fait en fonction de la distance et ainsi, le plus éloigné serait élu. Malheureusement, cette idée n'est pas réalisable dans la pratique. Les temps de communication étant très courts, l'établissement de cette élection n'est pas possible. L'utilisation d'une fonction indépendante des véhicules alentours permet d'accélérer la prise de décision. Pour déterminer si un véhicule doit rediffuser ou non, il pourrait aussi être intéressant de faire du monitoring des voisins. En fonction du nombre de voisins n'ayant pas reçu l'information, la décision pourrait être prise de rediffuser ou non. Là encore, les contraintes en bande passante et en temps de communication invalident cette possibilité. Dans le contexte que nous considérons, un monitoring est impossible. C'est pourquoi, bien que l'utilisation de notre temps d'attente n'est pas optimale et ne permet pas avec certitude de déterminer le meilleur candidat pour la rediffusion, c'est le meilleur compromis.

Enfin, le fait d'inclure un temps d'attente permet aussi de limiter le nombre de rediffusions. En effet, puisque les véhicules rediffusent les mêmes messages avec des temps d'attentes différents, la réception de deux messages successifs concernant le même événement est utilisée comme un accusé réception. Autrement dit, après la réception d'un message, le véhicule entre dans son temps d'attente. Pendant ce temps, s'il n'est pas le plus éloigné, il va recevoir l'information du véhicule le plus éloigné puisque ce dernier va rediffuser l'information avant lui. La réception de cette information est utilisée comme accusé réception et le véhicule peut alors annuler sa rediffusion. Ainsi, recevoir deux fois un événement est vu comme une confirmation qu'il a bien suivi sa chaîne de dissémination. Pour s'assurer que ce n'est pas le même véhicule qui aurait diffusé plusieurs fois le même événement, nous utilisons un Numéro de saut. Ce numéro est incrémenté à chaque fois qu'un nouveau véhicule rediffuse une information. De cette manière, lorsqu'un véhicule reçoit un événement avec un numéro de saut  $x$ , il attend le saut numéro  $x + 1$  pendant son temps d'attente. S'il le reçoit, il annule sa rediffusion. Sinon, il rediffuse ce saut numéro  $x + 1$  et attend le saut numéro  $x + 2$ . Ce mécanisme est détaillé dans la suite.

Pour finir, n'oublions pas qu'il peut arriver qu'aucun véhicule ne considère une information

reçue comme pertinente ou bien qu'aucun véhicule ne soit présent pour recevoir l'information et la relayer. Cela pourrait avoir comme conséquence la perte du message. Pour répondre à ce problème, nous utilisons de la même manière notre temps d'attente et notre numéro de saut. Chaque fois qu'un véhicule diffuse une information avec un numéro de saut  $x$ , il attend pendant un temps  $D$  l'information avec le numéro de saut  $x + 1$ . S'il la reçoit, c'est que la chaîne s'est poursuivie normalement et il n'a pas besoin de continuer de diffuser l'information. Par contre, s'il ne la reçoit pas, c'est que soit aucun véhicule n'a reçu l'information, soit aucun véhicule ne l'a considérée comme pertinente. Dans ce cas, au bout du temps d'attente  $D$ , il rediffuse l'information. Cela permet de maintenir, à la queue de la chaîne de diffusion, une diffusion de l'information pour informer les nouveaux véhicules. Ce mécanisme est réalisé jusqu'à ce que l'information ne soit plus considérée comme pertinente par le véhicule diffuseur.

Voilà donc comment, par le moyen d'une formule assez simple, nous parvenons à limiter le nombre de diffusions tout en garantissant une dissémination rapide de l'information. Le temps d'attente  $D$  peut sembler pénalisant et induire une dissémination plus lente mais puisque les véhicules éloignés n'attendent que très peu de temps, cela n'est pas significatif. Evidemment, la valeur de  $D$  a son importance. Mettre une valeur trop proche de 0 signifie que les véhicules, quelque soit leur distance, vont attendre très peu de temps. L'objectif étant qu'ils attendent le moins longtemps, cela semble une bonne solution. Pourtant, le risque ici est que les véhicules n'aient pas le temps de recevoir l'accusé réception pour annuler leur rediffusion. A l'inverse, une valeur trop importante provoque, malgré tout, un temps de dissémination plus long pouvant être gênant, en particulier pour les événements dangereux comme un freinage d'urgence par exemple. Le choix de la valeur de  $D$  dépend donc du degré de rapidité voulu pour un événement. Dans le cas d'un événement urgent, la valeur doit être très petite, de l'ordre de 0.3 secondes. Cela provoque des diffusions multiples du même saut mais permet vraiment une dissémination rapide. Concernant les événements moins urgents (comme la place de stationnement libre par exemple), une valeur de  $D$  à 2 secondes est raisonnable. Le temps de diffuser l'information n'en est pas trop ralenti et la bande passante est très fortement économisée.

Voyons maintenant comment notre protocole de dissémination est mis en place.

### 3 Mise en place du protocole de dissémination

Pour mettre en place notre solution, chaque véhicule doit posséder un certain nombre d'informations sur l'ensemble des messages qu'il a reçus. De plus, puisque nous ne rediffusons pas les informations immédiatement, cela signifie que nous avons une file d'attente d'émission. Autrement dit chaque véhicule se constitue une file contenant les messages à diffuser. Ensuite, il est nécessaire de gérer la réception de nouveaux messages et de mettre à jour la file d'attente en conséquence. Ces trois points sont présentés dans la suite de ce chapitre.

### 3.1 Données nécessaires

Les données nécessaires pour notre protocole de dissémination sont représentées dans le tableau 5.1.

Champs	Type
event	Event
cancelHopNumber	int
version	int
time	double

TABLE 5.1 – Représentation des DisseminationInformation.

Tout d'abord, nous avons l'événement considéré. Il s'agit donc de l'événement proprement dit, tel qu'il a été reçu. Il contient donc tous les attributs des événements (voir le tableau 3.1). C'est donc aussi l'événement qui doit être rediffusé.

Ensuite, nous avons le *cancelHopNumber*. Cette valeur permet de définir le numéro de saut qui peut annuler la rediffusion. Autrement dit, tant qu'aucun événement avec ce numéro de saut n'est reçu, la demande de diffusion est valide. Par contre, si l'événement est reçu avec ce numéro de saut, alors la rediffusion n'a pas besoin d'avoir lieu. Cela permet de limiter le nombre de dissémination dans le cas où le message a déjà été rediffusé par un autre véhicule.

Pour gérer les différentes versions des événements, nous avons un attribut *version* qui permet d'identifier la version de l'événement qui a été reçue. En effet, lorsqu'une nouvelle version d'un événement est générée, la chaîne de dissémination doit recommencer. Les disséminations des versions précédentes ne doivent pas interférer avec la nouvelle version. De manière plus équivoque, une nouvelle version d'un événement doit court-circuiter le mécanisme basé sur le numéro de saut. Peu importe le numéro de saut voulu par rapport à une version précédente, si une nouvelle version arrive, la version précédente doit être oubliée.

Pour finir, nous avons un temps *time*. Ce temps est le temps auquel l'événement doit être diffusé. C'est un temps propre à chaque véhicule et aucune synchronisation entre les différents véhicules n'est nécessaire. Il s'agit simplement ici de savoir si suffisamment de temps s'est écoulé pour que la dissémination doit être effectuée.

Grâce à ces données, nous pouvons maintenant gérer un ensemble d'événements à disséminer dans une file d'attente. Cette file d'attente contient donc un certain nombre de ces données. Une donnée pour chaque événement reçu. Voyons maintenant comment cette file d'attente d'émissions est utilisée.

### 3.2 File d'attente d'émission

La file d'attente d'émission est utilisée par une tâche qui s'occupe d'effectuer les disséminations aux temps voulus. La tâche effectuée est décrite dans l'algorithme 5.1. Tout d'abord, nous voyons qu'il n'y a aucune entrées/sorties dans cet algorithme. En effet, la file d'attente nécessaire n'est pas une entrée de l'algorithme mais une donnée statique du système. Il n'y en a qu'une et il n'est donc pas nécessaire de la spécifier dans l'algorithme.

---

**Algorithme 5.1 : runDisseminate()**

---

```
1:  début
2:      tant que vrai faire
3:          DisseminationInformation nextdiss;
4:          nextdiss ← me.getNextDisseminationInformation();
5:          si nextdiss ≠ null && nextdiss.time ≤ me.time alors
6:              double ep ← probabilitéRencontre(nextdiss.event);
7:              si ep > SEUILLDIFFUSION alors
8:                  Evenement tosend;
9:                  tosend ← nextdiss.event.clone();
10:                 tosend.hopNumber ← nextdiss.event.hopNumber + 1;
11:                 tosend.lastDiffuserPos ← me.currentPos;
12:                 diffuseEvent(tosend);
13:                 ediss.cancelHopNumber ← nextdiss.event.hopNumber + 2;
14:                 ediss.time ← ediss.time + DELAIMAX;
15:                 me.sortDisseminationInformation();
16:             sinon
17:                 removeDiffuse(nextdiss);
18:         sinon
19:             wait(10);
20:     fin
```

---

Ensuite, cet algorithme est constitué par une boucle infinie. La tâche doit être exécutée durant toute la durée d'utilisation du système. La boucle commence par récupérer les données de dissémination (tableau 5.1) dont le temps est le plus petit. En effet, on suppose que la file d'attente est triée par rapport aux temps de dissémination. De cette manière, on obtient rapidement l'élément qui doit être diffusé en premier. Si nous avons bien une information de dissémination dans notre file et si le temps actuel est supérieur ou égal au temps contenu dans l'information de dissémination, alors il est nécessaire d'exécuter cette dissémination.

Pour cela, nous commençons par vérifier la probabilité de rencontre avec l'événement. En effet, pour maintenir les événements dans leur zone de validité, nous avons présenté plus haut que l'utilisation de la probabilité de rencontre permettait de filtrer les événements pertinents et de les garder dans leur zone.

Si cette probabilité de rencontre est supérieure au seuil de dissémination, alors il faut effectivement disséminer l'information. Pour cela, nous faisons une copie de l'événement contenu dans l'information de dissémination. De cette manière, nous ne modifions pas l'événement reçu mais

créons bien un nouvel événement, identique au précédent mais dans lequel nous pouvons inclure les valeurs propres au véhicule.

Avant de diffuser l'information, nous modifions donc deux valeurs dans la copie de l'événement pour que les véhicules recevant l'événement de cette dissémination puisse réaliser leur traitement avec les bonnes valeurs en terme de dissémination. Les valeurs modifiées sont donc le numéro de saut de l'événement que nous incrémentons de 1 par rapport à l'événement reçu et la position du dernier diffuseur qui devient la courante position du véhicule lui-même. Cela permettra aux véhicules recevant ce message de calculer leur distance par rapport au nouveau diffuseur (le véhicule qui est en train de diffuser) avec le bon numéro de saut.

Après la dissémination de l'événement, le véhicule se met à jour en terme de dissémination. C'est-à-dire d'une part, le numéro de saut qui va annuler la rediffusion n'est plus le numéro de saut de l'événement reçu +1 mais le numéro de saut de l'événement reçu +2. En effet, puisque le véhicule a diffusé le numéro de saut de l'événement reçu +1, la réception de ce même numéro de saut n'est pas suffisante pour annuler la rediffusion, c'est le prochain saut qui doit annuler la rediffusion et donc, le numéro de saut de l'événement reçu +2. D'autre part, même si l'événement vient d'être diffusé, cela ne signifie pas que le véhicule ne devra plus le diffuser. En effet, nous avons montré précédemment que s'il ne recevait pas une nouvelle fois l'événement avec le numéro de saut suivant, le véhicule devait rediffuser l'information avec une fréquence donnée par  $D$  qui est le délai maximum entre deux disséminations. Nous mettons donc à jour l'information de dissémination pour lui ajouter au temps de dissémination ce délai. Ensuite, nous trions la file d'attente puisque l'information de dissémination actuellement traitée n'est plus à sa place. Elle doit être replacée en prenant en compte le temps de la prochaine dissémination.

Nous avons décrit le cas où la probabilité de rencontre était suffisamment importante pour que l'événement doive être diffusé. Si ce n'est pas le cas, l'information de dissémination est simplement effacée et la chaîne s'arrête alors. En effet, si la probabilité de rencontre n'est plus suffisante, cela signifie que le véhicule s'est déplacé hors de la zone de validité de l'événement ou simplement que l'événement est devenu obsolète. Dans l'un ou l'autre des cas, l'événement doit cesser d'être émis.

Pour finir, s'il n'y a aucune information de dissémination à traiter, la tâche se met en pause durant une courte période (0,01 seconde) avant de tester à nouveau la présence d'une information de dissémination. De manière infinie, les informations de dissémination sont ainsi évaluées pour permettre une diffusion au moment où elles doivent être effectuées.

Ce mécanisme de file d'attente permet donc de considérer chaque demande de diffusion avec les bons temps d'attente. La dernière question concernant notre protocole de dissémination porte donc sur la gestion des événements reçus et en particulier, sur la façon dont les numéros de saut sont traités pour prendre les bonnes décisions.

### 3.3 Réception d'événements et prise de décision

Il s'agit pour nous maintenant de montrer comment la réception d'événements est répercutée dans notre file d'attente. L'algorithme 5.2 représente le traitement effectué, au niveau de la dissémination, lorsqu'un événement est reçu. Nous avons donc en entrée un événement. C'est un événement qui vient d'être reçu. En sortie, nous n'avons aucune donnée. Seule la liste d'attente de diffusions est modifiée par l'algorithme.

---

**Algorithme 5.2 : evenementRecu(Evenement e)**

---

```
Entrées : Un événement
1:  début
2:      DisseminationInformation ediss;
3:      ediss ← me.getDisseminationInformation(e);
4:      si ediss ≠ null && ediss.version < e.version alors
5:          removeDiffuse(ediss);
6:          ediss ← null;
7:      si ediss = null alors
8:          double ep ← probabiliteRencontre(e);
9:          si ep > SEUILDIFFUSION alors
10:             double d ← distance(me.currentPos, e.lastDiffuserPos);
11:             double t ← DELAIMAX · (1 - d / PORTEEMAX);
12:             ediss ← new DisseminationInformation();
13:             ediss.event ← e;
14:             ediss.cancelHopNumber ← e.hopNumber + 1;
15:             ediss.version ← e.version;
16:             ediss.time ← t + me.currentTime();
17:             addNextDiffuse(ediss);
18:         sinon si ediss.cancelHopNumber ≤ e.hopNumber alors
19:             removeDiffuse(ediss);
20:     fin
```

---

L'algorithme commence par vérifier la présence d'une information de dissémination portant sur l'événement reçu. Ce n'est pas spécifié dans l'algorithme mais cela est réalisé grâce aux identifiants uniques des événements. Si une telle information existe mais qu'elle porte sur une version plus ancienne de l'événement, alors elle est supprimée et l'on considère qu'il n'y a pas d'information de dissémination sur l'événement. Cela permet de considérer l'événement comme étant un nouvel événement et de ne pas mélanger les différents numéros de sauts avec les différentes versions.

Ensuite donc, si aucune information de dissémination portant sur l'événement n'était présente (ou si elle a été effacée à cause de versions différentes), nous arrivons dans la création d'une information de dissémination. Cette création est tout d'abord conditionnée par la valeur de la probabilité de rencontre. En effet, dans le cas où la probabilité de rencontre n'est pas suffisante, aucune information de dissémination n'est créée puisque l'événement ne doit pas être rediffusé.

En supposant que la probabilité de rencontre est suffisante, nous calculons d'abord le temps d'attente. Pour cela, nous utilisons la formule présentée précédemment avec un délai maximum, une

portée maximale et la distance entre le véhicule et la position du dernier diffuseur. Cette position est insérée juste avant l'émission de l'événement (voir la sous-section 3.2). Elle est donc incluse dans l'événement reçu et représente la position à laquelle l'événement reçu a été émis. Une fois le temps d'attente calculé, une nouvelle information peut être construite. Dans cette information de dissémination, l'événement est l'événement reçu, le numéro de saut permettant l'annulation de la rediffusion est le numéro du saut de l'événement reçu +1, la version est la version de l'événement reçu et le temps est le temps actuel du véhicule ajouté au temps d'attente calculé. Il est important de noter que tant que l'événement n'a pas été rediffusé, le numéro de saut permettant l'annulation est bien le même que celui de l'événement reçu +1. En effet, le véhicule doit, pour l'instant, attendre et voir si le saut +1 va être diffusé par quelqu'un d'autre. Ce n'est que si ce n'est pas le cas qu'il va diffuser le saut numéro +1 et attendre le saut numéro +2. Quoiqu'il en soit, une fois l'information de dissémination créée, elle est ajoutée à la file d'attente. La fonction d'ajout s'occupe d'insérer l'information au bon endroit, en respectant le tri des temps. Grâce à l'utilisation de la tâche de dissémination, il n'est pas nécessaire ici de s'occuper de la dissémination effective.

Par contre, un dernier cas se pose concernant la réception d'un événement. C'est le cas où l'information de dissémination existe déjà. Dans ce cas, il n'est pas nécessaire d'en créer une nouvelle mais il faut vérifier que l'événement reçu n'annule pas une information de dissémination. Pour cela, nous comparons le numéro du saut de l'événement reçu avec le numéro du saut permettant l'annulation de l'information déjà existante. En fonction du résultat de cette comparaison, l'information de dissémination est enlevée ou non.

Ainsi, chaque réception d'événement met à jour la file d'attente d'émission. Grâce à notre mécanisme de numéro de saut, certaines diffusions sont annulées, nous permettant d'économiser la bande passante au mieux. Désynchroniser les diffusions permet d'augmenter le nombre d'annulations. En effet, les véhicules n'attendent pas pendant une durée fixe et identique et par conséquent, les diffusions et les réceptions des uns s'opèrent pendant les temps d'attentes des autres, ce qui rend le mécanisme particulièrement adapté à nos contraintes.

## 4 Le cas des ressources

La dissémination d'une information sur une ressource peut être problématique. En effet, dans le cas d'une place de stationnement par exemple, si l'on diffuse l'information à un ensemble de conducteurs, tous risquent de vouloir prendre la place mais un seul pourra l'obtenir. Nous risquons donc de créer une compétition entre des groupes de véhicules. Il serait plus judicieux de fournir l'information à un seul automobiliste. Pour cela, nous avons proposé une alternative au mécanisme de dissémination qui consiste en un protocole de réservation [DCIL09]. Il s'applique à toutes les ressources et en particulier aux places de stationnement dont la gestion représente un réel défi.

### 4.1 Protocole de réservation

Notre protocole de réservation a pour objectif d'associer chaque ressource à un unique véhicule. Pour éviter la compétition entre les véhicules, un tel protocole doit satisfaire les propriétés suivantes :

- ◊ Indiquer à l'automobiliste la disponibilité d'une ressource qui lui est réservée (et non pas une liste de toutes les ressources qu'il a reçues);
- ◊ Maximiser la probabilité que cette ressource soit toujours disponible quand le conducteur arrivera;
- ◊ Eviter qu'une ressource reste disponible alors que des véhicules proches la recherchent pour maximiser l'utilisation des ressources;
- ◊ Etre équitable en s'assurant qu'un véhicule n'a pas une priorité plus importante que les autres;
- ◊ Minimiser les actions que le conducteur doit faire, avec deux objectifs :
  1. Ne pas déranger le conducteur,
  2. Préserver l'équité du protocole en ne permettant pas aux conducteurs de diffuser des fausses informations qui tromperaient les autres automobilistes (afin d'obtenir la ressource plus facilement);
- ◊ Eviter une congestion du réseau.

Notre solution s'appuie sur *l'élection d'un coordinateur par ressource*. Il s'agit d'un véhicule particulier dont la mission est d'affecter la ressource au bon véhicule, en respectant une certaine politique. Ce coordinateur ne doit pas être intéressé par la ressource. Dans le cas d'une place de stationnement, le coordinateur est le véhicule qui la libère<sup>2</sup>. Dans la suite, nous détaillons le fonctionnement de notre protocole. Nous considérons dans un premier temps le cas où des véhicules proches du coordinateur recherchent la ressource. Dans ce cas, le coordinateur diffuse dans un premier temps l'information sur la disponibilité de la ressource. Ensuite, il attend des réponses pendant un certain temps. Chaque véhicule ayant reçu l'information et recherchant la ressource répond au coordinateur (voir l'algorithme 5.3 (page 79)) en lui donnant :

- ◊ Son identifiant.
- ◊ Les informations nécessaires à la politique d'attribution des ressources. Pour nous, il s'agit du temps auquel la recherche de la ressource a commencé et de la probabilité de rencontrer la ressource diffusée.

---

2. Si l'information est générée par un capteur sur la place de stationnement, un coordinateur est choisi parmi les véhicules alentours. Il est aussi possible d'utiliser les parcmètres comme coordinateur s'ils sont équipés du matériel nécessaire [BL04]

**Algorithme 5.3 : void declareInterest(Event re, Coordinator c)**


---

```

/* Exécutée lorsqu'un véhicule recherchant une ressource reçoit un événement
correspondant à une ressource disponible re d'un coordinateur c. */
1: début
2:   float ep = computeEp(re);
3:   si ep ≥ RelevanceThreshold alors
4:     send(c.identifieur, getSearchDuration(), ep, myID);
5: fin

```

---

Finalement, une fois que le temps d'attente des réponses est écoulé, le coordinateur choisi, parmi les véhicules ayant répondu, celui pour qui la ressource est réservée. Pour faire ce choix, différentes politiques peuvent être appliquées. Pour être équitable, la durée de recherche doit être prise en compte. Cette politique est très intéressante mais elle ne fonctionne très bien que pour les ressources pouvant être physiquement réservées. Cependant, pour les places de stationnement, il est impossible de réserver physiquement la ressource. De ce fait, il y a un risque que la ressource soit prise par un autre véhicule malgré tout, alors qu'elle était réservée pour un véhicule qui recherchait depuis longtemps. Pour cette raison, il est nécessaire de prendre en compte la probabilité de rencontre dans la politique d'attribution des ressources. En prenant en compte la probabilité de rencontre, nous minimisons le risque que la place soit prise par un autre usager. La politique choisie dépend donc de la ressource considérée.

**Algorithme 5.4 : void confirmInterest(Event re, Coordinator c)**


---

```

/* Exécutée lorsqu'un véhicule est informé par un coordinateur c que la
ressource re lui a été allouée. */
1: début
2:   si searching-for-a-parking-space alors
3:     searching-for-a-parking-space = false;
4:     send(c.identifieur, commitMessage, myID);
5:   sinon
6:     /* Une ressource a déjà été trouvée ou le conducteur a changé d'avis.
   */
7:     send(c.identifieur, abortMessage, myID);
7: fin

```

---

Pour éviter de perdre une ressource dans le cas où un véhicule élu renonce à sa recherche, le véhicule élu doit confirmer sa demande (voir l'algorithme 5.4 (page 79)) auprès du coordinateur après avoir été informé de son attribution. Dans le cas où le coordinateur ne reçoit pas de confirmation, il alloue de nouveau la ressource à un autre véhicule et le processus continue. Deux raisons peuvent faire que le coordinateur ne reçoit pas une confirmation :

1. Le véhicule n'est plus intéressé par la ressource. Dans ce cas, la ré-attribution de la ressource à un autre véhicule ne pose aucun problème.
2. La confirmation n'a pas été reçue parce que les véhicules sont devenus trop éloignés. Dans ce cas, la ré-attribution de la ressource va provoquer une compétition entre deux véhicules.

Cela est gênant puisque le protocole de réservation doit permettre d'éviter cette situation. Cependant, la compétition est tout de même très limitée et le plus important reste que la ressource soit utilisée autant que possible. C'est pourquoi cela nous semble malgré tout la meilleure solution.

---

### Algorithme 5.5 : void advertiseParking(Event re)

---

```
/* Exécutée par le coordinateur à la libération de la ressource. */
1:  début
2:      broadcast(re, myID);
3:      listen(T, answers);
4:      si answers ≠ null alors
5:          sort(answers);          /* En respectant la politique d'allocation. */
6:          i=0;
7:          notification=false;
8:          tant que i < answers.size && notification==false faire
9:              notification = notify(answers[i++].identifier); /* La notification devient
                                                                effective lorsque le véhicule élu confirme l'allocation. */
10:         sinon
11:             electNewCoordinator(); /* Election d'un nouveau coordinateur si aucun
                                     véhicule n'était intéressé. */
12:     fin
```

---

L'algorithme 5.5 (page 80)) représente ce qu'il se passe du côté du coordinateur. Il peut arriver qu'aucun véhicule voisin ne soit intéressé par la ressource diffusée. Pour atteindre des véhicules plus éloignés, il est intéressant de désigner un nouveau coordinateur. En effet, un nouveau coordinateur possède d'autres voisins qui peuvent être intéressés. L'élection d'un nouveau coordinateur est expliquée dans la suite.

#### 4.2 Election d'un nouveau coordinateur.

Pour permettre à l'information d'atteindre de nouveaux véhicules jusqu'à ce que la ressource soit allouée, il est nécessaire d'étendre la portée de dissémination de l'information. Pour cela, les techniques de multi-hop ne sont pas utilisables car le processus d'allocation ne peut pas être garanti. Les risques de perdre les confirmations deviennent trop importants. Pour cette raison, nous proposons de déplacer le processus de décision sur un autre véhicule. Ce nouveau véhicule devient alors coordinateur et le mécanisme d'élection peut se répéter. L'algorithme 5.6 (page 81) montre le processus d'élection d'un nouveau coordinateur.

**Algorithme 5.6 : void electNewCoordinator()**


---

```

/* Exécutée lorsqu'un coordinateur veut être remplacé. */
1:  début
2:      notification = false;
3:      broadcast(coord, myID);
4:      listen(T, answers);
5:      si answers ≠ null alors
6:          sort(answers);
7:          i=0;
8:          tant que i < answers.size && notification==false faire
9:              notification = notify(answers[i++].identifier); /* La notification devient
               effective lorsqu'un véhicule accepte de devenir le nouveau
               coordinateur. */
10:     si notification == false alors
11:         wait(T);
12:         advertiseParking(re);
13:  fin

```

---

Tout d'abord, le coordinateur actuel diffuse une *demande de remplacement*. Les véhicules alentours répondent alors en donnant certaines informations. Le coordinateur actuel choisit alors le nouveau coordinateur et attend la confirmation.

La politique d'élection d'un nouveau coordinateur peut donc dépendre de différents critères. Dans le cas où l'objectif est de maintenir un coordinateur proche de la ressource, la probabilité de rencontre est utilisée. En élisant comme nouveau coordinateur le véhicule ayant la plus grande probabilité de rencontrer la ressource, nous limitons le risque que l'information sur la ressource soit disséminée très loin de celle-ci. Une autre politique possible est de prendre en compte les voisins des coordinateurs potentiels. En supposant que les véhicules recherchant une ressource diffusent leur intérêt périodiquement, chaque véhicule peut savoir à chaque instant combien de voisins sont en recherche d'une ressource. En se basant sur cette information, l'élection d'un nouveau coordinateur permet d'allouer plus rapidement la ressource puisque le nouveau coordinateur est choisi en fonction de ses voisins. Par contre, cette politique nécessite la diffusion périodique des ressources recherchées par chaque véhicule.

## 5 Conclusion

Nous avons présenté un nouveau mécanisme de diffusion permettant aux véhicules de recevoir les événements susceptibles d'intéresser les conducteurs [CDI08]. Notre mécanisme de dissémination repose sur un échange limité de données. Dans un premier temps, nous utilisons la probabilité de rencontre pour déterminer la zone de validité des événements. Autrement dit, l'événement seul permet à chaque véhicule d'évaluer sa validité et l'évaluation sur chaque véhicule permet de définir de manière tout à fait implicite et transparente la zone de validité des événements. Ensuite, notre mécanisme repose sur le mode broadcast. Le nombre d'envois est donc aussi petit que possible tout

en permettant d'atteindre un grand nombre de véhicules. Enfin, parmi tous les véhicules, seuls certains rediffusent l'information. Alors que l'utilisation de monitoring ou d'un mécanisme d'élection aurait pu convenir, cela aurait provoqué une consommation importante de la bande passante, sans parler du temps nécessaire à la mise en place d'un tel mécanisme. Notre solution repose simplement sur un attribut des événements permettant d'évaluer facilement et rapidement quels véhicules vont diffuser les premiers. Sans avoir besoin d'établir d'accord entre les véhicules, simplement en respectant la règle du temps d'attente, nous déterminons les meilleurs candidats à la rediffusion. De cette manière, la rediffusion est non seulement rapide et efficace mais aussi économe. En allant encore plus loin, nous avons fourni un mécanisme permettant d'éviter certaines rediffusions. Notre utilisation d'un numéro de saut assure fiabilité et là encore, économie.

Nous avons aussi montré comment ce protocole était mis en place dans notre système. La structure de données et les algorithmes qu'il nécessite sont à la fois simples et efficaces. Ils permettent une gestion des diffusions adaptée à nos contraintes qui sont la limite de la bande passante et les temps de communication très courts. Pour la dissémination des événements, ce protocole est donc très efficace.

Par contre, dans le cas des ressources, le problème de compétition est très gênant puisqu'informer tous les véhicules n'est plus une bonne solution. Nous avons donc présenté un protocole de réservation permettant de sélectionner les véhicules qui tiendront compte d'une information. Ce protocole a été validé dans [DCIL09]. Bien que nos politiques d'allocation de ressources restent très simples, elles permettent déjà d'améliorer l'utilisation des ressources. De même, nous avons montré comment atteindre de nouveaux véhicules dans le cas des ressources. Pour cela, nous nous appuyons sur l'élection d'un coordinateur. Là encore, les politiques permettant de choisir les nouveaux coordinateurs sont simples mais elles montrent tout de même tout l'intérêt de notre proposition.

# 6

---

## Prototype, simulation et expérimentations

---

Ce chapitre présente les implémentations réalisées pour valider les solutions présentées dans ce manuscrit. Les deux objectifs principaux ici sont :

1. Présenter nos implémentations pour montrer leur faisabilité dans le contexte technologique actuel.
2. Présenter les résultats obtenus grâce à nos implémentations afin de valider nos solutions.

Ce chapitre est décomposé en trois sections. La première concerne le prototype de notre système. C'est-à-dire la mise en place à proprement parler de nos solutions dans un système embarqué dans les véhicules. La seconde section s'attarde sur notre simulateur. Pour pouvoir obtenir des résultats sans avoir à recourir au déploiement du système à grande échelle, l'utilisation de simulateur est inévitable. Enfin, la troisième section montre les différents résultats obtenus.

### 1 Un prototype...

L'objectif dans le développement d'un prototype était de vérifier la faisabilité d'un tel système. Le système embarqué dans les véhicules repose sur l'utilisation de technologies existantes. Dans un premier temps, nous présentons les outils matériels que nous avons utilisés pour notre système. Ensuite, nous présentons notre architecture logicielle déployée sur ce matériel.

#### 1.1 Architecture matérielle

L'architecture matérielle nécessaire à notre système peut sembler assez lourde. En effet, nous avons besoin d'un module de positionnement, d'un module de communication sans fil, d'une certaine puissance processeur, d'une certaine quantité de mémoire, etc... Pourtant, aujourd'hui, l'accès à ces technologies n'est pas problématique. En effet, les coûts sont de plus en plus réduits et les appareils offrant ce genre de modules sont en pleine expansion. Voyons en détail nos besoins et les possibilités qui nous étaient offertes.

##### Module de positionnement

Un module de positionnement est un module permettant à un appareil d'être localisé. Un tel module est une nécessité pour notre système. Que ce soit pour l'évaluation de la pertinence, pour l'évaluation continue ou pour la dissémination de l'information, il est absolument nécessaire, pour notre système, d'être localisé. Aujourd'hui, c'est le système américain GPS – Global Positioning System – qui est le plus utilisé de part le monde. La précision du GPS dépend du nombre de satellites captés. Dans les meilleures conditions, c'est-à-dire en extérieur dans un environnement dégagé, l'erreur est inférieure à 5 mètres. Par contre, dans un environnement urbain avec des hauts bâtiments tout autour, l'erreur peut atteindre 20 mètres. Malgré les problèmes de précision qui peuvent être très handicapant pour les services liés à la sécurité, dans le cadre d'un prototype, cette précision est suffisante pour valider notre approche. Enfin, notons qu'à l'heure actuelle, il n'existe pas d'autre solution utilisable dans notre environnement. Heureusement, dans un avenir plus ou moins proche, le nouveau système Européen GALILEO va faire son apparition, nous offrant ainsi une alternative au système Américain. Pour nous, le choix d'un système de positionnement n'est pas définitif. Il ne s'agit que d'un support pour notre système.

##### Module de communication sans fil

Le module de communication sans fil est un module crucial pour notre système. En effet, l'échange d'informations entre les véhicules se fait par le biais de ce module. Le choix de la technologie utilisée repose essentiellement sur son utilisation. Pour nous, la norme 802.11g est

celle qui s'impose d'elle-même à l'heure actuelle puisque c'est celle qui offre la plus grande portée. La portée annoncée en environnement dégagé est de plusieurs centaines de mètres. Notons que là encore, le choix de cette technologie n'est pas définitif. Les spécifications de la technologie utilisée doivent simplement être connues pour adapter notre protocole de dissémination à cette technologie. Nous attendons la norme 802.11p qui est en train d'être mise au point et devrait être particulièrement adaptée à l'environnement inter-véhiculaire.

### La machine choisie

La machine que nous devons embarquer dans les véhicules devait donc supporter les différents modules nécessaires et offrir des capacités de calcul et de mémoire suffisantes. Nous nous sommes donc tournés vers les appareils légers offrant des capacités de communication sans fil et intégrant une puce GPS directement. En effet, certains téléphones ont aujourd'hui de telles capacités. De plus, ces appareils sont particulièrement adaptés à une utilisation dans un véhicule puisqu'ils proposent généralement des supports pour voiture. Leur écran tactile permet une utilisation simple et intuitive qui correspond exactement à ce que nous voulons offrir. Enfin, de plus en plus de personnes possèdent ce genre de téléphone. La perspective d'un déploiement à grande échelle est donc beaucoup plus envisageable. Par contre, l'inconvénient majeur de ces terminaux est leur faible puissance autant au niveau processeur qu'au niveau mémoire. Le développement d'un prototype sur ce type d'appareil représente donc un défi important puisqu'il s'agit bien ici de confronter nos propositions face à une réalité technologique accessible à tous.



FIGURE 6.1 – Terminaux choisis pour le développement de VESPA.

## 1.2 Architecture de VESPA

Les terminaux que nous avons choisis nous offrent tout ce dont nous avons besoin pour mettre en place notre système. Nous avons choisi de baptiser notre système VESPA : Vehicular Event Sharing



une interaction avec le système et que la fiabilité des événements n'est pas à remettre en cause.

Voyons maintenant brièvement comment les différents modules interagissent les uns avec les autres :

1. Lorsqu'un événement est reçu, il est directement envoyé à l'*Event Manager*. L'*Encounter Probability Evaluator* calcule la probabilité de rencontrer l'événement en utilisant les informations contenues dans l'événement et les informations fournies par le *Position Manager*. Si la valeur de la probabilité de rencontre est suffisante, l'événement est envoyé au *Continuous Query Processor*. Dans le cas d'un événement ayant une haute *Importance* (par exemple, un freinage d'urgence), il est directement transmis au conducteur via la *Driver Interface*. Si l'événement n'a pas une haute importance, le *Continuous Query Processor* vérifie si l'événement répond à une requête continue active. Comme la pertinence de l'événement peut varier constamment (puisque la position du véhicule change constamment), le *Continuous Query Processor* évalue en utilisant l'*Encounter Probability Evaluator*, en continu les requêtes continues actives pour vérifier quelles informations doivent être présentées au conducteur.
2. Le *Storage Manager* est informé par le *Encounter Probability Evaluator* des probabilités de rencontre qu'il calcule. Si la probabilité de rencontre d'un événement précédemment stocké passe sous le seuil de stockage, alors le *Storage Manager* efface l'événement de la zone de stockage. A l'inverse, si la probabilité de rencontre d'un nouvel événement est plus grande que le seuil de stockage, l'événement est stocké.
3. Pour un événement nouvellement reçu, si la valeur de la probabilité de rencontre est plus grande que le seuil de diffusion, l'événement est envoyé au *Dissemination Manager* pour être ajouté à la pile de disséminations. L'ajout dans la pile de disséminations se fait en fonction de la distance par rapport au dernier diffuseur (voir la section 5).

Notons que si un véhicule reçoit plusieurs fois un événement, l'événement n'est pas présenté plusieurs fois au conducteur, à moins qu'une information relative à l'événement soit modifiée, rendant l'événement pertinent (par exemple, dans le cas d'un événement mobile). Par contre, le *Wireless Communication Manager* met à jour sa pile de disséminations en conséquence.

### 1.3 VESPA dans les véhicules

Le prototype développé a été déployé dans des véhicules pour pouvoir être confronté à la réalité. Cela nous a permis de voir le comportement de notre système face à l'imprécision du module de positionnement et de vérifier la capacité des modules de communication sans fil à échanger des informations lors de forte mobilité. Les figures 6.3 et 6.4 montrent le prototype utilisé dans les véhicules.

Le prototype nous a permis de vérifier la faisabilité de notre système en confirmant que les modules de positionnement sont suffisamment précis pour fournir des informations exploitables sur



FIGURE 6.3 – VESPA signalant une place de stationnement



FIGURE 6.4 – VESPA signalant un danger sur la route

les événements. De plus, l'échange d'informations entre les véhicules est bien possible en utilisant les puces 802.11g. Pourtant, les portées de communication relevées ne sont pas aussi importantes que celles annoncées. Dans le cas de véhicules circulant dans le même sens de circulation, nous avons pu établir une communication à une distance d'environ 100 mètres. Par contre, dans le cas de véhicules circulant en sens contraire, le temps pour initialiser la communication est tel que les premiers messages échangés le sont lorsque les véhicules sont très proches (de l'ordre de 30 mètres) en circulant à une vitesse de 50 km/h.

Cela vient d'une part des puces 802.11g qui sont intégrées dans nos terminaux et d'autre part du temps nécessaire à l'attribution des adresses réseaux. En effet, même si nous ne faisons que diffuser les événements, le protocole utilisé repose sur le protocole IP. C'est-à-dire qu'avant

qu'un échange d'informations soit possible, les terminaux ont besoin d'une adresse IP. Le temps nécessaire à l'obtention de ces adresses est très pénalisant puisqu'il nécessite une communication préalable pouvant durer jusqu'à 2 secondes. Ces 2 secondes peuvent être cruciales dès lors que nous considérons des événements comme un freinage d'urgence par exemple. Pourtant, notre système n'a pas besoin du protocole IP. En effet, puisque nous ne faisons que de la diffusion d'informations, il n'est pas nécessaire pour nous que chaque véhicule ait une adresse IP propre. Une puce permettant la diffusion et la réception de messages serait suffisante. De cette manière, nous éviterions le temps nécessaire au démarrage de la communication. C'est pourquoi, bien que notre prototype souffre de l'utilisation du protocole IP, nous cherchons un moyen d'utiliser les puces 802.11g sans ce protocole pour améliorer les communications. Quoiqu'il en soit, le prototype nous a permis de recueillir des informations très intéressantes que nous avons exploitées dans nos simulations.

## 2 Un simulateur...

Dans cette section, nous présentons notre simulateur. Le prototype développé nous a permis d'obtenir des informations intéressantes concernant le comportement de notre système en environnement réel. Pourtant, tester notre prototype à grande échelle n'a pas été possible. Un trop grand nombre de véhicules équipés est nécessaire pour pouvoir avoir une vision globale de l'impact de l'utilisation de notre système. C'est pourquoi, l'utilisation d'un simulateur a été nécessaire.

### 2.1 Le choix du simulateur

Les performances des applications et des protocoles destinés aux réseaux inter-véhiculaires sont souvent évaluées grâce à des simulateurs. Deux composants importants sont à prendre en compte pour cela. D'un côté, nous devons considérer des modèles de mobilité réalistes. Dans la réalité, les véhicules ne se déplacent pas aléatoirement. Chaque automobiliste suit une route bien précise en fonction de l'endroit où il veut se rendre. De plus, le réseau routier est un environnement fortement contraint où les chemins, les vitesses, les ordres de passages sont réglementés. Pour cela, il existe un certain nombre de simulateurs de trafic. Certains sont commerciaux – comme *CORSIM*<sup>2</sup> ou *TRANSIMS*<sup>3</sup> – et d'autres sont libres – comme *VanetMobiSim* [FHFB07] ou *GrooveNet* [MWR<sup>+</sup>06]. D'un autre côté, nous devons considérer les simulateurs de réseaux. Les plus connus sont *NS2*<sup>4</sup>, *GloMoSim*<sup>5</sup>, ou *JiST-SWANS*<sup>6</sup>. Quelques travaux proposent d'intégrer les deux, comme *TraNS*<sup>7</sup> [PRL<sup>+</sup>08].

---

2. <http://mctrans.ce.ufl.edu/featured/tsis/version5/corsim.htm>

3. <http://transims.tsasa.lanl.gov/>

4. <http://www.isi.edu/nsnam/ns/>

5. <http://pcl.cs.ucla.edu/projects/glomosis/>

6. <http://jist.ece.cornell.edu/>

7. <http://trans.epfl.ch/>

Le simulateur choisi devait répondre à certains critères propres à nos besoins. Nous nous sommes rendus compte qu'adapter un simulateur à nos besoins était une tâche très compliquée. En effet, s'il est possible d'adapter les simulateurs réseaux pour tester nos propres protocoles, cela est beaucoup plus difficile lorsque l'on veut évaluer le partage d'informations. Le simulateur doit nous permettre de représenter l'environnement, de définir différentes routes pour les véhicules, de simuler des véhicules sans destination précise (pour simuler la recherche de place de stationnement), d'intégrer différentes vitesses et différentes conditions de circulation. Lorsqu'un événement survient sur la route, il est nécessaire de pouvoir identifier les véhicules qui ont rencontré l'événement et à quel moment. De plus, le comportement des conducteurs doit pouvoir être influencé par les événements reçus. Le partage d'informations implique une exploitation de ces informations dans le simulateur. Le comportement des automobilistes doit être fortement lié aux informations transmises par le système simulé. Par exemple, si un conducteur reçoit un événement sur un embouteillage, il doit essayer de modifier sa route. De même, lorsqu'un conducteur cherche une place de stationnement, il doit circuler aléatoirement dans la zone de recherche et se diriger en fonction des événements concernant des libérations de places de stationnement lorsqu'il en reçoit. La simulation du réseau et la simulation du trafic ne peuvent donc pas être indépendantes. Elles s'influencent mutuellement puisque d'une part, les informations reçues doivent modifier les itinéraires des véhicules et d'autre part, la modification des itinéraires provoque des variations dans la topologie du réseau. Une difficulté supplémentaire liée à l'utilisation d'un simulateur existant est la récupération de résultats spécifiques. En effet, nos critères d'évaluation sont spécifiques puisqu'ils concernent la pertinence des informations fournies aux conducteurs et nos algorithmes de dissémination. Bien que les simulateurs de réseau fournissent des mesures intéressantes, elles ne correspondent pas nécessairement à nos besoins. Nous avons donc étudié les différentes possibilités qui s'offraient à nous. Face à la difficulté d'évaluer nos propositions avec les outils existants, nous avons décidé de développer un nouveau simulateur, adapté aux besoins de VESPA.

### 2.2 Notre simulateur

Le simulateur que nous avons développé nous a permis de simuler des contextes réalistes. Nous avons choisis de représenter les routes avec des courbes de Bezier. Cela nous a permis d'obtenir un modèle de mobilité proche de la réalité et d'éviter un déplacement rectiligne des véhicules. Un ensemble de routes avec différents points d'intersection peuvent ainsi être créés. Chacune des routes possède sa propre limitation de vitesse. Pour construire nos routes, nous avons deux possibilités :

1. Extraction de données cartographiques d'une base de données Télé Atlas.
2. Création manuelle de route.

De cette manière, nous pouvons utiliser des cartes réelles pour nos simulations sur une autoroute ou dans un centre ville (voir la figure 6.5 (page 91)) ou bien nous pouvons créer nos propres configurations pour des cas particuliers. Dans les parkings de supermarché par exemple, la base de données cartographiques ne contient pas les informations sur les routes mais nous avons un

générateur de configurations de route dans un parking qui nous permet d'avoir un environnement proche de la réalité. Chaque véhicule suit donc ces routes en partant d'un point de départ pour rejoindre un point d'arrivée. Pour déterminer le chemin du véhicule, le plus court chemin est utilisé. Pour éviter d'avoir des trains de véhicules trop réguliers (ce qui ne serait pas réaliste), nous avons associé à chaque véhicule une nervosité. Cette nervosité représente la manière de conduire du conducteur. Ainsi, pendant que certains véhicules roulent moins vite que la limite autorisée, d'autres peuvent faire des excès de vitesse. Cela permet d'avoir un comportement plus proche de la réalité. De plus, pour les variations de vitesse, les véhicules accélèrent et décélèrent de manière souple. En utilisant une fonction d'accélération et de freinage, nous parvenons à imiter le comportement des véhicules sur les routes.

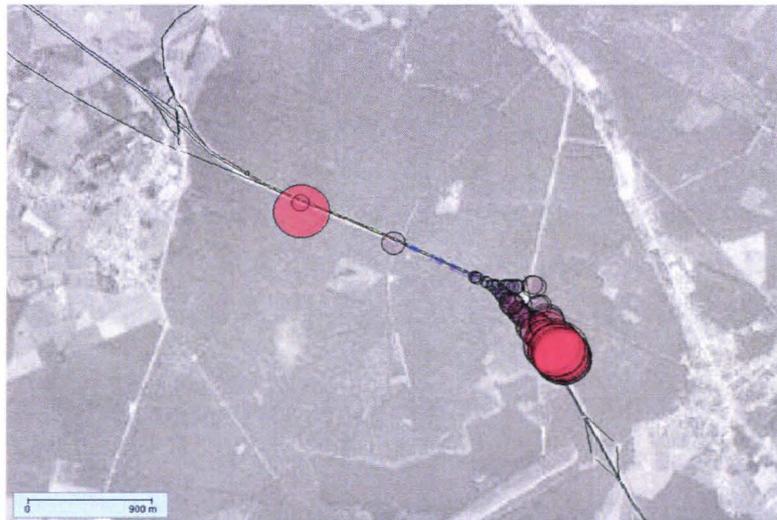


FIGURE 6.5 – Premier environnement de simulation.

Pour l'émission et la réception des informations, nous considérons que lorsqu'un message est envoyé, tous les véhicules suffisamment proches reçoivent l'information (cela dépend de la portée de communication choisie). Nous avons finalement intégré notre prototype dans notre simulateur pour que chaque information reçue soit traitée par notre système. Les événements sont ainsi rediffusés, transmis au conducteur, stockés ou ignorés en fonction du choix de notre système. Lorsqu'un événement est transmis au conducteur, un nouveau chemin est alors déterminé de sorte que les événements reçus aient bien une influence sur le comportement des véhicules. Enfin, nous avons utilisé les données observées avec le prototype en environnement réel pour calibrer notre simulateur.

### 3 Expérimentations

Les résultats présentés dans la suite correspondent à des simulations réalisées sur deux routes différentes représentant des segments d'autoroutes (voir les figures 6.6 et 6.7). La première route

correspond à un petit tronçon d'autoroute contenant deux éléments particuliers. Tout d'abord, il y a une sortie d'autoroute qui reste presque parallèle à l'autoroute. Ensuite, il y a une courbe importante à la fin. La seconde correspond à une portion de la route reliant les villes de Valenciennes et Lille (en France). Il s'agit d'un tronçon plus grand offrant une vue plus globale. Avec cette configuration, nous pouvons réellement voir à quel moment les véhicules prennent en compte un événement. Six différents événements, positionnés à des endroits stratégiques, ont été considérés pour évaluer notre proposition. Tous les événements sont immobiles<sup>8</sup> et dépendants de la direction sauf l'événement 5 qui est immobile indépendant de la direction. Entre 500 et 600 véhicules sont lancés avec des trajectoires aléatoires. Les événements restent valides pendant toute la durée des simulations. La vitesse des véhicules varie entre 90 km/h et 110 km/h, en fonction de la route (les véhicules ralentissent dans les courbes).



FIGURE 6.6 – Premier environnement de simulation.

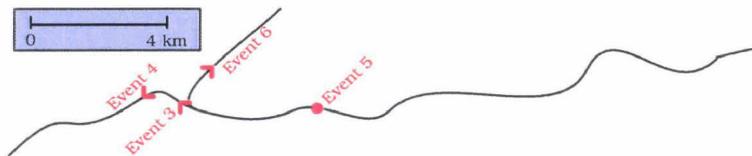


FIGURE 6.7 – Second environnement de simulation.

Pour nos expérimentations, nous avons testé différentes valeurs pour les coefficients de pénalité. Le meilleur compromis pour nos différentes routes s'est révélé être :  $\alpha = \frac{1}{3 \times 100}$ ,  $\beta = \frac{1}{3 \times 300}$ ,  $\gamma = 0$  et  $\zeta = \frac{1}{3 \times 50}$ . Ces valeurs nous donnent de bons résultats quelque soit la configuration de la route.

Les résultats de nos expérimentations nous ont permis de constater que les meilleurs vecteurs de mobilité sont ceux utilisant une position de référence distante de 500 mètres. Pour les vecteurs directions, nous utilisons une position de référence distante de 30 mètres. En théorie, nous aurions dû utiliser une position de référence plus proche pour le vecteur direction. Dans l'idéal, utiliser deux relevés successifs (c'est-à-dire avec environ une seconde d'intervalle) aurait dû être la meilleure solution. Pourtant, l'imprécision du module de positionnement rend inexploitable un vecteur basé sur deux positions successives. C'est pourquoi, même pour le vecteur direction, nous utilisons une position de référence un peu plus éloignée.

8. Nous ne présentons pas de résultats concernant des événements mobiles car ils sont beaucoup plus difficiles à interpréter. De plus, étant donné que notre mécanisme d'évaluation de la pertinence s'appuie sur un changement de référentiel pour les événements mobiles, cela signifie que pour le véhicule, un événement mobile est traité comme un événement immobile mais pour lequel la mobilité du véhicule est différente.

Ensuite, nous avons défini nos seuils de pertinence et de diffusion à 75%. La valeur de  $D$  pour le protocole de dissémination a été fixée à 1 seconde et la portée de communication à 200 mètres. Pour les communications entre les véhicules, nous avons décidé de fixer des valeurs très pessimistes, nous permettant d'évaluer VESPA dans un environnement où les délais de communication sont très importants. En effet, nous avons défini des délais de transmission des informations. Ainsi, lorsqu'une information est émise par un véhicule, elle n'est reçue qu'après un certain temps par les proches véhicules. Ce temps est compris entre 0.2 et 1.2 secondes et dépend de la distance entre les véhicules.

#### 3.1 Evaluation du mécanisme d'évaluation de la pertinence

Nous présentons ici les résultats concernant la pertinence des informations. L'objectif est d'évaluer l'utilité du concept de probabilité de rencontre. Pour cela, nous considérons que tous les événements sont diffusés en utilisant un mécanisme d'inondation du réseau. Cela permet d'assurer que notre mécanisme de dissémination n'a pas d'impact sur l'évaluation de la pertinence. L'utilisation d'un mécanisme d'inondation n'est pas réaliste en environnement réel mais cela nous permet d'assurer la bonne transmission des messages dans notre simulateur. De cette manière, nous avons la garantie que les véhicules possèdent l'information. Le fait qu'un événement soit pris en compte ne dépend donc que du mécanisme d'évaluation de la pertinence et n'est pas influencé par le mécanisme de dissémination.

Lors de nos simulations, nous mesurons le pourcentage de véhicules ayant estimé comme étant pertinent un événement avant de rencontrer cet événement. En comparant le temps au moment où l'événement a été considéré comme pertinent avec le temps où l'événement a été rencontré, nous déduisons une durée. Nous savons donc combien de temps avant de rencontrer l'événement le conducteur en a été informé. Par contre, nous ne considérons pas ici les véhicules n'ayant pas rencontré l'événement. Nos expérimentations montrent que tous les véhicules sont informés de la présence de l'événement avant de l'atteindre. La figure 6.8 présente combien de temps avant de rencontrer un événement celui-ci a été affiché. Il s'agit du temps dont dispose les véhicules pour réagir avant de rencontrer l'événement. Notons qu'aucun véhicule ne circule sur la route de l'événement 6. C'est pourquoi il n'est pas présenté dans nos résultats. Cet événement nous a servi à vérifier qu'aucun véhicule n'a considéré comme pertinent un événement qu'il ne pouvait pas rencontrer. Finalement, pour les 5 autres événements, les conclusions suivantes ont pu être obtenues :

- ◊ *Pour l'événement 1, situé sur une route droite juste après une légère courbe, tous les véhicules qui ont rencontré l'événement ont affiché une alerte au conducteur et plus de 93% ont eu une alerte plus de 30 secondes en avance. Dans cette configuration, les véhicules prenant la route alternative juste avant l'événement affichent aussi une alerte. Cela n'est pas surprenant puisque l'événement est très proche de la bifurcation. C'est pourquoi, tous les véhicules prenant cette bifurcation ont affiché une alerte bien avant de prendre la route alternative.*

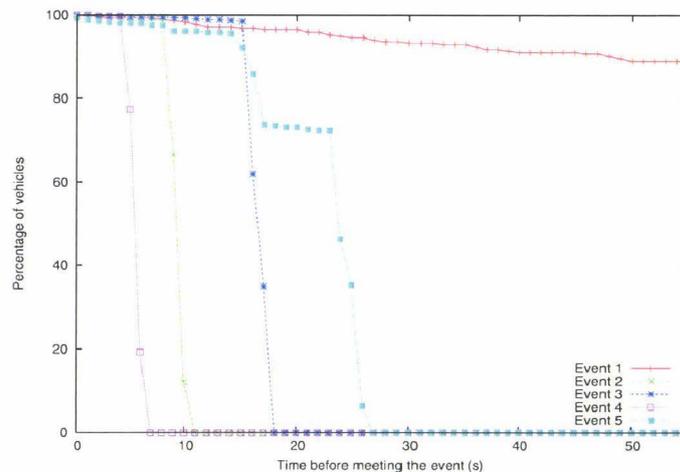


FIGURE 6.8 – Pourcentage de véhicules informés.

Heureusement, environ 2 secondes après avoir tourné à droite et après réévaluation de l'événement 1, ces véhicules ne considèrent plus l'événement comme étant pertinent.

- ◊ L'événement 2 est situé dans un virage en épingle (voir la figure 6.6 (page 92)). Le but ici était de tester le système dans une situation très difficile. Pourtant, le pourcentage de véhicules ayant transmis une alerte reste très important. De plus, l'alerte est transmise près de 10 secondes avant que les véhicules ne rencontre l'événement. Cela vient du fait que même si l'alerte est produite tardivement, la configuration de la route impose aux automobilistes de ralentir, offrant du temps supplémentaire avant d'arriver sur le lieu de l'événement.
- ◊ Les résultats obtenus pour l'événement 3 confirment les bons résultats de l'événement 1.
- ◊ Concernant l'événement 4, les résultats sont les moins bons avec seulement 4 secondes d'anticipation. Cet événement est situé après un fort changement de direction, ce qui est une situation très délicate pour notre solution. De ce fait, le système ne peut prévoir suffisamment à l'avance que les véhicules vont vers l'événement. L'alerte n'est produite qu'une fois que les véhicules sont largement engagés dans la courbe. De plus, ici, contrairement à l'événement 2, le virage n'est pas très serré. Cela a pour conséquence que les véhicules ne ralentissent pas autant que l'événement 2. C'est pourquoi, ici, il n'y a pas le gain de quelques secondes que nous observons pour l'événement 2. Face à ces résultats, nous avons donc mis en place un mécanisme de vecteurs adaptatifs. Au lieu de calculer les vecteurs de mobilité et de direction de façon arbitraire, nous les calculons en fonction de l'événement considéré. Au départ, nous prenons une position de référence distante de 500 mètres pour la mobilité et de 30 mètres pour la direction. Avec les vecteurs adaptatifs, la distance pour les positions de références n'est plus fixée à l'avance mais est calculée par rapport à l'événement considéré. Cela signifie que si un événement se trouve à 200 mètres, les positions de références seront distantes de 200 mètres. Au fil de son déplacement, le véhicule recalcule des nouveaux vecteurs avec sa

nouvelle position. Autrement dit, s'il se rapproche, il va recalculer des vecteurs avec des positions de références plus proches. Cela permet de diminuer l'impact du changement de direction en améliorant les vecteurs chaque fois que le véhicule se rapproche de l'événement. En effet, plus le véhicule est proche de l'événement, plus les vecteurs sont précis puisqu'ils sont basés sur des positions de références plus proches. Grâce à cette technique, l'alerte est générée 8 secondes avant de rencontrer l'événement. Ainsi, les conducteurs ont plus de temps pour réagir. La différence entre l'utilisation de vecteurs statiques et de vecteurs adaptatifs est présentée dans la figure 6.9. Dans cette figure, nous montrons le gain des vecteurs adaptatifs pour les *événements 2 et 4*. Nous voyons bien l'intérêt d'utiliser ces vecteurs adaptatifs, surtout pour les situations avec d'importantes courbes et donc, d'importants changements de direction. Pourtant, l'utilisation de ces vecteurs a un coût. En effet, alors que les vecteurs statiques sont calculés une fois pour tous les événements, les vecteurs adaptatifs imposent d'être calculés pour chaque événement considéré. Le véhicule devra donc calculer autant de vecteurs que d'événements alors qu'avec les vecteurs statiques, quelque soit le nombre d'événements, un seul vecteur de mobilité et un seul vecteur de direction sont calculés. C'est pourquoi le traitement de ces vecteurs représente donc un coût supplémentaire, surtout si l'on considère un grand nombre d'événements.

- ◊ L'événement 5 est considéré pertinent plus tôt que les événements 2, 3 et 4. Cela vient du fait que la direction ne pénalise pas la pertinence dans le cas d'un événement indépendant de la direction.

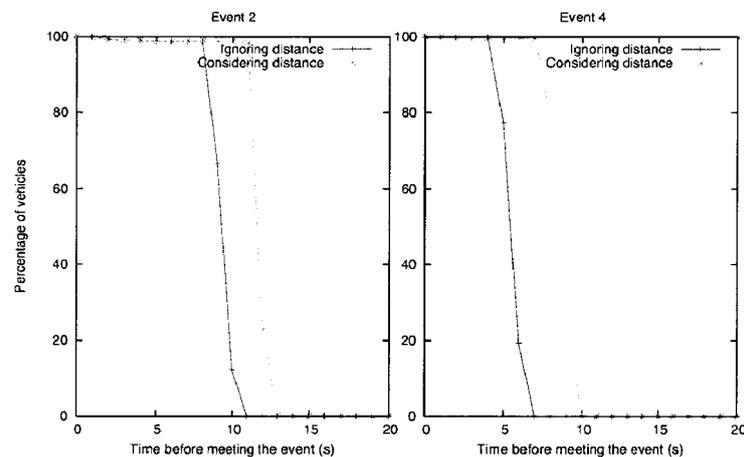


FIGURE 6.9 – Avantage de l'utilisation de vecteurs adaptatifs.

Pour résumer, les résultats obtenus sont largement satisfaisants puisque les véhicules ont tous présenté une alerte à leur conducteur avant que le véhicule n'arrive sur l'événement, quelque soit la configuration de la route.

### 3.2 Evaluation du protocole de dissémination

Dans la suite, nous nous intéressons aux résultats concernant le protocole de dissémination. Le premier résultat observé est que tous les véhicules reçoivent bien les événements avant de les rencontrer, quelque soit la position de l'événement considéré. Dans les expérimentations présentées ici, nous considérons deux critères d'évaluation :

1. Le nombre de messages diffusés en utilisant notre protocole pour vérifier que nous évitons l'inondation.
2. Le temps nécessaire pour atteindre les véhicules pour évaluer l'impact du délai d'attente que nous avons introduit.

Nous avons comparé notre solution avec un mécanisme d'inondation et un mécanisme d'inondation périodique. Le mécanisme d'inondation simple provoque la perte de certains événements lorsqu'il n'y a pas de véhicule pour relayer l'information. L'inondation périodique permet d'éviter cela en considérant qu'une fois qu'un véhicule possède une information, il la diffuse de manière périodique.

La figure 6.10 montre le nombre total de messages diffusés à chaque seconde à partir de la génération de l'événement dans un environnement ayant une faible densité de véhicules (environ un véhicule tous les 100 mètres). La figure 6.11 montre la même chose pour un environnement ayant une forte densité de véhicules (environ un véhicule tous les 10 mètres). De l'observation de ces figures, nous pouvons constater :

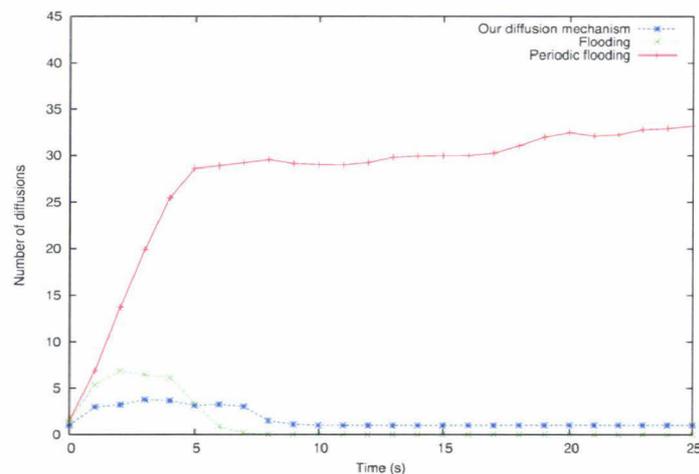


FIGURE 6.10 – Evolution des messages disséminés dans un environnement peu dense.

- ◇ Notre mécanisme de dissémination réduit fortement le nombre de messages diffusés.

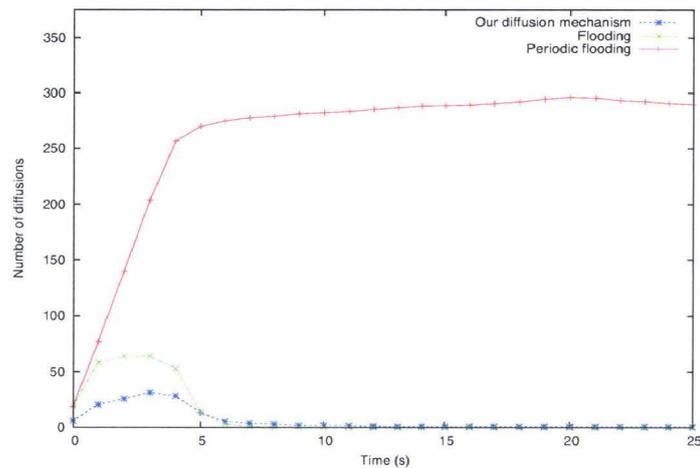


FIGURE 6.11 – Evolution des messages disséminés dans un environnement très dense.

- ◇ Que ce soit avec notre protocole de dissémination ou l'inondation simple, le nombre de messages diffusés décroît après quelques secondes. Cela vient du fait que la *queue de la chaîne de dissémination* a été atteinte. Ce que nous appelons la *queue de la chaîne de dissémination*, c'est l'endroit où les véhicules commencent à ne plus considérer un événement comme pertinent.
- ◇ Avec l'inondation traditionnelle, le nombre de messages atteint zéro après quelques secondes. Cela signifie que l'information cesse de se propager et par conséquent, qu'aucun nouveau véhicule n'est informé de la présence de l'événement. C'est pourquoi, il est nécessaire d'utiliser de l'inondation périodique.
- ◇ Avec l'approche de dissémination proposée, la diffusion des messages continue à la *queue de la chaîne* (même si cela n'est pas très visible sur la figure 6.11 à cause de l'échelle utilisée). En effet, quelques messages sont toujours diffusés tant que la probabilité de rencontre est suffisamment élevée. C'est pourquoi, le problème de l'inondation simple ne se pose pas pour notre protocole de dissémination.

Nous venons de montrer que notre proposition permettait de limiter le nombre de messages diffusés. Pourtant, notre solution introduit un temps d'attente avant la rediffusion. Par conséquent, l'utilisation de notre solution risque d'avoir un impact sur les temps nécessaires à la dissémination des informations. Des expérimentations pour mesurer ce temps additionnel ont été menées. Les figures 6.12 (pour l'environnement peu dense) et 6.13 (pour l'environnement très dense) montrent les temps nécessaires aux véhicules pour recevoir l'information en fonction de la distance séparant ces véhicules de l'événement. Nous avons évalué notre approche avec l'inondation simple. L'inondation périodique ne sert qu'à maintenir l'information dans les cas où aucun véhicule n'est présent pour la relayer. Pour l'évaluation que nous réalisons ici, l'inondation périodique est totalement équivalente

à l'inondation simple.

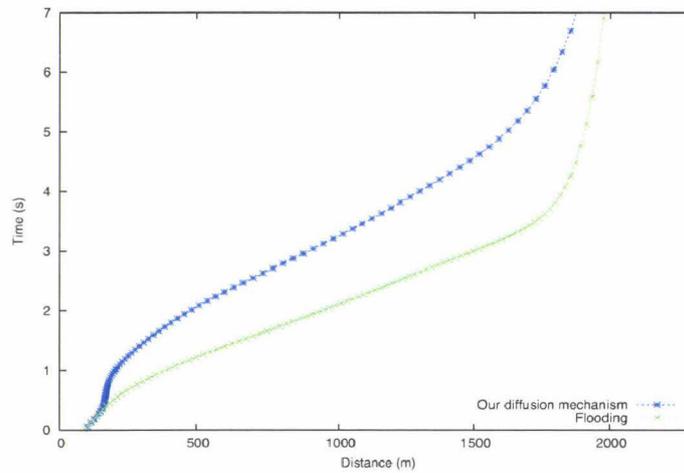


FIGURE 6.12 – Temps nécessaire pour atteindre la queue de la chaîne de dissémination en environnement peu dense.

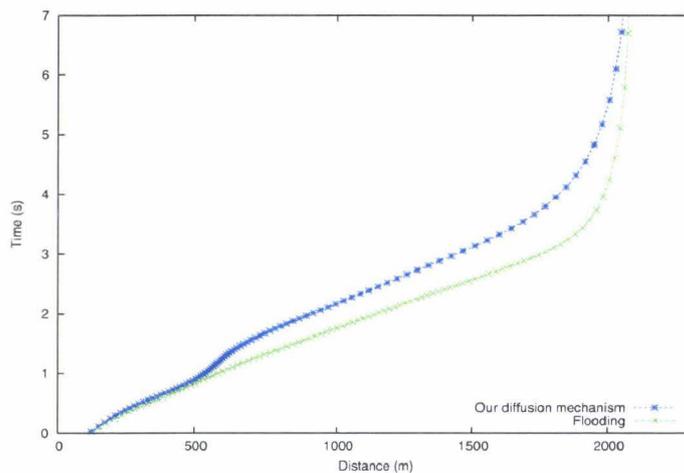


FIGURE 6.13 – Temps nécessaire pour atteindre la queue de la chaîne de dissémination en environnement très dense.

Nous voyons qu'en utilisant notre mécanisme de dissémination, les véhicules reçoivent l'information avec un léger retard par rapport aux autres solutions. Cependant, le temps additionnel est très limité, même dans le pire des cas (c'est-à-dire lorsque la densité de véhicules est faible). Le temps laissé au véhicule pour prendre en compte l'événement et donc au conducteur pour réagir face à la situation reste suffisant, même dans le cas d'un freinage d'urgence. A 180 km/h, un vé-

hicule parcourt 50 mètres par seconde. Dans le pire des cas, notre solution permet de remonter l'information à 500 mètres en environ 2 secondes, ce qui laisse 8 secondes et 400 mètres à l'automobiliste pour réagir. Normalement, un véhicule roulant à 180km/h a besoin de 210 mètres pour s'arrêter.

Notre protocole de dissémination est donc efficace et répond à nos attentes. D'une part, il permet de limiter le nombre de messages diffusés et d'autre part, il reste suffisamment rapide, même pour des situations d'urgences. Par contre, il ne s'intéresse qu'à la dissémination d'informations où le but est de prévenir un maximum de véhicules en un minimum de temps. Dans certains cas, il est nécessaire d'être plus spécifique, notamment pour les ressources. Ici, prévenir tous les véhicules n'a pas d'intérêt, mis à part pour créer encore plus de compétitions pour l'obtention de la ressource. Pour répondre à ce problème, nous avons proposé un protocole de réservation et son évaluation est présentée dans la suite.

### 3.3 Evaluation du protocole de réservation

Concernant le protocole de réservation, nous avons simulé le comportement des automobilistes dans un parking. La figure 6.14 montre l'environnement que nous avons utilisé. Le parking possède 60 places. Les véhicules démarrent à partir d'une entrée du parking et commencent à chercher une place de stationnement. Ils roulent à une vitesse de 10 km/h. Une fois qu'une place de stationnement est trouvée, les véhicules restent garés un certain temps puis libèrent leur place et se dirigent vers la sortie. Chaque fois qu'un véhicule libère sa place de stationnement, il diffuse un message informant qu'une place de stationnement s'est libérée et la position de cette place de stationnement.

Pour correspondre à la réalité, les conducteurs peuvent prendre une place de stationnement lorsqu'ils en voient une. Autrement dit, si un véhicule passe à côté d'une place libre, il peut se garer, même s'il n'a pas reçu l'information concernant cette place de stationnement. Chaque véhicule en recherche de place de stationnement se déplace donc aléatoirement dans le parking jusqu'à trouver une place. Lorsqu'il reçoit une information sur une place qui vient de se libérer, il se dirige alors vers cette place en utilisant le plus court chemin. Si sur son chemin, il trouve une autre place, il prend cette place. Autrement dit, les conducteurs ne privilégient pas les informations reçues.

La figure 6.14 montre l'interface graphique de notre simulateur. Les voitures sont colorées en fonction de leur état. Les véhicules noirs sont garés. Les véhicules bleus viennent de libérer leur place et se dirigent vers la sortie. Les véhicules en recherche ont une couleur qui dépend de leur temps de recherche. Plus ils sont en recherche depuis longtemps, plus ils sont rouges. Cela nous permet d'évaluer, visuellement, la qualité de notre système.

#### Critères et stratégies évalués

Concernant la recherche de places de stationnement, nous avons choisi d'évaluer les différentes propositions suivant certains critères qui nous semblent les plus importants lorsqu'il s'agit de

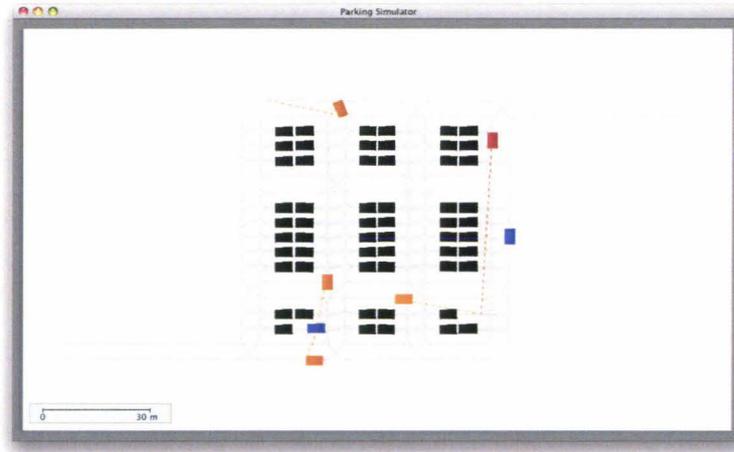


FIGURE 6.14 – Environnement de simulation d'un parking.

trouver une place de stationnement. Tout d'abord, nous avons évalué le temps moyen nécessaire à l'obtention d'une place de stationnement. L'objectif est d'évaluer combien de temps un véhicule recherche une place de stationnement avant d'en trouver une. Ensuite, nous avons évalué l'écart type des temps de recherche d'une place de stationnement. Cela a une certaine importance si nous voulons être équitable. Dans l'idéal, il faudrait que tous les véhicules mettent autant de temps pour trouver une place de stationnement. Il faudrait éviter d'avoir des véhicules qui trouvent une place très rapidement pendant que d'autres cherchent pendant plusieurs heures. Enfin, nous avons évalué le pourcentage d'informations pertinentes transmises aux conducteurs. C'est-à-dire le nombre de places de stationnement transmises au conducteur avant qu'il en trouve effectivement une. En effet, ici, tous les véhicules peuvent prendre une place de stationnement et il est donc fortement probable qu'un véhicule soit informé de la libération d'une place de stationnement mais que celle-ci soit prise une fois arrivé à l'endroit où la place a été libérée. Contrairement au cas des événements où l'important est d'être prévenu, ici, l'important est de fournir une place de stationnement qui soit réellement libre. Dans le cas contraire, c'est une source d'insatisfaction et il est compréhensible qu'un système fournissant peu d'informations pertinentes ne soit pas utilisé. Il est donc très important de faire en sorte que les informations fournies soient valides. Ce critère d'évaluation est donc d'une importance capitale dans ces simulations.

Ensuite, nous avons évalué la recherche de place de stationnement en suivant plusieurs stratégies :

- ◇ La première ne considère aucune communication entre les véhicules. Les automobilistes cherchent une place de stationnement aléatoirement et se garent lorsqu'ils en voient une. Cette stratégie est appelée « View only »
- ◇ « VESPA » est la stratégie considérant un échange d'informations mais aucun protocole de réservation et aucun message d'invalidation. Chaque libération génère un événement et tous les véhicules le recevant estiment sa pertinence avec notre probabilité de rencontre.

- ◇ « VESPA with invalidation messages » est la même que la précédente mis à part que lorsqu'une place est prise, un message d'invalidation est disséminé pour informer les véhicules se dirigeant vers cette place qu'elle est devenue occupée. Cette stratégie permet d'informer de l'état exact de la place de stationnement. L'objectif est de voir s'il est réellement nécessaire de savoir quand une place est prise pour que notre système fonctionne.
- ◇ « VESPA with reservation protocol » est une stratégie où la libération d'une place génère un événement mais les véhicules recevant et cherchant une place s'inscrivent auprès du générateur de l'événement. C'est alors le générateur qui détermine qui doit avoir la place en utilisant la probabilité de rencontre. Seul le véhicule élu transmet l'information à son conducteur.
- ◇ « Distance and Time based relevance » est une stratégie considérant un échange d'informations mais aucun protocole de réservation et aucun message d'invalidation. Chaque libération génère un événement et tous les véhicules le recevant estiment sa pertinence mais au lieu d'estimer la pertinence avec notre probabilité de rencontre, ils estiment la pertinence simplement par rapport à la distance et au temps de l'événement.
- ◇ « Distance and Time based relevance with invalidation messages » est la même que la précédente mis à part que lorsqu'une place est prise, un message d'invalidation est disséminé pour informer les véhicules se dirigeant vers cette place qu'elle est devenue occupée.
- ◇ « Distance and Time based relevance with reservation protocol » est une stratégie où la libération d'une place génère un événement mais les véhicules recevant et cherchant une place s'inscrivent auprès du générateur de l'événement. C'est alors le générateur qui détermine qui doit avoir la place en utilisant une fonction de pertinence basée seulement sur la distance et le temps. Seul le véhicule élu transmet l'information à son conducteur. Cette stratégie représente un défi pour nous puisqu'elle permet de montrer que notre protocole de réservation peut être appliqué à d'autres systèmes que VESPA.

## Résultats

Nous avons évalué les différentes stratégies avec un parking où le nombre de places libres est plus important que le nombre de véhicules en recherche. Nous ne présentons pas les résultats puisque dans cette configuration, il n'y a pas de différence significative entre les différentes stratégies. Par contre, dans le cas d'un parking surchargé, les résultats sont très intéressants. C'est d'ailleurs le cas le plus intéressant puisque les places sont rares et il est donc très important de bien les allouer. Pour cette configuration, nous avons considéré un nombre de véhicules en recherche entre 1% et 10% de la capacité totale du parking. Les coefficients de pénalité utilisés sont :  $\alpha = \frac{1}{50}$ ,  $\beta = \frac{1}{30000}$ ,  $\gamma = \frac{1}{600}$  et  $\zeta = 0$ . Cela permet de privilégier les places se trouvant dans la même rangée (une grande valeur pour  $\alpha$  permet de privilégier les événements en face plutôt que sur les côtés). Nous avons utilisé les mêmes coefficients  $\alpha$  et  $\beta$  pour les stratégies « Distance and time based relevance ». En effet, dans notre configuration, la pénalité doit être plus importante

pour la distance que pour l'âge puisqu'il y a un grand nombre de véhicules en recherche. Autrement dit, c'est bien la proximité des véhicules qui doit faire la différence dans l'évaluation de la pertinence. Concernant le protocole de dissémination, nous avons utilisé notre solution avec une valeur  $D = 0.5$  secondes et une portée de communication de 200 mètres.

La figure 6.15 montre les résultats moyens obtenus après 20 simulations. Les critères évalués sont ceux présentés précédemment : le temps moyen nécessaire pour trouver une place, l'écart type des temps de recherche et le pourcentage d'informations utiles (le pourcentage des places fournies par le système et réellement prises par le conducteur).

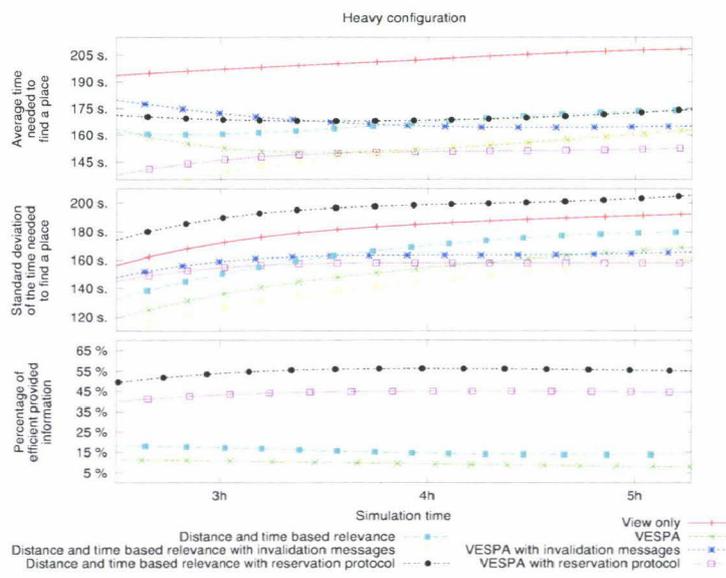


FIGURE 6.15 – Résultats des expérimentations pour le protocole de réservation.

Concernant les résultats, les temps moyens nécessaires à l'obtention d'une place de stationnement montrent l'intérêt de la communication entre les véhicules. En effet, toutes les stratégies basées sur la communication V2V sont meilleures que la stratégie « View only ». Ensuite, nous voyons que l'approche « VESPA » offre de meilleurs résultats que l'approche « Distance and time based relevance ». Cela vient du fait que la probabilité de rencontre permet de privilégier les places de stationnement dans la même rangée. En effet, une place peut-être plus proche en terme de distance mais plus éloignée en terme de temps si pour l'atteindre, il faut changer de rangée ou faire un demi-tour.

Pour fournir des informations plus pertinentes, nous avons introduit des messages d'invalidation. Les véhicules se dirigeant vers une place déjà occupée peuvent ainsi être avertis. L'utilisation de ces messages d'invalidation permet d'améliorer la stratégie « Distance and time based relevance » puisque cela permet d'éviter la perte de temps lorsqu'un véhicule se dirige vers une place déjà prise. Cependant, notons que ces messages risquent d'avoir un impact sur la charge réseau. En effet, ils requièrent la génération d'un message supplémentaire à chaque fois qu'une place est

prise. De plus, ces messages doivent être diffusés et rediffusés suffisamment loin pour atteindre tous les véhicules précédemment informés de la présence de la place de stationnement.

Pour ce qui est des stratégies utilisant le protocole de réservation, les temps moyens d'obtention d'une place sont plutôt bons lorsque c'est la probabilité de rencontre qui est utilisée. En effet, cela est normal puisque nous considérons que tous les conducteurs peuvent prendre une place s'ils la voient. Il est donc très important d'affecter la place au véhicule le plus proche, circulant dans le bon sens de circulation et dans la bonne rangée. Si un autre véhicule est choisi, le risque que la place soit prise par un autre véhicule est plus important. Cela justifie l'écart type de la stratégie « Distance and Time based relevance with reservation protocol ». En effet, les véhicules pour lesquels une place dans une rangée voisine a été affectée perdent du temps en se rendant jusqu'à cette place qui, en plus, a de fortes chances d'être déjà prise. Dans certains cas, certains véhicules subissent plusieurs fois ce phénomène et c'est la raison pour laquelle l'écart type n'est pas bon pour cette stratégie.

Le pourcentage d'informations utiles nous donne des résultats très intéressants. Tout d'abord, il apparaît que les stratégies utilisant le protocole de réservation ont de bons résultats alors que les autres stratégies ne sont pas satisfaisantes. Cela est évident puisque les stratégies sans protocole de réservation informent potentiellement un grand nombre de véhicules alors qu'un seul ne peut avoir la place de stationnement. Si 10 véhicules reçoivent l'information et la considère pertinente, 9 seront insatisfaits et nous ne pouvons pas avoir plus de 10% d'informations utiles. Par contre, pour les stratégies reposant sur le protocole de réservation, nous arrivons à un taux d'informations utiles d'environ 50%. De plus, un élément très intéressant est que la stratégie « Distance and Time based relevance with reservation protocol » possède de meilleurs résultats ici. En confrontant ce résultat avec le temps moyen d'obtention d'une place de stationnement et l'écart type, il apparaît que cette stratégie parvient à fournir une bonne qualité des informations mais que lorsqu'elle se trompe, cela est très pénalisant. Cela confirme donc que certains véhicules perdent beaucoup de temps lorsqu'une mauvaise place leur est attribuée et nous voyons ici l'importance de privilégier les places qui sont dans la même rangée.

## 4 Conclusion

Dans ce chapitre, nous avons d'abord présenté notre prototype puis notre simulateur. Grâce à eux, nous avons pu évaluer aussi bien notre mécanisme d'évaluation de la pertinence, basé sur la probabilité de rencontre, que notre protocole de dissémination. Les résultats obtenus montrent l'utilité de baser la pertinence des informations sur la probabilité de rencontre. Malgré la non-utilisation de données cartographiques, notre probabilité de rencontre permet d'informer les conducteurs suffisamment en avance des événements qu'ils risquent de rencontrer. Concernant notre protocole de dissémination, il est particulièrement adapté au contexte inter-véhiculaire. Nous avons montré d'une part qu'il permettait de limiter fortement le nombre de messages diffusés et donc, d'économiser la bande passante. D'autre part, il assure la pérennité des événements tant que

ceux-ci sont considérés comme pertinents, ce qui était l'objectif. Enfin, notre protocole de dissémination induit un temps pour que la dissémination soit réalisée mais ce temps n'est pas pénalisant puisque la dissémination reste suffisamment rapide pour que les conducteurs soient informés des événements pertinents.

Enfin, nous avons évalué notre protocole de réservation pour les ressources. Nous avons montré l'importance d'utiliser un tel protocole avec l'attribution des places de stationnement. Le plus important étant de diminuer les temps moyens de recherches de places de stationnement tout en garantissant un taux d'informations utiles suffisamment élevé pour ne pas rendre le système inexploitable.

Ainsi, les différents développements que nous avons réalisés permettent de valider nos solutions et sont autant d'outils qui s'amélioreront avec VESPA.

# 7

---

## Conclusion et Perspectives

---

Nous arrivons au terme de ce manuscrit et il est temps de tirer les dernières conclusions. Nous verrons ensuite comment ces travaux peuvent trouver une suite au travers des nombreuses perspectives envisageables, certaines étant déjà en cours de réalisation.

### 1 Conclusion

Nous avons d'abord vu dans le chapitre 2 (page 7) que de nombreux projets ont permis d'envisager des solutions visant à assister les conducteurs. Aujourd'hui, l'accès à l'information est de plus en plus facile et cela permet d'envisager de nouveaux services. Quelque soit la manière utilisée pour accéder aux données, l'intérêt des solutions n'est pas à démontrer. Pourtant, toutes les solutions présentées sont spécifiques à un cas d'utilisation. Pendant que certains projets s'intéressent à prévenir les automobilistes des dangers présents sur la route, d'autres se focalisent sur la gestion des places de stationnement. Cependant, aujourd'hui, il est clair que nous devons proposer une solution suffisamment générique pour répondre à toutes les problématiques liées à la conduite.

Nous avons donc présenté notre solution qui se décompose en trois points principaux. Dans le chapitre 3 (page 29), nous avons montré l'importance d'un mécanisme d'évaluation de la pertinence. Notre proposition repose sur l'utilisation de la *probabilité de rencontre*. Notre approche a l'avantage de ne nécessiter aucune donnée cartographique. En ne s'appuyant que sur un historique des positions du véhicule, nous avons défini une fonction permettant efficacement et rapidement de déterminer la probabilité qu'a un véhicule de rencontrer un événement. De plus, grâce aux différentes manipulations mathématiques, nous avons montré que notre fonction n'est pas spécifique à un type d'événement. Nous avons donc proposé un mécanisme générique traitant tous les événements, qu'ils soient dirigés ou non, mobiles ou non. Grâce à cette évaluation de la pertinence, nous permettons à chaque véhicule de ne garder que les informations utiles [DCI08, DCI10].

Bien que cela réduise fortement la quantité d'informations à traiter, le nombre d'événements peut tout de même être important. De plus, pour répondre aux besoins spécifiques de chaque conducteur, nous avons ensuite proposé un mécanisme d'évaluation continue des informations dans le chapitre 4 (page 47). Les informations étant reçues pendant l'utilisation du système, nous avons montré comment les ajouts, les suppressions et les mises à jour sont intégrés pour permettre d'offrir à chaque instant des résultats cohérents. Nous avons ensuite montré comment le déplacement du véhicule était intégré. En effet, la mobilité du véhicule nous oblige à remettre en cause en permanence la validité des informations. L'approche naïve qui consiste à évaluer périodiquement l'ensemble des données n'est pas réalisable si l'on considère des terminaux légers dans nos véhicules. Nous avons donc montré comment prendre en compte la mobilité pour optimiser la réévaluation des données [CD07].

Dans le chapitre 5 (page 63), nous nous sommes finalement intéressés à l'échange de données proprement dit. Nous avons considéré une communication pair-à-pair entre les véhicules, sans infrastructure. C'est-à-dire que nous avons supposé que seuls les véhicules communiquaient entre eux et qu'il n'y avait aucun autre mode de communication. Cette contrainte forte nous a poussés à proposer un mécanisme de dissémination. En effet, l'architecture que nous avons choisie ne nous permet pas de faire appel à des serveurs distants. Notre mécanisme de dissémination consiste donc en la diffusion d'informations par le générateur d'un événement. Les véhicules recevant l'information l'évaluent puis décident si elle doit être rediffusée pour atteindre de nouveaux véhicules. Grâce à l'utilisation de la *probabilité de rencontre*, nous avons montré comment limiter la zone où une information était disséminée. Bien que cela permette de limiter fortement le nombre de messages échangés dans une zone, cela ne constitue que le premier point de notre mécanisme. En effet, dans une zone donnée, le nombre de messages diffusés doit être limité au risque de saturer le réseau et de perdre toutes les informations. Pour répondre à ce problème, notre mécanisme de dissémination désynchronise les envois de messages des véhicules en instaurant un *temps d'attente*. Grâce à ce *temps d'attente*, les véhicules ne diffusent pas les informations en même temps, ce qui limite déjà le risque de collision de messages. Pourtant, nous sommes allés plus loin en permettant à des véhicules d'annuler leur dissémination de messages. En considérant les informations comme des *accusés de réception*, nous avons pu garantir la bonne dissémination des messages sans que tous les véhicules ne les rediffusent. De plus, nous garantissons le maintien d'une information tant

qu'elle est valide. Ainsi, notre mécanisme de dissémination est parfaitement adapté au système que nous voulions proposer [CDI08].

Pour répondre aux problèmes de ressources, comme les places de stationnement, nous avons ensuite décrit notre protocole de réservation [DCIL09]. Ce protocole permet d'attribuer les ressources à des véhicules choisis afin d'éviter les situations de compétition. En effet, contrairement aux événements qui peuvent être exploités par tous les véhicules, les ressources ne peuvent satisfaire qu'un nombre limité de véhicules. Ainsi, notre protocole de réservation permet d'éviter de diriger toute une flotte de véhicules vers une ressource qui ne pourra satisfaire que le premier arrivé.

Pour valider nos contributions, nous avons présenté, dans le chapitre 6 (page 83), un prototype, un simulateur et nos résultats. Le prototype nous a permis de montrer la faisabilité d'un système de partage d'informations entre les véhicules. Ensuite, pour considérer un nombre suffisamment important de véhicules, nous avons utilisé notre simulateur. Grâce à lui, nous avons pu valider nos contributions en évaluant les critères qui nous semblaient les plus pertinents.

Pourtant, de nombreuses améliorations sont encore possibles.

## 2 Perspectives

Bien que les travaux présentés ici montrent des contributions importantes pour le partage d'informations dans les réseaux de communication inter-véhiculaire, de nombreux éléments peuvent être améliorés ou ajoutés. Tout d'abord, concernant la *probabilité de rencontre*, la fonction que nous avons choisie est bien adaptée dans la mesure où nous n'utilisons pas de données cartographiques. Pourtant, il serait intéressant de réfléchir à l'impact de l'utilisation de données cartographiques sur notre système. En effet, nous pourrions évaluer de manière plus fine la *probabilité de rencontre* en connaissant les chemins menant aux différents événements. Les données cartographiques nous permettraient de connaître les positions futures des véhicules avec plus de précision. Ensuite, nous pourrions utiliser les données cartographiques pour déterminer jusqu'où une information doit être remontée. En effet, dans le cas d'un embouteillage sur autoroute par exemple, il serait intéressant de connaître les différentes sorties permettant d'éviter l'embouteillage. L'information pourrait alors être relayée suffisamment loin pour donner l'opportunité aux automobilistes de prendre une sortie. Pour l'instant, avec la *probabilité de rencontre* telle que nous l'avons définie, il est impossible de garantir qu'une sortie est présente.

Ensuite, les coefficients de pénalité utilisés dans la *probabilité de rencontre* doivent être étudiés plus rigoureusement. Nous avons montré que notre fonction de *probabilité de rencontre* était valide mais qu'elle était grandement influencée par les *coefficients de pénalité*. Pour l'instant, nous les définissons de manière statique. Il serait très intéressant de proposer un mécanisme permettant de déterminer ces coefficients. En effet, même si pour chaque situation, il existe la meilleure

combinaison de *coefficients de pénalité*, il n'existe pas une meilleure combinaison unique qui convienne à toutes les situations. Il serait donc très intéressant de réaliser une étude du *contexte* et de voir dans quelle mesure nous pourrions exploiter ces informations pour définir, dynamiquement, les *coefficients de pénalité*. Par exemple, la vitesse du véhicule peut nous donner une indication sur le type de route sur lequel il se trouve. Le jour et l'heure peuvent nous donner une information sur la densité de véhicules. Pendant la nuit d'un mardi au mercredi, il y a moins de véhicules sur les routes qu'un samedi dans l'après-midi. Toutes ces informations peuvent certainement être exploitées pour déterminer, ou tout du moins ajuster, les *coefficients de pénalité*.

Le mécanisme d'évaluation continue que nous avons proposé peut aussi être amélioré et étendu. Le principe de prendre en considération les modifications de l'ensemble de données et le déplacement du véhicule est une première étape. Il serait très intéressant maintenant d'étendre ce mécanisme à toutes sortes de prédicats. Pour l'instant, nous nous sommes intéressés aux prédicats spatiaux et temporels. Nous avons considéré le déplacement du véhicule comme étant l'élément à prendre en compte pour remettre en cause l'évaluation précédente. Il serait intéressant de définir pour chaque prédicat (spatial, temporel ou autres) les éléments qui peuvent interagir avec l'évaluation continue de ces prédicats. Cela nous permettrait d'étendre le mécanisme à toutes sortes d'informations dynamiques.

Concernant le protocole de dissémination, nous utilisons la *probabilité de rencontre* pour déterminer les véhicules qui doivent rediffuser l'information. La manière dont nous le faisons nous empêche d'utiliser le principe de *transporter et retransmettre*. Pourtant, ce principe est très intéressant, surtout lorsqu'il y a peu de véhicules. En utilisant une fonction légèrement différente de la *probabilité de rencontre* ou en utilisant des *coefficients de pénalité* différents, il serait probablement possible d'intégrer ce principe à notre solution.

D'autres perspectives concernent la gestion des données. Dans notre approche, les données sont consommées puis jetées. Pourtant, ces données peuvent encore servir, d'autant plus que ce n'est pas parce qu'un véhicule ne reçoit pas une information sur un événement que cet événement n'existe pas. Autrement dit, il peut arriver qu'un événement ne soit pas diffusé. Cela peut être dû au fait que les véhicules précédents ne sont pas équipés ou bien simplement que l'information n'a pas été relayée suffisamment loin. Dans tous les cas, la répétition d'événements à un endroit donné peut signifier beaucoup de choses. Par exemple, si tous les jours à une certaine heure, un embouteillage est constaté à un endroit donné, il n'est peut-être pas nécessaire d'attendre de recevoir l'information pour prévoir la présence de cet embouteillage. En utilisant des *techniques d'agrégation*, chaque véhicule pourrait se constituer une sorte de base de connaissances. Ainsi, sans recevoir d'information, un véhicule pourrait « deviner » qu'il y a un embouteillage à un certain endroit. De la même manière, sans recevoir d'information sur une place de stationnement disponible, un véhicule pourrait se diriger vers une zone où, habituellement, il y a des places disponibles. Même les portions de routes accidentogènes pourraient être détectées et ainsi évitées.

Si nous arrivons à constituer ces bases de connaissances pour chaque véhicule, il sera alors intéressant de les échanger entre les véhicules. Pourtant, à l'heure actuelle, nous avons considéré

des temps de communication très courts. Il serait dès lors intéressant de détecter des groupes de véhicules circulant sur une même route et par conséquent, pouvant communiquer plus longtemps. Ces véhicules pourraient s'échanger une quantité plus importante de données et en particulier, leur base de connaissances. De cette manière, nous aurions sur chaque véhicule une très bonne connaissance de la route. De plus, cette connaissance serait en parfaite adéquation avec la réalité, ce qui ne serait pas possible si l'on considère une centralisation des données sur des serveurs distants.

Dans le même esprit que la constitution d'une base de connaissances à partir des événements reçus, une autre perspective serait de construire dynamiquement des cartes dans chaque véhicule. En effet, nous avons montré comment des données cartographiques pourraient nous aider à améliorer notre *probabilité de rencontre*. Pour obtenir ces données, nous pouvons utiliser les bases de données cartographiques déjà existantes. Cependant, la mise à jour de ces données n'est pas aisée. Une alternative serait que chaque véhicule se construise sa propre carte. En s'appuyant sur les groupes de véhicules, ils pourraient même échanger leurs données cartographiques et ainsi posséder la carte la plus à jour de l'environnement dans lequel ils se trouvent. De plus, chaque automobiliste réalise généralement des trajets quotidiens. Ces trajets pourraient être détectés et ainsi, nous pourrions proposer un système de navigation qui ne demande aucune destination à l'utilisateur. Pour cela, il serait nécessaire de proposer un mécanisme de détection du chemin sur lequel l'automobiliste se trouve. Finalement, en partant d'un système vide, nous pourrions, au fil des jours, fournir à l'utilisateur les meilleurs itinéraires pour les déplacements quotidiens.

La gestion des *ressources* est aussi une problématique qui demande encore beaucoup de recherches. Notre protocole de réservation repose sur la diffusion préalable d'un événement. C'est-à-dire qu'il y a d'abord mise à disposition d'une *ressource*. Il serait intéressant de renverser le problème et de considérer les véhicules en recherche d'une *ressource* comme des événements. Autrement dit, au lieu d'avoir une diffusion des *ressources*, nous aurions des événements mobiles dirigés, comme les véhicules d'intervention d'urgence. Les véhicules en recherche d'une *ressource* diffuseraient alors un événement spécifiant leur recherche. Les *coordinateurs* des ressources pourraient alors les allouer de la même manière que nous le faisons actuellement.

Pour finir sur les perspectives, nous ne devons pas oublier la possibilité d'utiliser une infrastructure. En effet, pour l'instant, nous avons considéré un système totalement autonome. Il serait intéressant de voir comment une infrastructure pourrait nous aider. Dans un premier temps, cette infrastructure pourrait être composée par des terminaux disséminés le long des routes. Ces terminaux agiraient alors comme des véhicules immobiles. Ces relais seraient très utiles dans le cas où peu de véhicules sont sur les routes. Mais l'infrastructure pourrait peut-être aussi fournir des accès à des serveurs distants. Il faudrait alors s'interroger sur la manière d'utiliser ces accès et sur la manière d'intégrer les données de ces serveurs dans notre système. Dans l'éventualité où aucune infrastructure n'offre cet accès, nous pourrions tout de même nous poser ces questions avec l'utilisation des réseaux téléphoniques.

Ainsi, même si cette thèse s'achève ici, les perspectives restent très nombreuses. Des travaux de recherche ont déjà commencé, particulièrement sur *l'intégration de données cartogra-*

*phiques [DICH09], l'adaptation du protocole de dissémination, l'agrégation des données [DDIC08] et la prise en compte du contexte.*

---

# Bibliographie

---

- [ABD<sup>+</sup>06] A. Aijaz, B. Bochow, F. Dötzer, A. Festag, M. Gerlach, R. Kroh, and T. Leinmuller. Attacks on inter vehicle communication systems—an analysis. In *Proceedings of the 3rd International Workshop on Intelligent Transportation, WIT 2006*, Hamburg, Germany, 14 – 15 mar 2006.
- [ARP04] M.M. Artimy, W. Robertson, and W.J. Phillips. Connectivity in inter-vehicle ad hoc networks. In *IEEE Engineering Canadian Conference on Electrical and Computer*, volume 1, pages 293–298, Niagara Falls, Ontario, Canada, 2–5 May 2004.
- [BCCF02] F. Borgonovo, A. Capone, M. Cesana, and L. Fratta. RR-ALOHA, a Reliable R-ALOHA broadcast channel for ad-hoc inter-vehicle communication networks. In *the proceedings of the First Annual Mediterranean Ad Hoc Networking Workshop, MedHocNet 2002*, Sardegna, Italy, 4–6 Sep 2002.
- [BEH04] Jeremy J. Blum, Azim Eskandarian, and Lance J. Hoffman. Challenges of inter-vehicle ad hoc networks. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 5(N° 4) :pages 347–351, 2004.
- [BFW03] M. Bechler, W.J. Franz, and L. Wolf. Mobile internet access in FleetNet. In *13. Fachtagung Kommunikation in verteilten Systemen*, Leipzig, Germany, 2003.
- [BHM07] R. Baumann, S. Heimlicher, and M. May. Towards realistic mobility models for vehicular ad-hoc networks. In *26th Annual IEEE Conference on Computer Communications IEEE INFOCOM*, Anchorage, Alaska, USA, 6–12 May 2007. Citeseer.
- [BL04] P. Basu and T.D.C. Little. Wireless ad hoc discovery of parking meters. In *WAMES'04 : Proceedings of the MobiSys 2004 Workshop on Applications of Mobile Embedded Systems*, pages 8–11, Boston, Massachusetts, USA, 6 Jun 2004. Citeseer.
- [CBSM07] M. Caliskan, A. Barthels, B. Scheuermann, and M. Mauve. Predicting parking lot occupancy in vehicular ad hoc networks. In *Proc. of the 65th IEEE Vehicular Technology Conference (VTC 2007 Spring)*, pages 277–281, Dublin, Ireland, 22–25 Apr 2007. Citeseer.

- [CD07] N. Cenerario and T. Delot. Evaluation continue de requêtes dans les réseaux de communication inter-véhicules. In *Actes de l'Atelier Gestion de Données dans les Systèmes d'Information Pervasifs (GEDSIP) organisé dans le cadre du XXVème congrès INFORSID*, Perros-Guirec, France, 22–25 May 2007.
- [CDI08] N. Cenerario, T. Delot, and S. Ilarri. Dissemination of information in inter-vehicle ad hoc networks. In *Proc. of the IEEE Intelligent Vehicles Symposium (IV'08)*, Eindhoven, Netherlands, 4–6 Jun 2008.
- [CDI09] N. Cenerario, T. Delot, and S. Ilarri. Towards the use of multimedia contents to represent events in vehicular ad hoc networks. In *Proceedings of the 9th Workshop on Multimedia Metadata (WMM'09)*, Toulouse, France, 19–20 Mar 2009.
- [CGM06] Murat Caliskan, Daniel Graupner, and Martin Mauve. Decentralized discovery of free parking places. In *The Third ACM International Workshop on Vehicular Ad Hoc Networks (VANET 2006)*, pages 30–39, Los Angeles, California, USA, 29 Sep 2006.
- [CWXY05] H. Cao, O. Wolfson, B. Xu, and H. Yin. Mobi-dic : Mobile discovery of local resources in peer-to-peer wireless network. *IEEE Data Engineering Bulletin*, Vol. 28(n°3) :pages 11–18, Sep 2005.
- [dBKvKN04] D. de Bruin, J. Kroon, R. van Klaveren, and M. Nelisse. Design and test of a cooperative adaptive cruise control system. In *IEEE Intelligent Vehicles symposium*, pages 392–396, Parma, Italy, 14–17 jun 2004.
- [DCI08] T. Delot, N. Cenerario, and S. Ilarri. Estimating the relevance of information in inter-vehicle ad hoc networks. In *the International Workshop on Sensor Network Technologies for Information Explosion Era (SeNTIE2008) In Conjunction with International Conference on Mobile Data Management (MDM 2008)*, Beijing, China, 27 Apr 2008.
- [DCI10] T. Delot, N. Cenerario, and S. Ilarri. Vehicular event sharing with a mobile peer-to-peer architecture. *Transportation Research Part C : Emerging Technologies*, In Press, Corrected Proof, 2010. To appear.
- [DCIL09] T. Delot, N. Cenerario, S. Ilarri, and S. Lecomte. A cooperative reservation protocol for parking spaces in vehicular ad hoc networks. In *6th International Conference on Mobile Technology, Applications and Systems (ACM Mobility)*, ACM Digital Library, Best Paper Award, Nice, France, Sep 2009.
- [DDIC08] B. Defude, T. Delot, S. Ilarri, and N. Cenerario. Data aggregation in vanets : the vespa approach. In *First International Workshop on Computational Transportation Science (IWCTS'08), in conjunction with the 5th Annual International Conference on Mobile and Ubiquitous Systems : Networks and Services (MOBIQUITOUS'08)*, Dublin, Ireland, 21 July 2008.
- [DICH09] Thierry Delot, Sergio Ilarri, Nicolas Cenerario, and Thomas Hien. Event sharing in vanets : Evaluating the benefits of using road map information. To be published, dec 2009.

- 
- [Döt05] Florian Dötzer. Privacy issues in vehicular ad hoc networks. In *Workshop on Privacy Enhancing Technologies*, volume 3856, pages 197–209, Dubrovnik, Croatia, 30 May – 1 Jun 2005.
- [ERW<sup>+</sup>03] A. Ebner, H. Rohling, L. Wischhof, R. Halfmann, and M. Lott. Performance of UTRA TDD ad hoc and IEEE 802.11 b in vehicular environments. In *The 57th IEEE Semiannual Vehicular Technology Conference*, volume 2, pages 960–964, Jeju, Korea, 22–25 Apr 2003.
- [FFBA07] E. Fonseca, A. Festag, R. Baldessari, and R. Aguiar. Support of anonymity in vanets—putting pseudonymity into practice. In *IEEE Wireless Communications and Networking Conference, WCNC*, Hong Kong, China, 11–15 Mar 2007.
- [FFH<sup>+</sup>04] A. Festag, H. Füßler, H. Hartenstein, A. Sarma, and R. Schmitz. FLEETNET : Bringing car-to-car communication into the real world. In *Proceedings of the 11th World Congress on ITS*, pages 16–23, Nagoya, Japan, oct 2004.
- [FHB01] Walter J. Franz, Hannes Hartenstein, and Bernd Bochow. Internet on the Road via Inter-Vehicle Communications. In *GI Workshop, Communication over Wireless LANs*, pages 577–584, Vienna, Austria, Sep 2001.
- [FHFB07] M. Fiore, J. Haerri, F. Filali, and C. Bonnet. Vehicular mobility simulation for VANETs. In *Proceedings 40th Annual Simulation Symposium (ANSS-40 2007)*, volume 40, pages 301–309, Norfolk, Virginia, USA, 26–28 Mar 2007. IEEE Computer Society.
- [FJM<sup>+</sup>01] Lino Figueiredo, Isabel Jesus, J. A. Tenreiro Machado, José Rui Ferreira, and J. L. Martins de Carvalho. Towards the development of intelligent transportation systems. In *IEEE 4th International Conference on Intelligent Transportation Systems*, pages 1206–1211, Oakland, California, USA, 25–29 Aug 2001.
- [FMH<sup>+</sup>02] H. Füßler, M. Mauve, H. Hartenstein, M. Käsemann, and D. Vollmer. A comparison of routing strategies for vehicular ad hoc networks. *Department of Computer Science, University of Mannheim, Tech. Rep. TR-02-003*, 2002.
- [FMH<sup>+</sup>03] Holger Füßler, Martin Mauve, Hannes Hartenstein, Michael Käsemann, and Dieter Vollmer. Mobicom poster : location-based routing for vehicular ad-hoc networks. *Mobile Computing and Communications Review*, Vol. 7(n°1) :page 47–49, Jan 2003.
- [GDAD06] R.H. Güting, V.T. De Almeida, and Z. Ding. Modeling and querying moving objects in networks. *The VLDB Journal*, Vol. 15(n°2) :pages 165–190, Jun 2006.
- [GGS04] P. Golle, D. Greene, and J. Staddon. Detecting and correcting malicious data in vanets. In *Proceedings of the 1st ACM international workshop on Vehicular Ad Hoc Networks (VANET 2004)*, pages 29–37, Philadelphia, PA, USA, 1 Oct 2004.
- [GION03] S. Goel, T. Imielinski, K. Ozbay, and B. Nath. Grassroots : A Scalable and Robust Information Architecture. Technical report, Technical Report DCS-TR-523, Department of Computer Science, Rutgers University, 2003.
-

- [GOM08] I. Ganchev, M. O'Droma, and D. Meere. Intelligent Car Parking Locator Service. *International Journal on Information Technologies & Knowledge*, Vol. 2 :166–173, 2008.
- [HČL04] J.P. Hubaux, S. Čapkun, and J. Luo. The security and privacy of smart vehicles. *IEEE Security & Privacy*, Vol. 2(n°3) :49–55, 2004.
- [HFMF03] H. Hartenstein, H. Füßler, M. Mauve, and W. Franz. Simulation results and a proof-of-concept implementation of the fleetnet position-based router. *Lecture notes in computer science*, pages 192–197, 2003.
- [HKB99] W.R. Heinzelman, J. Kulik, and H. Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks. In *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pages 174–185, Seattle, Washington, 15–20 Aug 1999. ACM New York, NY, USA.
- [JMS07] M. Jerbi, P. Marlier, and SM Senouci. Experimental assessment of V2V and I2V communications. In *IEEE International Conference on Mobile Adhoc and Sensor Systems, MASS, Pisa, Italy, 8–11 Oct 2007*.
- [KGMRR+03] S. Katragadda, CNS Ganesh Murthy, MS Ranga Rao, S. Mohan Kumar, and R. Sachin. A decentralized location-based channel access protocol for inter-vehicle communication. In *IEEE Vehicular Technology Conference (VTC Spring 2003)*, volume 3, pages 1831–1835, Jeju, Corea, 21–24 Apr 2003.
- [KML05] F. Karnadi, Z.H. Mo, and K. Lan. Rapid generation of realistic mobility models for vanet. In *The 11th International Conference on Mobile Computing and Networking (ACM MOBICOM 2005)*, Cologne, Germany, 28 Aug – 2 Sep 2005.
- [KTT+02] S. Kato, S. Tsugawa, K. Tokuda, T. Matsui, and H. Fujii. Vehicle control algorithms for cooperative driving with automated vehicles and intervehicle communications. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 3(n°3) :155–161, September 2002.
- [KV99] Y.B. Ko and N.H. Vaidya. Geocasting in mobile ad hoc networks : Location-based multicast algorithms. In *Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications*, page 101, New Orleans, LA, 25–26 Feb 1999.
- [KWZ02] F. Kuhn, R. Wattenhofer, and A. Zollinger. Asymptotically optimal geometric mobile ad-hoc routing. In *Proceedings of the 6th international workshop on Discrete algorithms and methods for mobile computing and communications*, page 33, Atlanta, Georgia, 28 Sep 2002. ACM.
- [LCL10] Y. LIN, Y.U.H.S. CHEN, and S.L. LEE. Routing Protocols in Vehicular Ad Hoc Networks : A Survey and Future Perspectives. *Journal of Information Science and Engineering*, Vol. 26(N°3), May 2010.
- [LCM09] I. Leontiadis, P. Costa, and C. Mascolo. Persistent content-based information dissemination in hybrid vehicular networks. In *Proceedings of the 2009 IEEE International Conference on Pervasive Computing and Communications*, pages 1–10, Galveston, Texas, 9–13 Mar 2009.

- 
- [LH06] J. Luo and J.P. Hubaux. A survey of research in inter-vehicle communications. *Securing Current and Future Automotive IT Applications*, pages 111–122, 2006.
- [LHT<sup>+</sup>03] C. Lochert, H. Hartenstein, J. Tian, H. Füßler, D. Hermann, and M. Mauve. A routing strategy for vehicular ad hoc networks in city environments. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 156–161, Columbus, OH, USA, 9–11 Jun 2003. Citeseer.
- [LLZ<sup>+</sup>08] X. Lin, R. Lu, C. Zhang, H. Zhu, P. Ho, and X. Shen. Security in vehicular ad hoc networks. *IEEE Communications Magazine*, Vol. 46(n<sup>o</sup>4) :pages 88–95, 2008.
- [LM07] I. Leontiadis and C. Mascolo. Opportunistic spatio-temporal dissemination system for vehicular networks. In *Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*, page 46, San Juan, Puerto Rico, 11–14 Jun 2007. ACM.
- [LpH04] Jun Luo and Jean pierre Hubaux. A Survey of Inter-Vehicle Communication. Technical report, School of computer and Communication Sciences, EPFL, July 15 2004.
- [LW08] Y. Luo and O. Wolfson. Mobile p2p databases. *Encyclopedia of GIS*, pages 671–677, Jul 2008.
- [LWX08] Y. Luo, O. Wolfson, and B. Xu. Mobile local search via p2p databases. In *2nd IEEE International Interdisciplinary Conference on Portable Information Devices, 2008 and the 2008 7th IEEE Conference on Polymers and Adhesives in Microelectronics and Photonics. PORTABLE-POLYTRONIC 2008*, pages 1–6, Garmisch-Partenkirchen, Germany, 17–20 Aug 2008.
- [MA08] M.F. Mokbel and W.G. Aref. Sole : scalable on-line execution of continuous queries on spatio-temporal data streams. *The VLDB Journal*, Vol. 17(N<sup>o</sup> 5) :pages 971–995, Aug 2008.
- [Mai04] C. Maihöfer. A survey of geocast routing protocols. *IEEE Communications Surveys and Tutorials*, Vol. 6(n<sup>o</sup>2) :32–42, 2004.
- [MCF<sup>+</sup>03] C. Maihofer, C. Cseh, W. Franz, R. Eberhardt, T. Res, AG DaimlerChrysler, and G. Ulm. Performance evaluation of stored geocast. In *2003 IEEE 58th Vehicular Technology Conference, 2003. VTC 2003-Fall*, volume 5, Orlando, Florida, USA, 6–9 Oct 2003.
- [MCGM02] P. Morsink, C. Cseh, O. Gietelink, and M. Miglietta. Preliminary design of an application for communication based longitudinal control in the cartalk2000 project. In *Proc. of IT Solutions for Safety and Security in Intelligent Transport (e-Safety)*, Lyon, France, Sep 2002.
- [MWH01] M. Mauve, A. Widmer, and H. Hartenstein. A survey on position-based routing in mobile ad hoc networks. *IEEE network magazine*, Vol. 15(n<sup>o</sup>6) :page 30–39, 2001.
- [MWR<sup>+</sup>06] Rahul Mangharam, Daniel Weller, Raj Rajkumar, Priyantha Mudalige, and Fan Bai. GrooveNet : A hybrid simulator for vehicle-to-vehicle networks. In *Second International Workshop on Vehicle-to-Vehicle Communications (V2VCOM'06), in conjunction with the Third Annual International Conference on Mobile and Ubiquitous*
-

- Systems (MOBIQUITOUS'06)*, pages 1–8, San Jose, California, USA, 21 Jul 2006. IEEE Computer Society.
- [MXA04] M.F. Mokbel, X. Xiong, and W.G. Aref. Sina : Scalable incremental processing of continuous queries in spatio-temporal databases. In *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, pages 623–634, Paris, France, 13–18 Jun 2004.
- [NDLI04] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode. Trafficview : Traffic data dissemination using car-to-car communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, Vol. 8(n°3) :6–19, July 2004.
- [NI97] J.C. Navas and T. Imielinski. GeoCast – geographic addressing and routing. In *Proceedings of the 3rd annual ACM/IEEE international conference on Mobile computing and networking*, pages 66–76, Budapest, Hungary, 26–30 Sep 1997.
- [NSI06] T. Nadeem, P. Shankar, and L. Iftode. A comparative study of data dissemination models for vanets. In *International Conference on Mobile and Ubiquitous Systems : Networks and Services (MOBIQUITOUS)*, San Jose, California, 17–21 Jul 2006. Citeseer.
- [PEBS00] P. Pieda, J. Ethridge, M. Baines, and F. Shallwani. A network simulator, differentiated services implementation. Technical report, Open IP, Nortel Networks, 2000.
- [PRL+08] M. Piorkowski, M. Raya, A.L. Lugo, P. Papadimitratos, M. Grossglauser, and J.P. Hubaux. Trans : realistic joint traffic and network simulator for vanets. *ACM SIGMOBILE Mobile Computing and Communications Review*, Vol. 12(n°1) :31–33, Jan 2008.
- [RH05a] M. Raya and J.P. Hubaux. Security aspects of inter-vehicle communications. In *the 5th Swiss Transport Research Conference (STRC)*, Monte Verita, Ascona, 9–11 Mar 2005.
- [RH05b] Maxim Raya and Jean-Pierre Hubaux. The security of vehicular ad hoc networks. In *The Third ACM Workshop on Security of Ad Hoc and Sensor Networks, SASN 2005*, pages 11–21, Alexandria, VA, USA, 7 Nov 2005.
- [RH07] Maxim Raya and Jean-Pierre Hubaux. Securing vehicular ad hoc networks. *Journal of Computer Security, Special Issue on Security of Ad-hoc and Sensor Networks*, Vol. 15(n°1) :39–68, 2007.
- [RYA+09] A. Rahim, M. Yasin, I. Ahmad, ZS Khan, and M. Sher. Relevance based approach with virtual queue for vehicular adhoc networks. In *the 2nd International Conference on Computer, Control and Communication (IC4 2009)*, pages 1–4, Karachi, Pakistan, 17–18 Feb 2009.
- [SG07] Chakkaphong Suthaputchakun and Aura Ganz. Priority based inter-vehicle communication in vehicular ad-hoc networks using ieee 802.11e. In *Proceedings of the 65th IEEE Vehicular Technology Conference, VTC Spring 2007*, pages 2595–2599, Dublin, Ireland, 22–25 Apr 2007.

- 
- [STC02] S. Srivastava, S. Tripathi, and A. Chaturvedi. Resource optimization in cdma based wireless ad hoc networks. Technical report, IIT Kanpur BTech, 2002.
- [SWCD97] A. Sistla, O. Wolfson, S. Chamberlain, and S. Dao. Modeling and querying moving objects. In *Proceedings of the International Conference on Data Engineering*, pages 422–433, Birmingham, UK, 7–11 Apr 1997.
- [SWCD98] PA Sistla, O. Wolfson, S. Chamberlain, and S. Dao. Querying the uncertain position of moving objects. *Temporal Databases : Research and Practice*, Vol. 1399 :pages 310–337, 1998.
- [SWX05] A.P. Sistla, O. Wolfson, and B. Xu. Opportunistic data dissemination in mobile peer-to-peer networks. *Lecture Notes in Computer Science*, Vol. 3633 :pages 346–363, 2005.
- [SYG06] Weihua Sheng, Qingyan Yang, and Yi Guo. Cooperative driving based on inter-vehicle communications : Experimental platform and algorithm. In *the IEEE/RIS International Conference on Intelligent Robots and Systems (IROSS 2006)*, pages 5073–5078, Beijing, China, 9–15 Oct 2006.
- [THRC03] J. Tian, L. Han, K. Rothermel, and C. Cseh. Spatially aware packet routing for mobile ad hoc inter-vehicle radio networks. In *Proc. of the IEEE 6th Intl. Conf. on Intelligent Transportation Systems (ITSC)*, volume 2, pages 1546–1551, 2003.
- [TSWN04] G. Trajcevski, P. Scheuermann, O. Wolfson, and N. Nedungadi. Cat : Correct answers of continuous queries using triggers. In *the 9th International Conference on Extending Database Technology, EDBT 2004*, pages 837–840, Heraklion, Crete, Greece, 14–18 Mar 2004.
- [VW01] M. Vazirgiannis and O. Wolfson. A spatiotemporal model and language for moving objects on road networks. In *the 7th International Symposium on Advances in Spatial and Temporal Databases, SSTD 2001*, pages 20–35, Redondo Beach, CA, USA, 12–15 Jul 2001.
- [WCD<sup>+</sup>98] O. Wolfson, S. Chamberlain, S. Dao, L. Jiang, and G. Mendez. Cost and imprecision in modeling the position of moving objects. In *Proceedings of the Fourteenth International Conference on Data Engineering*, pages 588–596, Orlando, Florida, USA, 23–27 Feb 1998. IEEE Computer Society.
- [WER<sup>+</sup>03] L. Wischhof, A. Ebner, H. Rohling, M. Lott, and R. Halfmann. Adaptive broadcast for travel and traffic information distribution based on inter-vehicle communication. In *Proc. of IEEE Intelligent Vehicles Symposium (IV'03)*, Columbus, OH, USA, 9–11 Jun 2003.
- [WER04] L. Wischhof, A. Ebner, and H. Rohling. Self-organizing traffic information system based on car-to-car communication : Prototype implementation. In *International Workshop on Intelligent Transportation (WIT)*, Hamburg, Germany, 23–24 Mar 2004.
- [WJS<sup>+</sup>99] O. Wolfson, L. Jiang, A.P. Sistla, S. Chamberlain, N. Rische, and M. Deng. Databases for tracking mobile units in real time. In *Proceedings of the Seventh International*
-

- Conference on Database Theory (ICDT)*, pages 169–186, Jerusalem, Israel, 10–12 Jan 1999.
- [Wol02] O. Wolfson. Moving objects information management : The database challenge (vision paper). In *the proceedings of the 5th international workshop on Next generation information technologies and systems (NGITS 2002)*, pages 75–89, Caesarea, Israel, 24–25 Jun 2002. Springer.
- [WQQ08] Xiong Wei and Li Qing-Quan. Performance evaluation of data disseminations for vehicular ad hoc networks in highway scenarios. In *XXIth ISPRS Congress*, Beijing, China, 3–11 Jul 2008.
- [WSCY99] O. Wolfson, A.P. Sistla, S. Chamberlain, and Y. Yesha. Updating and querying databases that track mobile units. *Distributed and Parallel Databases*, Vol. 7(n°3) :257–387, 1999.
- [WSX<sup>+</sup>99] O. Wolfson, P. Sistla, B. Xu, J. Zhou, S. Chamberlain, Y. Yesha, and N. Rische. Tracking moving objects using database technology in DOMINO. In *Proceedings of the Next Generation Information Technologies and Systems, 4th International Workshop, NGITS'99*, page 112, Zikhron-Yaakov, Israel, 5–7 Jul 1999. Springer.
- [WX04] O. Wolfson and B. Xu. Data-on-the-road in intelligent transportation systems. In *IEEE International Conference on Networking, Sensing, and Control (ICNSC 2004)*, Taipei, Taiwan, 21–23 March 2004. Citeseer.
- [WXT07] O. Wolfson, B. Xu, and R.M. Tanner. Mobile peer-to-peer data dissemination with resource constraints. In *Proc. of the 8th International Conference on Mobile Data Management*, Mannheim, Germany, 7–11 May 2007. Citeseer.
- [WXY04] O. Wolfson, B. Xu, and H. Yin. Dissemination of spatial-temporal information in mobile networks with hotspots. In *Proceedings of the Second International Workshop On Databases Information Systems and Peer-to-Peer Computing*, Toronto, Canada, 29–30 Aug 2004.
- [WXYR05] O. Wolfson, B. Xu, H. Yin, and N. Rische. Resource discovery using spatio-temporal information in mobile ad-hoc networks. *Lecture notes in computer science*, Vol. 3833 :page 129, 2005.
- [XSJ03] Q. Xu, R. Sengupta, and D. Jiang. Design and analysis of highway safety communication protocol in 5.9 ghz dedicated short range communication spectrum. In *IEEE Vehicular Technology Conference (VTC Spring 2003)*, volume 4, pages 2451–2455, Jeju, Corea, 21–24 Apr 2003.
- [XW03] B. Xu and O. Wolfson. Time-series prediction with applications to traffic and moving objects databases. In *Proceedings of the 3rd ACM international workshop on Data engineering for wireless and mobile access*, pages 56–60, San Diego, California, 19 Sep 2003. ACM New York, NY, USA.
- [XWR06] B. Xu, O. Wolfson, and N. Rische. Benefit and pricing of spatio-temporal information in Mobile Peer-to-Peer networks. In *Hawaii International Conference on System Sciences*, volume 39, page 223, Koloa, Hawaii, 4–7 Jan 2006.

- [YKLH07] L. Yan, M. Kihl, Y. Liu, and L. He. The application of inter-vehicle communication system to its. In *Geographical Information Science Research Conference, GISRUK*, pages 134–139, NUI Maynooth, Ireland, 11th–13th Apr 2007.
- [ZCY03] S. Zhong, J. Chen, and Y.R. Yang. Sprite : A simple, cheat-proof, credit-based system for mobile ad-hoc networks. In *The 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM 2003)*, volume Vol. 3, pages 1987–1997, san Francisco, California, USA, 1–3 Apr 2003.
- [ZWXS08] T. Zhong, O. Wolfson, B. Xu, and P. Szczurek. Trafficinfo : An algorithm for vanet dissemination of real-time traffic information. In *the Proceedings of the 15th World Congress on Intelligent Transportation Systems*, New York City, NY, 16–20 Nov 2008.
- [ZXW08a] T. Zhong, B. Xu, and O. Wolfson. Disseminating real-time traffic information in vehicular ad-hoc networks. In *Proc. of the IEEE Intelligent Vehicles Symposium (IV'08)*, Eindhoven, Netherlands, 4–6 Jun 2008.
- [ZXW08b] Xinjuan Zhu, Bo Xu, and Ouri Wolfson. Spatial queries in disconnected mobile networks. In *the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (ACM GIS 2008)*, page 38, Irvine, CA, USA, 5–7 Nov 2008.



## Résumé

Cette thèse s'intéresse au partage d'informations dans les réseaux de communication inter-véhiculaire. Grâce aux systèmes de positionnement et aux capacités de communication sans fil, les véhicules peuvent aujourd'hui s'échanger des données relatives à différents types d'événements (freinages d'urgence, embouteillages, places de stationnement disponibles, etc.) afin de proposer de nouveaux systèmes d'assistance aux conducteurs. Nous nous sommes focalisés dans ce travail sur la communication véhicule-à-véhicule (V2V). L'absence de toute infrastructure de communication fixe et la forte mobilité des nœuds (i.e., les véhicules) dans le réseau posent des problèmes particulièrement intéressants en terme de gestion des données, par exemple en terme d'estimation de la pertinence dans ces environnements particulièrement dynamiques. Dans ce contexte, nos principales contributions concernent :

- La proposition de mécanismes, basés sur des calculs de probabilité de rencontre, permettant d'estimer la pertinence des informations échangées entre les véhicules.
- Des mécanismes d'évaluation de requêtes continues dans les environnements décentralisés permettant de maintenir l'information à jour.
- Un protocole de dissémination, exploitant notre probabilité de rencontre, permettant d'acheminer les données vers les véhicules potentiellement intéressés.

Nos propositions ont été évaluées dans le système VESPA (Vehicular Event Sharing with a mobile Peer-to-peer Architecture) ainsi qu'au travers de nombreuses simulations.

## Abstract

This thesis is interested in the information sharing in inter-vehicle communication networks. Today, thanks to the positioning systems and to the wireless communication capacities, vehicles can exchange data relating different kind of events (emergency brake, traffic jam, available parking spot, etc.) in order to provide new assistance systems for the drivers. In this work, we focused on vehicle-to-vehicle communication (V2V). The absence of any fixed communication infrastructure and the high nodes' mobility (i.e. vehicles) in the network raise particularly interesting problems in term of data management, for example, in term of relevance estimation in these particularly dynamic environments. In this context, our main contributions concern :

- The proposal of mechanisms, based on the computation of an encounter probability, allowing to estimate the relevance of information exchanged between vehicles.
- Mechanisms of continuous query processing in decentralized environments allowing to keep information updated.
- A dissemination protocol, exploiting our encounter probability, allowing to route data to vehicles potentially interested.

Our proposals were evaluated in the VESPA (Vehicular Event Sharing with a mobile Peer-to-peer Architecture) system as

