

الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
وزارة التعليم العالي والبحث العلمي
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE
جامعة عمارة تليجي بالأغواط
UNIVERSITÉ AMAR TÉLIDJI LAGHOUAT



كلية العلوم
FACULTE DES SCIENCES
DEPARTEMENT D'INFORMATIQUE

THESE DE DOCTORAT

Discipline : Informatique

Présentée par :

Mr BRIK Bouziane

LA COLLECTE ET L'AGREGATION DE DONNEES DANS LES RESEAUX MOBILES

Soutenue le 03/07/2017 devant le jury composé de :

Mr A. TAHARI	M.C.A	Université de Laghouat	Président
Mr M. B. YAGOUBI	Professeur	Université de Laghouat	Examineur
Mr S. BITAM	M.C.A	Université de Biskra	Examineur
Mr A. KORICHI	M.C.A	Université de Ouargla	Examineur
Mr N. LAGRAA	Professeur	Université de Laghouat	Rapporteur

LAGHOUAT, Juillet 2017

Remerciements

JE tiens à remercier en priorité ALLAH LE TOUT PUISSANT de m'avoir donné le courage, la force et la volonté d'achever ce travail.

Je remercie chaleureusement mes parents qui m'ont beaucoup aidé matériellement et moralement, depuis mon enfance jusqu'à ce que j'arrive à ce point-là, sans oublier aussi ma femme, mes sœurs, mes frères et tous les membres de ma famille.

Je tiens à adresser mes remerciements et ma profonde gratitude à mon directeur de thèse Professeur Lagraa Nasreddine pour m'avoir permis d'intégrer son équipe au pour son encadrement, son soutien, son aide précieuse et sa confiance sans lesquels ce travail n'aurait pas été ce qu'il est.

J'exprime tous mes remerciements aux membres de mon jury d'avoir acceptés de rapporter cette thèse et d'y avoir apportés des remarques et questions, lesquelles ont contribué à son amélioration :

- Mr A. TAHARI, maître de conférences à l'université de Laghouat.
- Mr M. B. YAGOUBI, professeur à l'université de Laghouat,
- Mr S. BITAM, maître de conférences à l'université de Biskra,
- Mr A. KORICHI, maître de conférences à l'université de Ouargla,

Je remercie également tous mes enseignants qui ont contribué à ma formation et un remerciement particulier à ma tante Mme Hadda BOUZOUAD qui était toujours disponible pour m'apporter son aide. Ses conseils m'ont été très utiles scientifiquement et humainement.

Merci à tous ceux qui ont participé de près ou de loin à l'aboutissement de ce travail, plus particulièrement, aux membres des deux laboratoires LIM et L3i de Laghouat et la Rochelle, respectivement, sans oublier aussi mes amis intimes Brahim, Mostafa, Smail, Hakim, Jaloul, Oussama, Mohamed, Omar, Ahmed, etc.

Brik Bouziane

*Je dédie ce travail à mes **parents** qui ont toujours cru en moi et m'ont soutenu,*

Brik Bouziane

Table des matières

Table des figures	7
Liste des tableaux	10
1 Introduction générale	16
1.1 Vue générale sur les réseaux véhiculaires	17
1.1.1 Environnement de déploiement des réseaux véhiculaires	18
1.1.2 Caractéristiques des réseaux véhiculaires	18
1.1.3 Applications des réseaux véhiculaires	19
1.1.4 Activités de normalisation et de standardisation des réseaux véhiculaires	20
1.2 Problématiques	22
1.3 Objectifs et contributions	23
1.4 Organisation de la thèse	25
2 Revue de littérature sur la collecte de données dans les réseaux véhiculaires	27
2.1 La collecte de données dans les réseaux Ad hoc	27
2.1.1 La collecte de données dans les réseaux de capteurs	28
2.1.2 La collecte de données dans les réseaux véhiculaires (VANet)	28
2.2 Les défis de conception d'un protocole de collecte dans les VANet	29
2.3 Les étapes de la collecte de données dans les VANet	30
2.4 Les travaux connexes à la collecte des données dans les VANet	34
2.4.1 Les protocoles de collecte de données dans les VANet	35
2.4.2 Classification des protocoles de collecte de données	40
2.5 Les métriques d'évaluation des protocoles de collecte de données dans les VANet	42
2.6 Conclusion	45
3 Le protocole DDGP : Distributed Data Gathering Protocol for Vehicular Networks	46
3.1 Description de notre protocole DDGP	48

3.1.1	Modèle du réseau	49
3.1.2	Format de paquets	49
3.1.3	La phase d'initiation du processus de la collecte	50
3.1.4	La phase de la collecte de données	51
3.1.5	La phase d'agrégation de données collectées	57
3.1.6	La phase d'envoi des données collectées vers l'initiateur	58
3.1.7	Preuve de correction et analyse de la complexité de DDGP	58
3.1.8	Simulation et résultats	65
3.2	Conclusion	70
4	Revue de la littérature sur l'offre et la consommation de services dans les réseaux véhiculaires	71
4.1	Du Cloud Computing au Cloud véhiculaire	72
4.2	Les types du Cloud véhiculaire	73
4.3	Les services du Cloud véhiculaire mobile	73
4.4	Les architectures du Cloud véhiculaire mobile	74
4.5	Les défis de conception d'un protocole d'offre et de consommation de services dans le CAV	75
4.6	Les acteurs d'un protocole d'offre et de consommation de service dans CAV . .	76
4.7	Les étapes d'un protocole d'offre et de consommation de service dans CAV . .	76
4.8	Les travaux connexes au Cloud véhiculaire	77
4.8.1	Les architectures du Cloud véhiculaire	77
4.8.2	Les protocoles d'offre et de consommation de service	79
4.9	Discussion et objectifs de recherche	82
4.10	Conclusion	84
5	Nos protocoles d'offres et de consommation de services dans le Cloud véhiculaire	85
5.1	DCCS-VC : Discovering and Consuming Cloud Services in Vehicular Cloud . .	86
5.1.1	Vue générale sur le protocole DCCS-VC	86
5.1.2	Les phases du protocole DCCS-VC	87
5.1.3	Sélectionner le meilleur répertoire de Cloud dans DCCS-VC	90
5.1.4	Simulation et résultats	92
5.2	RCS-VC : Renting out and Consuming Services in Vehicular Clouds	98
5.2.1	Long Term Evolution-Advanced (LTE-A)	98
5.2.2	Vue générale sur le protocole RCS-VC	99
5.2.3	Les phases du protocole RCS-VC	100

5.2.4	Simulation et résultats	102
5.3	Conclusion	108
6	Nos modèles de sélection de services dans le Cloud véhiculaire	109
6.1	La sélection de service dans DCCS-VC fondée sur la logique floue	110
6.1.1	Introduction à la logique floue	110
6.1.2	Description du modèle flou de sélection de service dans DCCS-VC	114
6.2	La sélection de service fondée sur la théorie des jeux	120
6.2.1	Introduction à la théorie des jeux	120
6.2.2	Description du jeu dans GSS-VC	124
6.3	Conclusion	132
7	Conclusion générale et perspectives	133
	Liste des publications	136
	Bibliographie	138
	Glossaire	147

Table des figures

1.1	Les VANet et véhicules intelligents [1].	17
1.2	La bande de fréquence DSRC.	20
1.3	Le modèle WAVE 1609.	21
2.1	Types de données potentielles à la collecte de données dans les VANet.	29
2.2	Les étapes de la collecte de données.	31
2.3	Les phases du TrafficGather : (A) La répartition des véhicules en plusieurs groupes et élection de CHs. (B) La collecte de données par les CHs. (C) La diffusion des données collectées vers le nœud initiateur (CH_0).	35
2.4	Les phases du CGP : (A) partitionnement de la route et l'élection des Cluster-Heads. (B) La collecte de données et l'envoi des données collectées vers le RSU via une communication V2V puis V2I.	36
2.5	Les phases du Road Probing : (A) La sélection des véhicules Probes par un RSU. (B) La collecte et l'envoi des données collectées vers le RSU via une communication multi-saut (V2V).	37
2.6	Les phases du CDGP : (A) Initiation du processus de la collecte par un RSU. (B) La segmentation de la zone de la collecte et l'élection des CHs. (C) La collecte de données par les CHs sur les CSs.	38
3.1	Un exemple illustrative du fonctionnement des deux protocoles ECDGP et DDGP. (A) Initiation du processus de la collecte. (B) La propagation du paquet Col. (C) La segmentation de la zone de la collecte. (D) La sélection du RVS (ou CHs). (E) La collecte de données par un RVS (CH). (F) L'envoi des données collectées vers l'initiateur.	48
3.2	Les formats de paquets de DDGP.	50
3.3	Notre algorithme d'élection.	53
3.4	La segmentation des segments de la collecte (CS).	54

3.5	La topologie en anneau des véhicules RVCs.	55
3.6	Le principe de la technique VL-CSMA.	56
3.7	Notre mécanisme d'agrégation.	58
3.8	La retransmission du paquet Coll.	60
3.9	Evaluation des performances de DDGP dans un environnement autoroutier. (a) le temps d'attente moyen pour les applications DTD. (b) le temps d'attente moyen pour les applications RTD. (c) Le taux de retransmission des données pour les applications DTD. (d) Le taux d'agrégation de données.	66
3.10	Evaluation des performances de DDGP dans un environnement autoroutier. (a) Le délai de la phase de la collecte pour les applications DTD. (b) Le délai de la phase de la collecte pour les applications RTD. (c) Le délai de la phase de la collecte pour les applications RTD entre DDGP et DDGP sans agrégation. (d) Le délai de la phase de la collecte pour les applications DTD entre DDGP, DDGP sans agrégation, DDGP sans retransmission.	68
4.1	Le Cloud Ad hoc Véhiculaire.	74
4.2	Les véhicules utilisant le Cloud Computing.	75
4.3	Les étapes d'un protocole d'offre et de consommation de services.	77
4.4	L'architecture VANET-CLOUD.	79
4.5	Le déroulement du protocole CROWN.	80
4.6	Les niveaux du protocole proposé dans [2].	81
5.1	Vue générale sur DCCS-VC.	86
5.2	Les étapes de fonctionnement du protocole DCCS-VC.	87
5.3	Evaluation des performances du protocole DCCS-VC dans l'environnement urbain en fonction de la densité des véhicules fournisseurs. (a) Le Délai de Découverte de services (DD). (b) Le Délai de Consommation de services (CD). (c) Les acquittements positifs. (d) Les acquittements négatifs. (e) Le trafic des véhicules (VT). (f) Le délai de bout en bout des paquets de données (E2ED).	95
5.4	Evaluation des performances du protocole DCCS-VC dans l'environnement urbain par rapport aux autres protocoles. (a) Le Délai de Découverte de services (DD). (b) Le Délai de Consommation de services (CD). (c) Le Trafic des Véhicules (VT). (d) Le délai de bout en bout des paquets de données (E2ED).	97
5.5	L'architecture de LTE-A.	99
5.6	Une vue générale sur RCS-VC.	100
5.7	Les étapes de fonctionnement du protocole RCS-VC.	101

5.8	La localisation des trois eNodeBs dans la ville de Laghouat.	103
5.9	Evaluation de performance du protocole RCS-VC. (a) Le Délai de Découverte de services (DD). (b) Le Délai de Consommation de services (CD). (c) La durée d'attente dans la file d'attente (QD). (d) Le nombre des acquittements positifs. (e) Le nombre des acquittements négatifs.	106
5.10	Evaluation de performance du protocole RCS-VC par rapport aux autres protocoles. (a) Le Délai de Découverte de services (DD). (b) Le Délai de Consommation de services (CD).	107
6.1	Fonction d'appartenance caractérisant le sous-ensemble 'bon' de la qualité du service.	111
6.2	Variable linguistique qualité de service.	112
6.3	Les étapes d'un système flou.	112
6.4	Les fonctions d'appartenances des variables linguistiques : (a) le coût d'exécution, (b) la durée d'exécution, (c) réputation et rapport d'exécution avec succès, (d) débit, (e) service	115
6.5	La défuzzification de l'ensemble flou Moyen.	118
6.6	Jeu sous forme extensive.	123
6.7	Évaluation des performances du GSS-VC en termes du gain de service et le rapport d'offre de service avec succès durant la période on-peak.	131

Liste des tableaux

2.1	Les paramètres potentiels d'entrée du processus de la collecte.	31
2.2	Comparison entre les protocoles d'accès ordonné.	41
2.3	Comparison entre les protocoles d'accès aléatoire.	43
3.1	Les bits du champ Option dans le paquet Coll.	51
3.2	Les paramètres de notre simulation.	65
4.1	Comparison entre les travaux proposés dans le Cloud véhiculaire.	83
5.1	Les attributs de services de Cloud véhiculaire.	88
5.2	Les préférences d'un véhicule consommateur pour chaque type de service. . . .	89
5.3	Les critères de qualité des bus public.	90
5.4	Les paramètres de simulation.	93
5.5	Le pourcentage d'enregistrement des fournisseurs avec succès.	98
5.6	Les paramètres de simulation.	104
6.1	Ensemble des règles.	116
6.2	Ensemble de règles obtenues.	117
6.3	Scénarios de simulation.	119
6.4	Résultats de simulation	120
6.5	Exemple d'une matrice de gain.	123
6.6	Matrice des gains du VF et VC.	125
6.7	Les paramètres de simulation.	128
6.8	Les cinq meilleurs véhicules fournisseurs dans chaque période.	130

ملخص

شهدت صناعة السيارات نموا قويا بسبب التكنولوجيا اللاسلكية وقدرات الكشف والإدراك للبيئة الى جانب قدرات المعالجة و التخزين للمعلومات. وقد ادت هذه التطورات الى ظهور شبكات المركبات اللاسلكية التي تحفز على تحقيق مجموعة واسعة من التطبيقات لتحسين سلامة وكفاءة حركة المرور وراحة مستخدمي الطريق. في الوقت الحاضر، تعتبر المركبة كمنصة متحركة قادرة على التواصل مع المركبات أو غيرها من البنى التحتية على الطريق، وعلى إدراك بيئتها وجمع المعلومات وتوفير الموارد كخدمة إلى غيرها من المركبات.

المعلومات التي تجمعها أجهزة الاستشعار المختلفة لنفس السيارة توفر في الغالب معرفة ناقصة وأقل دقة. ولكن، من خلال مقارنة المعلومات مع مركبات متعددة، يصبح من الممكن استكمال تصور المركبات في بيئتهم وتصبح المعلومات أكثر دقة وذات الصلة.

من ناحية أخرى، مجال توفير الخدمات يوفر للسائقين العديد من الخدمات مثل الإنترنت، المعلومات، التخزين والحوسبة، الخ ولكن، ونظرا لسرعة تنقل المركبات، يجب على المركبات المستهلكة ان تجد أولا المركبات الموفرة للخدمات قبل أن تتمكن من استخدام هذه الخدمات.

في هذه الأطروحة، نركز على جمع البيانات وتقديم الخدمات في شبكات المركبات. في أول مساهمة، اقترحنا بروتوكولين لجمع البيانات بطريقة فعالة و موثوقة بها وفي أقل مدة زمنية ممكنة. و بينت نتائج المحاكاة أن البروتوكولات التي اقترحناها هي أكثر كفاءة من حيث الموثوقية وجمع البيانات، مقارنة بالبروتوكولات الأخرى.

في مساهمة ثانية، اقترحنا بروتوكولين لعرض واستهلاك الخدمات المقترحة من طرف المركبات والذان يرتكزان اساسا على الحافلات العامة والبنية التحتية للشبكة الخلوية للجيل الرابع وهذا لتسهيل عمليتي اكتشاف و استهلاك الخدمات المقدمة. وقد أثبتت نتائج المحاكاة فعالية البروتوكولين في التقليل من المدة الزمنية لاكتشاف واستهلاك الخدمات.

في مساهمة أخيرة، اقترحنا نموذجين لاختيار افضل الموردين للخدمات، اللذين ياخذان بعين الاعتبار أدواق المستهلكين ومتطلبات الموردين لتحديد أفضل مورد من حيث جودة الخدمة.

الكلمات الأساسية: شبكة المركبات، جمع البيانات، توفير الخدمات، اختيار افضل الموردين للخدمات.

Résumé

Le développement de l'industrie automobile connaît une forte croissance en raison des technologies sans fil, des aptitudes de détection et de perception de l'environnement, et des capacités de stockage et de traitement présents dans les véhicules. Ces évolutions ont permis l'apparition des réseaux Ad hoc de véhicules VANets qui stimulent la réalisation d'une large variété d'applications afin d'améliorer la sécurité, l'efficacité du trafic routier et le confort des usagers de la route. De nos jours, un véhicule est devenu une sorte d'une plateforme mobile capable de communiquer avec les autres véhicules ou l'infrastructure sur la route, de percevoir son environnement et de collecter des informations, et d'offrir ses ressources en tant que service à d'autres véhicules.

Acquises individuellement, les informations collectées par les différents capteurs d'un même véhicule fournissent le plus souvent une connaissance incomplète et moins précise. Par contre, en confrontant les informations issues de plusieurs véhicules, il devient possible de compléter la perception qu'ont les véhicules de leur environnement et l'information devient plus précise et plus pertinente.

D'un autre côté, l'offre de services ou bien le Cloud véhiculaire fournit aux conducteurs l'accès à plusieurs services tels que, Internet en tant que service, informations en tant que service, stockage et calcul en tant que services, etc. Néanmoins, vue la forte mobilité des véhicules, les véhicules consommateurs doivent tout d'abord découvrir les véhicules fournisseurs et leurs services avant de pouvoir consommer les services requis.

Dans cette thèse, nous nous intéressons à la collecte de données et l'offre de services dans les réseaux véhiculaires. Dans une première contribution, nous avons proposé deux protocoles de collecte de données baptisés ECDGP (Extended Cluster-based Data Gathering Protocol) et DDGP (Distributed Data Gathering Protocol). Ils fournissent un processus de collecte efficace et fiable, en collectant le maximum de données en le minimum de temps et en minimisant le taux de données erronées. Les résultats de simulation ont montré que nos protocoles sont plus performants en termes de fiabilité et d'efficacité de la collecte de données, par rapport aux protocoles de collecte existants.

Dans une seconde contribution, nous avons proposé deux protocoles d'offre et de consommation de services, nommés DCCS-VC (Discovering and Consuming Cloud Services in Vehicular Clouds) et RCS-VC (Renting out and Consuming Services in Vehicular Clouds). DCCS-VC et

RCS-VC exploitent, respectivement, les bus publics et l'infrastructure du réseau cellulaire LTE-A comme étant des répertoires de Cloud, dans lesquels les véhicules fournisseurs enregistrent leurs offres et à partir desquels les véhicules consommateurs découvrent tous les services offerts. Les résultats de simulation ont prouvé leur efficacité à minimiser les délais de découverte et de consommation de services.

Dans une dernière contribution, nous avons proposé deux modèles de sélection des fournisseurs les plus adéquats dans le Cloud véhiculaire. Nos modèles prennent en considération les préférences des consommateurs et les exigences des fournisseurs afin de déterminer le meilleur fournisseur en termes de qualité de service.

Mots clés : Réseaux Ad hoc véhiculaire (VANet), Collecte de données, Cloud véhiculaire, Sélection de services (fournisseurs), ECDGP, DDGP, DCCS-VC, RCS-VC.

Abstract

The automotive area is witnessing a tremendous evolution due to the increasing growth of in-vehicles wireless technologies, environmental sensing & perception aptitudes, and storage & processing capacities. These advances have seen the emergence of new communication systems including Vehicular Ad hoc Network (VANet) which stimulates the development of a wide range of applications to enhance driving safety and efficiency, and make traveling more comfortable. Nowadays, a vehicle has become a kind of a mobile platform that can communicate with other vehicles or infrastructure on the road, to perceive its environment and to collect information as well as to provide resources as a service to other vehicles.

Information collected by the various sensors of the same vehicle give a less precise and incomplete knowledge about its environment. However, exploiting the collected information by multiple vehicles makes the knowledge built up more accurate and relevant.

Besides, service offering or vehicular Cloud enables drivers to access to several services, such as Internet as a service, information as a service, storage and computing as services, and so on. Nevertheless, due to the high mobility of vehicles, consumer vehicles need first to discover provider vehicles and their services before being able to consume required services.

In this thesis, we targeted both data collection and service offering in vehicular networks. First, we proposed two data collection protocols called ECDGP (Extended Cluster-based Data Gathering Protocol) and DDGP (Distributed Data Gathering Protocol). They provided an efficient and reliable collection process by collecting the maximum amount of data in the minimum amount of time and minimizing the rate of erroneous data. Simulation results have shown that our protocols outperformed existing data collection protocols in terms of reliability and efficiency.

Second, we proposed also two service offering and consuming protocols, named DCCS-VC (Discovering and Consuming Cloud Services in Vehicular Clouds) and RCS-VC (Renting out and Consuming Services in Vehicular Clouds). DCCS-VC and RCS-VC exploited public buses and LTE-A cellular network infrastructure as Cloud directories, respectively, in which provider vehicles stored their offers and from which consumer vehicles discovered all offered services. The simulation results showed the efficiency of our protocols to minimize both service discovering and consuming delays compared to the other protocols.

Finally, we proposed two service selection schemes in vehicular Cloud to select the most suitable consumer vehicle. Our schemes took into account consumers preferences and providers requirements and determined the best provider in terms of quality of service.

Keywords: Vehicular Ad hoc Network (VANet), Data Collection, Vehicular Cloud, service selection, ECDGP, DDGP, DCCS-VC, RCS-VC.

Chapitre 1

Introduction générale

DE nos jours, l'augmentation du nombre des véhicules en circulation a accru les problèmes de la congestion du trafic routier et de la sécurité des usagers de la route. En effet, le problème de la congestion du trafic dans les grandes villes se traduit par des embouteillages et des temps de parcours qui s'allongent. Par exemple, il y a en moyenne plus de deux heures d'embouteillage par jour à Moscou [3]. Le deuxième problème encore important est celui de la sécurité des usagers de la route. D'après l'OMS (Organisation Mondiale de la Santé), les accidents de la route entraînent chaque année 1,25 million de décès et 50 millions de blessés dans le monde. Ces chiffres peuvent s'aggraver de 65% de plus dans les vingt prochaines années, si la sécurité routière ne s'améliorera pas [4].

Une des solutions qui a été établie pour résoudre le problème de la congestion et améliorer la sécurité routière est celle des STIs (ou ITSs pour Intelligent Transportation Systems). Les STIs consistent en l'intégration des Technologies de l'Information et de la Communication (TIC) aux systèmes de transport afin de le rendre plus sûr, plus efficace et plus confortable [5]. Dans ce contexte, la communication sans-fil véhiculaire (V2X), ou ce qu'on appelle les réseaux VANets pour Vehicular Ad hoc Networks [6, 1], émergent comme une technologie sans fil prometteuse sur lesquels les STIs se basent afin d'atteindre leurs objectifs. Cette technologie est capable d'améliorer la vision des conducteurs en ouvrant la voie à une variété d'applications et services innovants. Dans cette thèse, nous nous adressons deux problématiques différentes, à savoir la collecte de données et l'offre de services (ressources) dans les réseaux véhiculaires.

Dans ce premier chapitre, nous allons donner une vue générale sur le contexte, la problématique

ainsi que nos contributions dans le cadre de cette étude.

1.1 Vue générale sur les réseaux véhiculaires

Dans cette section, nous présentons une vue générale sur le contexte de cette thèse qui porte sur les réseaux véhiculaires. Nous y présentons une définition, les caractéristiques et les environnements de déploiement de ces réseaux, les applications qui peuvent être envisagées dans ces réseaux ainsi que quelques travaux de normalisation et de standardisation lancés afin de répondre aux spécificités des réseaux véhiculaires.

Un réseau Ad hoc de véhicules (VANet) est un réseau mobile constitué des véhicules qui sont capables de communiquer les uns avec les autres via une communication sans fil Véhicule à Véhicule (V2V) et avec aussi les infrastructures installées au bord de la route, RSU pour Road Side Units, via une communication sans-fil Véhicule à Infrastructure (V2I) (voir la FIGURE 1.1). Les objectifs principaux de ce type de réseau consistent à améliorer la sécurité routière et le confort des usagers de la route en rendant le trafic plus sûr, plus efficace et plus confortable.

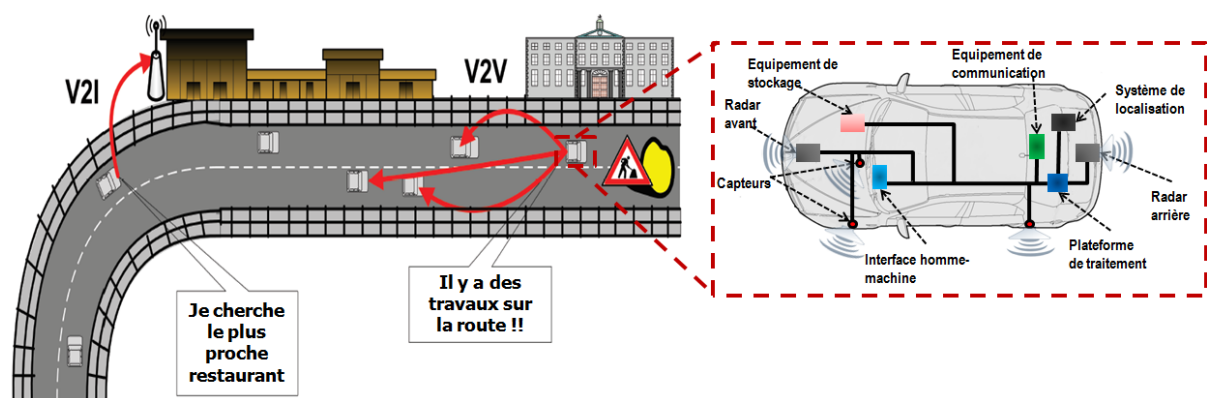


FIGURE 1.1: Les VANet et véhicules intelligents [1].

Actuellement, les nouveaux véhicules, dits "Intelligents", sont dotés de plusieurs dispositifs de détection (caméras, capteurs, radars, etc.), d'un système de localisation (GPS, Global Positioning System), de grandes capacités de stockage, des ressources importantes de traitement et d'une énergie illimitée (voir la FIGURE 1.1).

1.1.1 Environnement de déploiement des réseaux véhiculaires

Vu les spécificités du réseau (la vitesse et la densité des véhicules), il existe trois milieux de déploiement des réseaux véhiculaires [6] :

1. Milieu urbain : se caractérise par une forte densité de véhicules ainsi qu'une vitesse des véhicules réduite (une moyenne de 50km/h). Cependant, il s'agit d'un milieu qui est très perturbant pour l'établissement des liens radio lors d'une communication V2V, à cause de la multitude d'obstacles (immeubles, bâtiments, etc).
2. Milieu rural (suburbain) : il s'agit d'un milieu moins perturbant qui est caractérisé par une densité de véhicules faible et une vitesse moins importante.
3. Milieu autoroutier : en raison de l'absence des obstacles, ce milieu est moins perturbant pour les ondes radio par rapport au milieu urbain, mais il se caractérise par une vitesse des véhicules très élevée, et donc il nécessite une grande fiabilité de la part des conducteurs afin d'éviter les accidents de la route.

1.1.2 Caractéristiques des réseaux véhiculaires

Les réseaux véhiculaires se caractérisent par plusieurs propriétés qui doivent être prises en considération lors de la réalisation des protocoles dans ce type de réseau [6] :

- **La forte mobilité** : un des facteurs qui distingue les réseaux véhiculaires des autres réseaux sans fil est la vitesse des véhicules qui varie selon l'environnement. Par conséquent, un changement très dynamique de la topologie du réseau véhiculaire.
- **Les ressources de véhicules** : en plus des interfaces radio sans fil, les véhicules sont dotés de grandes capacités de stockage, de puissance de traitement ainsi que de sources d'énergie illimitées. Ce qui présente un des avantages des réseaux véhiculaires par rapport aux autres réseaux tels que MANet (Mobile Ad-hoc Network) et WSN (Wireless Sensor Network), qui sont limités en termes de ressources.
- **Le modèle de mobilité** : à la différence des autres réseaux sans fils où la mobilité des nœuds est généralement aléatoire tels que les réseaux MANets, les réseaux véhiculaires se caractérisent par une mobilité prédictible à cause de l'infrastructure routière (route, autoroute, panneaux de signalisation, limites de vitesse, ...). De plus, la mobilité des véhicules peut être aussi affectée par le comportement des conducteurs et leurs réactions face à des situations différentes, telles que les accidents, les embouteillages, ...
- **Le canal radio** : dans les réseaux sans fil traditionnels, les communications s'effectuent généralement dans des espaces ouverts sans obstacle. Les communications dans les réseaux véhiculaires se font en environnement externe contenant une multitude d'obstacles

notamment en zones urbaines. Ces obstacles causent une sévère dégradation de la qualité et de la puissance des signaux de communication.

- **Diversité de la densité** : la densité des véhicules n'est pas uniforme mais sa variation est spatio-temporelle. Par exemple, la densité en milieu urbain est beaucoup plus élevée qu'en milieu rural. D'un point de vue temporel, la densité est par exemple différente selon qu'on considère la nuit ou la journée, les heures de pointe ou les heures creuses.

1.1.3 Applications des réseaux véhiculaires

La capacité de communication de véhicules et les ressources embarquées ont permis d'envisager plusieurs applications pour les réseaux véhiculaires. Ces applications peuvent être classifiées en trois grandes catégories [6] :

- A. Les applications de sécurité : c'est le type d'applications le plus important qui vise à améliorer la sécurité des usagers de la route. Il se base sur la diffusion périodique des messages d'alerte sur l'état du trafic, des routes, de la météo, les accidents et les travaux sur la route, les limites de vitesse, la distance de sécurité, etc.
- B. Les applications de gestion du trafic : elles ont pour objectif de réduire les embouteillages et les risques d'accidents sur la route et donc de rendre le trafic plus efficace. Elles consistent aussi à collecter et à diffuser des informations aux conducteurs afin de leur faciliter la conduite. Parmi ces applications nous pouvons citer, le contrôle du trafic, la gestion des feux de signalisation, l'aide aux dépassements, etc.
- C. Les applications de confort ou de divertissement : elles offrent aux usagers de la route un ensemble de services dont l'objectif est de rendre les voyages plus agréables tels que l'accès mobile à Internet, le chat inter-véhicules, les jeux en réseaux, la gestion des parkings, etc.

En fonction de ces trois types d'applications, trois types de messages peuvent être échangés dans les réseaux véhiculaires :

- I. Messages périodiques : se sont les messages de contrôle (Beacon), diffusés à intervalle régulier et comprenant les informations générales de l'émetteur telles que l'identifiant, la vitesse, la position, la direction, l'accélération, etc. Ce type de messages permet à chaque véhicule d'avoir une vue locale de son voisinage.
- II. Messages basés sur un événement : se sont les messages d'alertes, générés et diffusés à intervalle régulier après la détection d'un événement (accidents, obstacles...). Ce type de messages doit être diffusé le plus rapidement possible et comprend essentiellement les coordonnées du lieu de l'accident (obstacle).

III. Autres messages : se sont tous les autres types de messages qui ne sont pas diffusés à intervalle régulier et peuvent nécessiter un routage pour atteindre la destination.

1.1.4 Activités de normalisation et de standardisation des réseaux véhiculaires

Plusieurs organismes nationaux, régionaux et internationaux ont lancé des activités de standardisation et de normalisation portant sur les réseaux véhiculaires. Nous pouvons citer :

– IEEE (Institute of Electrical and Electronics Engineers)

Les réseaux VANets sont basés sur des transmissions sans fil à courte portée. En 1999, Federal Communications Commission (FCC) aux Etats-Unis a dédié une bande de fréquence de 75 MHz, entre 5.850 et 5.925 GHz, spécifique aux communications intervéhicules, c'est le DSRC (Dedicated Short Range Communication) [7]. Comme le montre la FIGURE 1.2, la bande DSRC est divisée en sept canaux de communications de 10 MHz chacune et une bande de garde de 5 MHz. Ainsi, DSRC définit un canal de contrôle (CCH pour Control CHannel) pour l'envoi des paquets de contrôles et d'alertes et six canaux de service (SCH pour Service CHannel) pour les autres types de messages (de données).

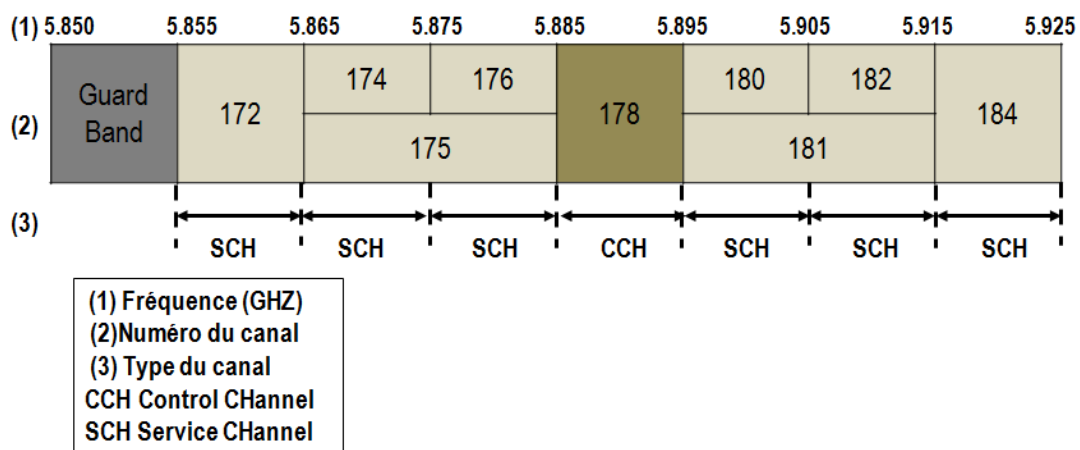


FIGURE 1.2: La bande de fréquence DSRC.

Par la suite, la bande DSRC a été reprise par l'organisme IEEE, en 2003, afin de définir un standard dédié aux communications inter-véhicules en ajoutant la norme 802.11p [8]. Cette norme ne définit que les spécifications de la couche physique et la sous couche MAC. Elle permet d'offrir des débits allant de 3 à 27Mbit/s et des portées de communications de 300m à 1000m.

Dans le but de définir les spécifications des couches supérieures, IEEE a lancé la famille des protocoles 1609 [9], appelée aussi WAVE (Wireless Ability in Vehicular Environments) (voir la FIGURE 1.3). Ce standard est structuré en quatre composantes : (1) **IEEE 1609.1 - WAVE Resource Manager**, pour la gestion des ressources. (2) **IEEE 1609.2 - WAVE Security Services for Applications and Management Messages**, pour la sécurisation des messages. (3) **IEEE 1609.3 - WAVE Networking Services**, pour les services de niveau réseau et transport incluant l'adressage et le routage. (4) **IEEE 1609.4 - WAVE Multi-Channel Operation**, pour la coordination et la gestion des canaux de DSRC.

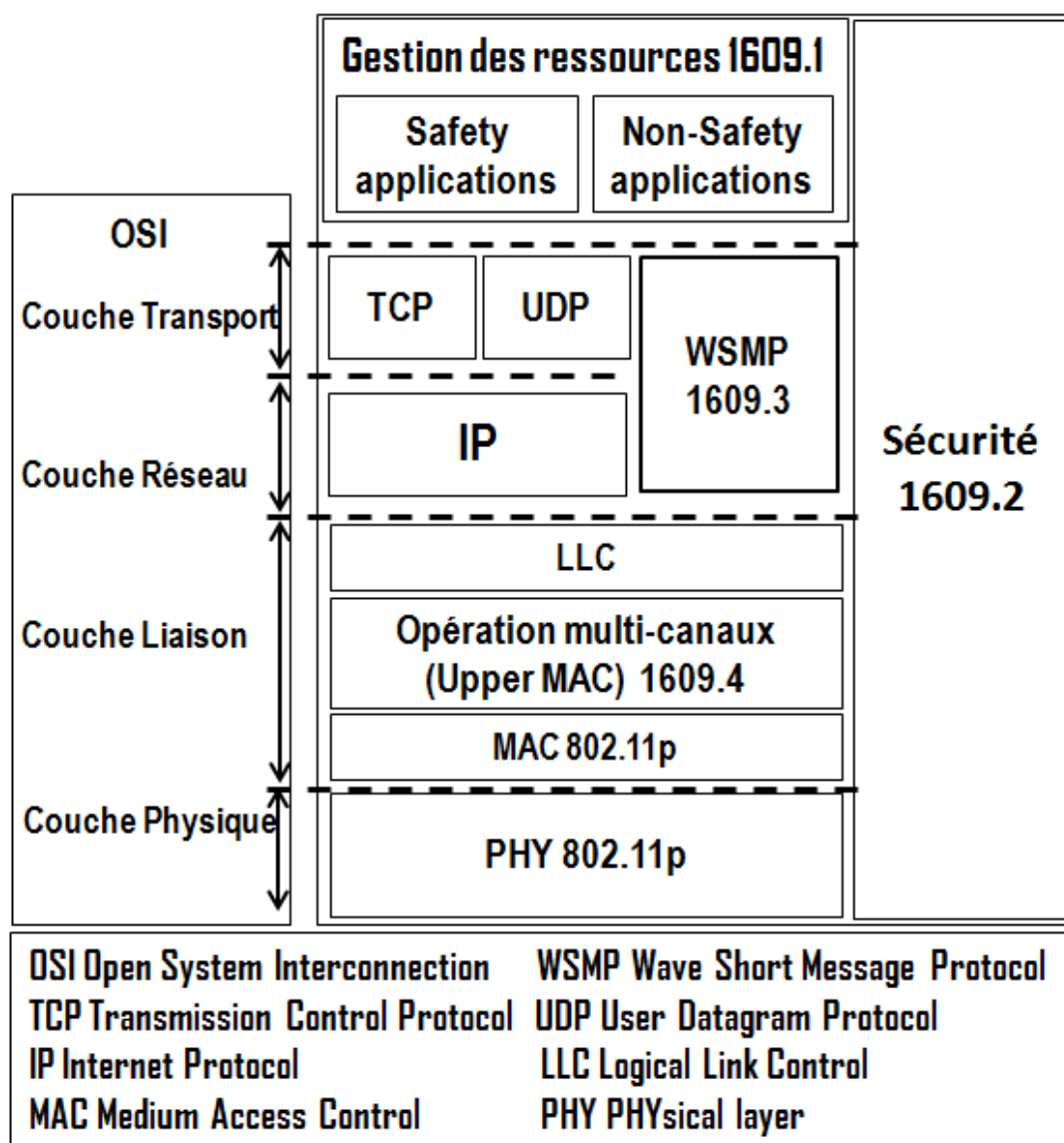


FIGURE 1.3: Le modèle WAVE 1609.

- ISO (International Organization for Standardization)
ISO a doté les véhicules d'une plateforme logicielle dite CALM (Continuous Air-interface, Long and Medium range) [10], dont l'objectif est d'assurer une communication sans fil entre plusieurs technologies de communication hétérogènes (2G, 3G, DSRC, etc), ainsi que le handover entre ces technologies.
- ETSI (European Telecommunications Standards Institute)
Au niveau Européen, l'institut ETSI a dédié cinq groupes de travail afin de développer des standards et des spécifications pour les réseaux véhiculaires, allant de l'architecture de communication à la spécification des protocoles [11] : WG1- User Application requirements, WG2 -Architecture and cross layer issues, WG3 Transport and Network, WG4 - Media and related issues et le WG5 - Security. Dans le WG3 par exemple, ils s'intéressent à la spécification des protocoles d'adressage et de routage géographiques.

1.2 Problématiques

Comme mentionné précédemment, les STIs reposent sur des systèmes de collecte, de traitement et d'offres de plusieurs services afin de réaliser diverses applications qui vont des applications basiques telles que la gestion des feux de circulation et les panneaux à messages dynamiques, à des applications plus compliquées comme le contrôle de la congestion et la gestion des urgences. Dans ce contexte, les communications sans fil véhiculaires, y compris V2V et V2I, représentent une composante de base des STIs afin de soutenir l'intégration des TICs dans les STIs. Cependant, vu les différentes spécificités et la nature dynamique qui caractérisent les réseaux véhiculaires, ils ne sont qu'en phase expérimentale et plusieurs problématiques doivent être traitées avant de pouvoir déployer un tel réseau, telles que le contrôle d'accès au canal [12], le routage de données [13], la dissémination de données [14], la collecte et l'échange de données [15, 16], l'offre de services [17] ou encore la sécurité des données échangées [18].

En plus de la capacité de communication sans fil, les véhicules dans les STIs sont capables de collecter différentes informations provenant de leurs capteurs locaux. Ensuite, ces informations sont envoyées aux centres de gestion du trafic, dans le but de fournir un trafic routier plus sûr, plus efficace et plus confortable.

Cependant, prises individuellement, les informations collectées par les différents capteurs d'un même véhicule fournissent le plus souvent une connaissance incomplète. Par contre, en confrontant les informations issues de plusieurs véhicules, il devient possible de compléter la percep-

tion qu'ont les véhicules de leur environnement. En d'autres termes, en échangeant et en comparant les informations collectées par plusieurs véhicules, l'information devient plus précise et plus pertinente.

D'un autre côté, les capacités de véhicules en termes de ressources ont conduit aussi à la naissance d'un nouveau concept nommé Cloud véhiculaire [17]. Il permet aux véhicules d'agir comme étant des serveurs du Cloud mobile et d'offrir leurs ressources en tant que services à d'autres véhicules, qui ont besoin de plus de ressources. Ce nouveau concept fournit aux conducteurs l'accès à une large variété de services tels que, l'Internet en tant que service, des informations en tant que service, le stockage et le calcul en tant que services, etc. Néanmoins, vue la forte mobilité des véhicules, les véhicules consommateurs doivent tout d'abord découvrir les véhicules fournisseurs et leurs services avant de pouvoir consommer les services requis.

En effet, pour collecter des données et offrir des services, il s'avère nécessaire de développer des protocoles fiables et efficaces, qui prennent en compte les contraintes liées à chacune de ces deux applications ainsi que les différentes propriétés qui caractérisent les réseaux véhiculaires telles que la forte mobilité, le changement dynamique de la topologie, la déconnexion fréquente du réseau, la non fiabilité des communications, le délai, etc.

1.3 Objectifs et contributions

Le but de ce travail est double. Premièrement, nous nous sommes intéressés à la collecte de données au sein des réseaux véhiculaires. Notre objectif est de proposer des protocoles de collecte en définissant la façon dont les données vont être collectées par les nœuds du réseau. Dans ce contexte, nous avons pu apporter les contributions suivantes :

- Recenser et étudier les différents protocoles de collecte de données au sein des réseaux véhiculaires avant d'établir une étude comparative selon plusieurs critères de classification.
- Proposer le protocole ECDGP pour Extended Cluster-based Data Gathering Protocol [19]. ECDGP représente une version améliorée et étendue de l'un de nos protocoles de collecte que nous avons proposé auparavant (CDGP) [20]. Par rapport à CDGP, ECDGP supporte : (i) la collecte de données tolérante et non tolérante au délai. (ii) une nouvelle technique d'agrégation durant la phase de collecte, afin d'agréger les données collectées avant de les envoyer vers l'initiateur. (iii) un processus de collecte qui est complet en implémentant toutes les phases nécessaires. Nous notons que nos deux protocoles se basent

sur une architecture en clusters et utilisent une version centralisée et dynamique de la technique Space Division Multiple Access (SDMA) [21] proposée elle aussi dans [20]. De plus, ils assurent aussi une grande fiabilité en implémentant une technique de retransmission des données erronées.

- Proposer le protocole DDGP pour Distributed Data Gathering Protocol [22] dont les caractéristiques principales sont : (a) de permettre aux véhicules d'accéder au canal de façon distribuée en se basant sur leurs positions géographiques, grâce à notre nouvelle technique d'accès au canal qu'on a nommé VL-CSMA pour Vehicular Location-based Carrier Sense Multiple Access. (b) De supporter la collecte de données en temps réel et tolérante au délai. (c) D'implémenter une nouvelle technique d'agrégation afin de ne pas collecter les données non désirées et redondantes. (d) D'être adapté aux deux environnements autoroutier et urbain.

Le deuxième apport de cette thèse se porte sur l'offre et la consommation de services dans les réseaux véhiculaires (appelé aussi le Cloud véhiculaire). Nous avons fixé l'objectif de concevoir des protocoles d'offre et de consommation de services permettant aux véhicules consommateurs de découvrir les véhicules fournisseurs, connaître leurs services et de leur envoyer des requêtes sur les services requis afin de les consommer. Dans ce contexte, nos contributions peuvent être résumées comme suit :

- Explorer et étudier les travaux de recherche proposés dans le contexte du Cloud véhiculaire. Nous avons réparti ces travaux en deux grandes classes : (i) Architectures et protocoles d'offres et de consommation de services afin de permettre aux véhicules fournisseurs d'offrir leurs services et aux véhicules consommateurs de découvrir et de consommer les services des fournisseurs [23, 24, 25, 26, 27, 2, 28, 29]. (ii) L'approvisionnement de ressources pour gérer leur allocation de ressources et la sélection de services ou bien du nœud fournisseur le plus adéquat, pour chaque demande de consommation de service [30, 31].
- Proposer le protocole DCCS-VC pour Discovering and Consuming Cloud Services in Vehicular Clouds [32]. DCCS-VC se repose essentiellement sur l'utilisation des bus publics qui comme des répertoires du Cloud en formant un index dynamique pour les véhicules consommateurs et fournisseurs. Nous avons choisi d'utiliser les bus publics comme répertoires pour les deux raisons suivantes : (i) leurs prédictibilités en temps et en espace dans l'environnement urbain, (ii) ils couvrent la plupart des zones urbaines et donc ils améliorent la connectivité du réseau.
- Comme extension du DCCS-VC, nous avons proposé un modèle de sélection de services qui se base sur la logique floue pour permettre à chaque bus public de sélectionner le

véhicule fournisseur le plus adéquat, au lieu d'envoyer aux consommateurs la liste de tous les véhicules fournisseurs.

- Proposer le protocole RCS-VC pour Renting out and Consuming Services in Vehicular Clouds [33] qui est un protocole centralisé à l'aide du réseau de la quatrième génération LTE (Long Term Evolution) [34]. Le principe de RCS-VC repose sur l'exploitation de l'infrastructure de LTE-A comme étant un répertoire de Cloud, dans lequel les véhicules fournisseurs enregistrent leurs offres et à partir duquel les véhicules consommateurs découvrent tous les services offerts.
- Proposer un modèle générique de sélection de services dans le Cloud véhiculaire, nommé GSS-VC pour Game theory-based approach for Service Selection in Vehicular Cloud. Ce modèle se base sur la théorie des jeux pour permettre aux véhicules consommateurs de sélectionner les meilleurs fournisseurs sur la route. Il prend en considération les préférences des consommateurs et les exigences des fournisseurs afin de déterminer le meilleur fournisseur en termes de qualité de service.

1.4 Organisation de la thèse

Ce manuscrit de thèse est composé de cinq chapitres, organisés comme suit :

Dans le **chapitre 2**, nous passons en revue la collecte de données dans les réseaux véhiculaires. Nous commençons par une brève présentation sur la collecte de données dans les réseaux Ad hoc. Puis, nous exposons les défis de conception d'un protocole de collecte suivis des étapes de la collecte de données dans les VANet. Par la suite, nous exposons les recherches menées sur les protocoles de collecte de données dans les VANet et nous terminons ce chapitre par les métriques d'évaluation de ces protocoles.

Le **chapitre 3** est dédié à la présentation de notre protocole de collecte de données dans les VANet, nommé DDGP (Distributed Data Gathering Protocol) proposé dans le contexte de cette thèse. Nous commençons par la description de ses phases principales. Puis, nous analysons ses performances analytiquement et par simulations.

Nous présentons dans le **chapitre 4** le Cloud véhiculaire et ses caractéristiques. Nous commençons par donner un aperçu sur le passage du Cloud Computing au Cloud véhiculaire. Puis, nous présentons les concepts liés au Cloud véhiculaire ainsi que ses défis. Nous terminons ce chapitre par une discussion sur les travaux proposés et nos objectifs de recherche.

Le **chapitre 5** présente nos deux protocoles d'offre et de consommation de services, DCCS-VC (Discovering and Consuming Cloud Services in Vehicular Cloud) et RCS-VC (Renting out and Consuming Services in Vehicular Cloud). Pour chaque protocole, nous commençons tout d'abord par décrire son principe de fonctionnement. Puis, nous détaillons ses phases principales et nous terminons par l'analyse de ses performances par simulation.

Dans le **chapitre 6**, nous présentons nos deux modèles de sélection de services dans le Cloud véhiculaire. Le premier modèle est à base de la logique floue et le deuxième utilise la théorie des jeux. Tous les deux prennent en considération les préférences des consommateurs et les exigences des fournisseurs de services avant de déterminer le meilleur fournisseur en termes de qualité de service. Pour cela, nous commençons tout d'abord par donner une vue générale sur la logique floue et la théorie de jeux et leurs concepts de base. Ensuite, nous détaillons la modélisation théorique de chaque modèle.

Nous concluons cette thèse en rappelant les contributions et les résultats de celle-ci, ainsi que quelques perspectives soulevées par nos travaux.

Chapitre 2

Revue de littérature sur la collecte de données dans les réseaux véhiculaires

ACCOMPAGNANT l'évolution de l'industrie automobile, les véhicules sont de plus en plus équipés de radars, capteurs et caméras. Ces artefacts sont embarqués afin d'améliorer à la fois la sécurité des passagers et le confort de conduite. Ces véhicules sont capables de collecter différentes informations provenant de leurs capteurs locaux ou bien issues des autres véhicules sur la route. Ces informations peuvent être exploitées pour réaliser de nombreuses applications, telles que l'estimation du trafic routier, la vitesse moyenne sur un segment de route, la détection des segments dangereux, le nombre de places disponibles sur un parking, etc.

À travers ce chapitre, nous allons passer en revue les concepts et enjeux liés à la première problématique traitée dans cette thèse. Vu que l'un des objectifs de cette thèse est de proposer des protocoles de collecte de données dans les réseaux véhiculaires, ce chapitre est consacré à la collecte de données dans les réseaux véhiculaires. Nous commençons par une brève présentation de processus de la collecte de données dans les réseaux Ad Hoc. Puis, nous exposons les défis de conception d'un protocole de collecte dans les VANet. Ensuite, nous décrivons les étapes de la collecte de données dans les VANet. Par la suite, nous exposons les recherches menées sur les protocoles de collecte de données dans les réseaux véhiculaires et nous terminons ce chapitre par les métriques d'évaluation de ces protocoles.

2.1 La collecte de données dans les réseaux Ad hoc

La collecte de données consiste à collecter des données spécifiques et de les délivrer aux nœuds concernés, dans le but de réaliser une application donnée. Dans la littérature, la collecte

de données a été principalement traitée dans deux types de réseaux Ad hoc à savoir *les réseaux de capteurs* et *les réseaux véhiculaires*.

2.1.1 La collecte de données dans les réseaux de capteurs

Un réseau de capteurs (en anglais WSN pour Wireless Sensors Networks) est un réseau, généralement statique, composé d'un ensemble de capteurs sans fil qui se chargent de collecter les données depuis l'environnement physique sur lequel ils sont déployés. Les données collectées par les capteurs sont acheminées vers l'utilisateur final (l'observateur) via une communication multi-sauts [35, 36, 37]. La collecte de données dans ce type de réseaux permet de réaliser une large variété d'applications dans des domaines différents tels que les domaines médical, environnemental et militaire afin de contrôler le rythme cardiaque des patients, de détecter les feux de forêt et pour le contrôle de la qualité de l'eau ou de l'air, de collecter des informations sur la position de l'ennemi, surveillance des zones hostiles (contaminées)... [38, 39]. Cependant, dans un WSN, les capteurs sont des petits dispositifs électroniques à faible coût, ce qui implique forcément qu'ils ont des capacités réduites (capacité de calcul, de stockage, d'énergie, portée de communication), d'où la prise en considération de toutes ces spécificités et caractéristiques dans le développement des différents protocoles de collecte [40].

2.1.2 La collecte de données dans les réseaux véhiculaires (VANet)

La collecte dans les VANet a pour objectif de supporter plusieurs types d'applications liées à ce réseau. En effet, un véhicule peut être considéré comme étant un collecteur mobile qui recueille des données liées à des applications tolérantes et non tolérantes au délai. Ces informations peuvent être regroupées en trois grandes classes : (i) *Les informations liées à la sécurité routière* telles que les accidents sur la route et les conditions météorologiques afin d'avertir les conducteurs (voir la FIGURE 2.1 (C)-(D)). (ii) *Les informations liées à la gestion du trafic routier* afin d'améliorer les conditions de la circulation telles que la vitesse moyenne, la densité et la congestion sur la route (voir la FIGURE 2.1-(A) et (B)). (iii) *Les informations liées au confort des usagers de la route* afin d'offrir un ensemble de services de communication et d'information (voir la FIGURE 2.1 (E)-(L)), tels que le programme des matchs dans les stades ainsi que les places disponibles, les rendez-vous de consultation disponibles depuis l'hôpital, des informations sur le menu dans les restaurants, les prix des produits dans les magasins, les espaces vides dans un parking, etc.

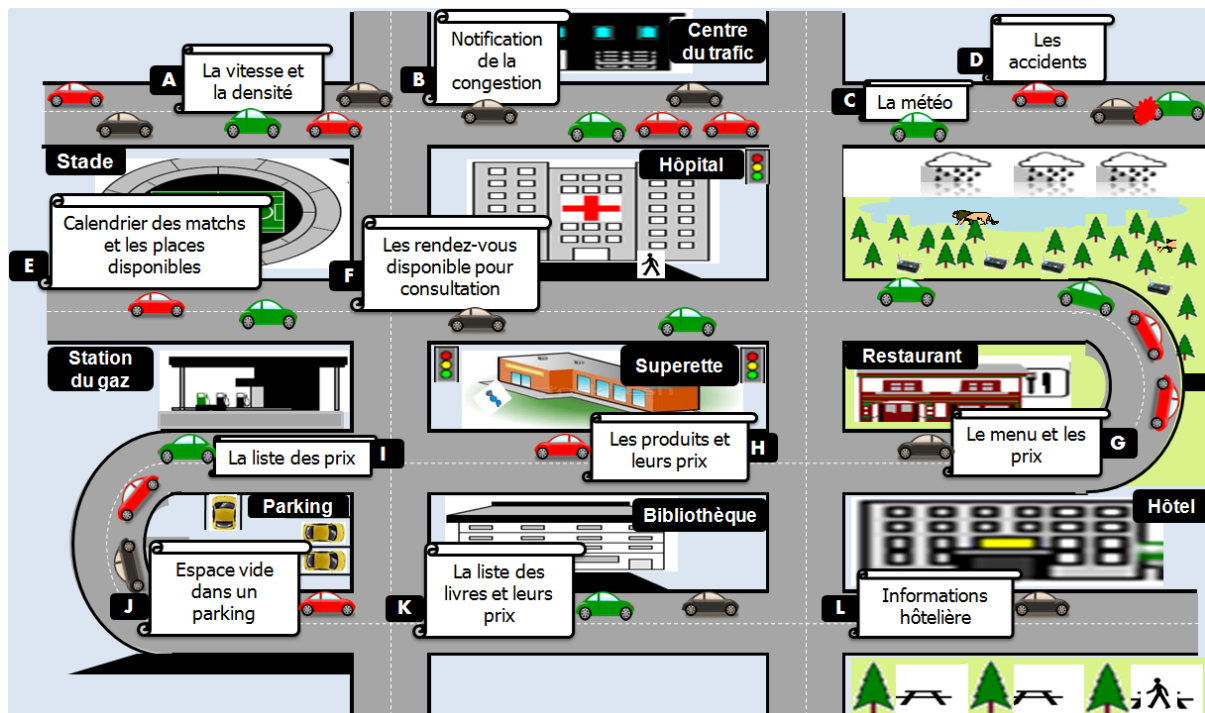


FIGURE 2.1: Types de données potentielles à la collecte de données dans les VANet.

Prises individuellement, les informations collectées par les différents capteurs d'un même véhicule fournissent le plus souvent une connaissance incomplète. Cependant, en confrontant les informations issues de plusieurs véhicules, il devient possible de compléter la perception qu'ont les véhicules de leur environnement. En d'autres termes, en échangeant et en comparant les informations collectées par plusieurs véhicules, l'information devient plus précise et plus pertinente. Ceci montre bien la nécessité de concevoir et de développer des protocoles de collecte de données dans les VANet.

2.2 Les défis de conception d'un protocole de collecte dans les VANet

Par rapport aux réseaux WSNs, les réseaux VANet possèdent plus de performances en terme de ressources. En effet, les véhicules sont dotés de grandes capacités de stockage, de traitement et de source d'énergie illimitée. Néanmoins, les défis de conception des protocoles de collecte dans les VANet sont plus importants. Ils découlent principalement de leurs caractéristiques telles que : la forte mobilité et la densité des véhicules. Parmi ces défis, on trouve essentielle-

ment :

- *Les défis liés à la forte mobilité de véhicules* et par conséquent la connectivité intermittente entre eux, car un lien de communication entre deux véhicules peut rapidement disparaître à cause de la forte mobilité.
- *Les défis liés à la densité* : lorsque la densité de véhicules est faible, les véhicules se déplacent alors en groupes isolés et non connectés, ce qui produit le problème du partitionnement du réseau. Il devient dans ce cas difficile d'assurer les communications de bout-en-bout.
- *Les défis liés au canal radio* : le rôle des techniques d'accès au canal est d'offrir des transmissions *fiabiles, efficace et équitables*. En effet, la fiabilité permet de minimiser les données erronées et la retransmission des données en cas d'erreur. L'efficacité assure la collecte d'un maximum de données visées en un minimum de temps. Tandis que l'équité du médium de communication donne une chance égale aux véhicules pour accéder au canal et envoyer leurs données collectées vers les nœuds collecteurs.
- *Les défis liés à l'application de la collecte*, un protocole de collecte de données doit prendre en considération toutes les exigences de cette application telles que, la durée de la collecte, la zone de la collecte ou bien la zone d'intérêt, le type de données à collecter, la validité de données à collecter.

2.3 Les étapes de la collecte de données dans les VANet

Généralement, la collecte de données dans les réseaux véhiculaires comprend quatre grandes phases (voir la FIGURE 2.2) :

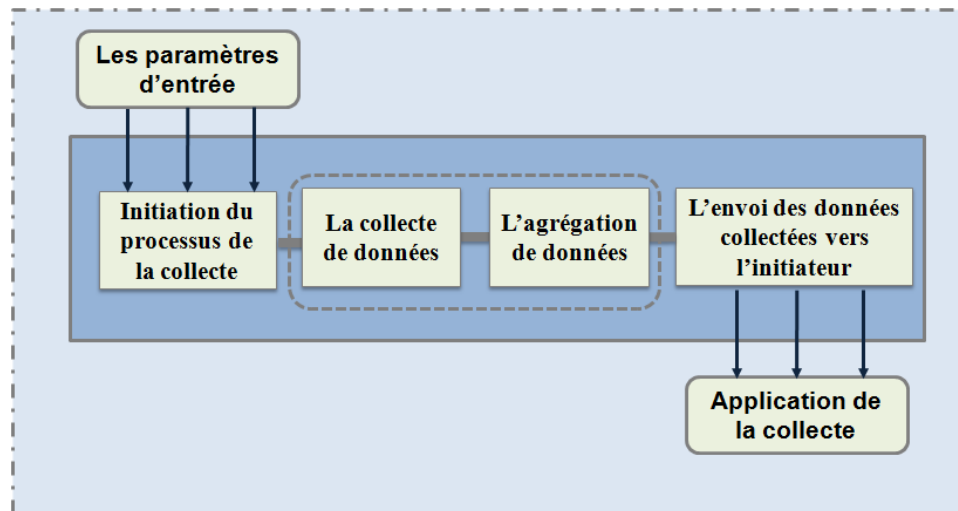


FIGURE 2.2: Les étapes de la collecte de données.

1. **Initiation du processus de la collecte** : durant cette phase et en fonction des exigences de chaque type d'application, un nœud initiateur (véhicule ou RSU) commence par spécifier les paramètres d'entrée du processus de la collecte avant d'informer les autres véhicules. Ces paramètres peuvent être la durée de la collecte, la validité et le type de données à collecter, etc. Le Tableau 2.1 recense les paramètres potentiels d'entrée du processus de la collecte que un initiateur doit spécifier.

Tableau 2.1: Les paramètres potentiels d'entrée du processus de la collecte.

Paramètre d'entrée	Description
Le type de la collecte	Afin de spécifier une collecte en temps réel ou bien tolérante au délai.
Le type et la validité de données	Ils dépendent de l'application (voir la sous-section 2.1.2).
La direction de la collecte sur la route	Qui peut être soit dans la même direction avec le flux du trafic, dans la direction inverse, ou bien dans les deux directions.
La zone de la collecte	Depuis laquelle les véhicules collecteurs doivent collecter les données.

Dans la littérature, on distingue des protocoles qui considèrent la zone de la collecte comme étant une seule zone [41, 42] alors que d'autres la partitionnent en plusieurs segments afin de regrouper les véhicules en plusieurs groupes (clusters) et d'élire dans

chaque groupe un chef de groupe [15, 16, 20, 43]. Dans ce cas, la segmentation peut être soit statique où la zone de la collecte est partitionnée en plusieurs segments de mêmes longueurs avant que la phase de la collecte de données commence [15, 20, 43], ou bien dynamique où la zone de la collecte est partitionnée en plusieurs segments de même longueur lors de la collecte de données par les véhicules eux-mêmes [16].

2. **La collecte de données** : après avoir initié la collecte, les véhicules collecteurs commencent la collecte de données désignées par l'initiateur, depuis les autres véhicules.

Les performances de cette phase dépendent fortement de la méthode d'accès au canal utilisée (voir section 2.2). Ainsi, les protocoles de collecte se basent sur deux grandes classes des techniques d'accès au canal, les techniques dites contention-based [15, 41, 42, 44] et les techniques dites contention-free [16, 20, 43, 45, 46, 47].

– Les techniques contention-based : elles se caractérisent par un accès aléatoire et asynchrone au canal, telles que Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) avec utilisation des RTS, CTS et ACK. On note que la norme 802.11p [8] qui est le standard de communication du VANet (voir section activités de normalisation et de standardisation dans l'introduction générale), se base sur la technique CSMA/CA dont le principe général est le suivant :

- (1) Chaque nœud veut envoyer des données déclenche un back-off pendant un temps aléatoire. Ce back-off est arrêté (pause) une fois le canal est occupé,
- (2) À la fin du temps du back-off, le nœud envoie un message RTS (Request To Send) vers le destinataire,
- (3) Le destinataire accuse le message RTS en envoyant un message CTS (Clear To Send),
- (4) Les autres nœuds qui ont reçu les deux messages RTS et CTS doivent s'abstenir pendant la durée de communication,
- (5) Le nœud émetteur envoie ses données vers le destinataire,
- (6) À la fin de la communication, le nœud destinataire acquitte la bonne réception des données en envoyant un paquet Ack (Acknowledge).

Les méthodes de cette catégorie présentent l'avantage qu'elles ne sont pas sensibles à la forte mobilité des nœuds et aux changements fréquents de la topologie. Cependant, le taux important de collisions et le délai illimité de communication présentent les majeurs inconvénients de ces méthodes.

– Les techniques contention-free : à la différence des techniques de la première classe, ces techniques se basent sur un accès ordonné (structuré) et synchrone au canal. Dans ces techniques, des intervalles de temps, time-slots, sur des bandes de fréquences sont attribués aux nœuds durant lesquelles, ils peuvent accéder au canal. Parmi ces méthodes on trouve :

- (1) La technique TDMA (Time Division Multiple Access) [48] : elle consiste à partager le temps d'utilisation du canal en plusieurs intervalles de temps (time-slots), qui seront alloués aux différents véhicules. Ainsi, la totalité de la bande passante est allouée à un véhicule à un instant donné, ce qui permet d'éviter les collisions des paquets.
- (2) La technique SDMA (Space Division Multiple Access) [21] : représente une classe particulière de TDMA et consiste à diviser chaque voie de la route en plusieurs blocs et allouer un time-slot à chaque bloc. Autrement dit, dans SDMA, un intervalle de temps est alloué à chaque véhicule en fonction de sa position géographique (son bloc). On note bien que l'affectation des time-slots peut être soit dynamique en affectant des time-slots seulement aux blocs contenant des véhicules ou bien statique où un time-slot est alloué à chaque bloc même si ce dernier est vide.

L'avantage de ces méthodes consiste en la résolution du problème de collisions lors de la communication. Cependant, elles nécessitent une grande coordination et une synchronisation entre les nœuds du réseau afin d'ordonner l'accès au canal.

3. **L'agrégation de données collectées** : afin d'améliorer les performances du processus de la collecte, les véhicules collecteurs peuvent aussi agréger les données collectées en éliminant les données redondantes, inutiles et non valides. De plus, l'agrégation de données permet aussi d'économiser la consommation de la bande passante en minimisant le nombre de paquets échangés. Dans [15, 45, 46, 47, 49], les auteurs utilisent une simple technique d'agrégation qui se base sur le calcul de la moyenne. Par exemple, au lieu que chaque véhicule envoie sa vitesse et son identifiant vers les RSUs, les véhicules chefs de groupes agrègent les identifiants et les vitesses de leurs véhicules membres, en envoyant seulement la densité et la vitesse moyenne vers les RSUs. Une autre méthode d'agrégation est utilisée dans [19, 22] qui est celle de Flajolet-Martin sketch [50]. Elle consiste à détecter le nombre des éléments distincts dans un ensemble et donc d'éliminer l'information redondante.
4. **L'envoi de données collectées vers l'initiateur** : une fois le processus de la collecte est

terminé, les véhicules collecteurs doivent délivrer leurs données collectées vers l'initiateur.

2.4 Les travaux connexes à la collecte des données dans les VANet

Plusieurs travaux de recherche portant sur la collecte de données dans les réseaux véhiculaires ont été proposés. Ces travaux visent à organiser la collecte de données dans le réseau routier entre les différents acteurs (véhicules et RSUs). En d'autres termes, l'objectif principal est de définir la façon dont les données vont être collectées par les nœuds du réseau. Dans cette section, nous présentons les principaux protocoles de collecte existant dans la littérature. Puis, nous exposons une classification de ces protocoles.

2.4.1 Les protocoles de collecte de données dans les VANet

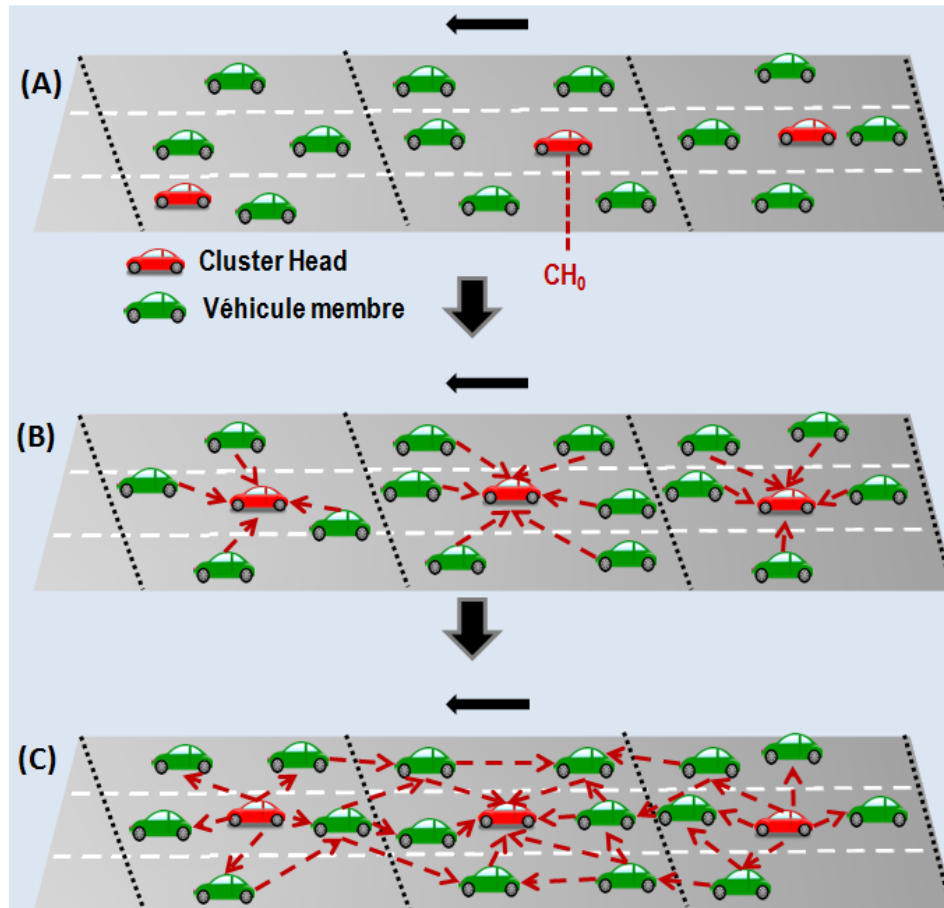


FIGURE 2.3: Les phases du TrafficGather : (A) La répartition des véhicules en plusieurs groupes et élection de CHs. (B) La collecte de données par les CHs. (C) La diffusion des données collectées vers le nœud initiateur (CH_0).

1. **Le protocole TrafficGather** : est un protocole qui se base sur une architecture en clusters dont la segmentation de la route est dynamique [16]. TrafficGather se limite au communication V2V et comprend trois phases (voir la FIGURE 2.3) : (i) Configuration du réseau : durant laquelle un véhicule, Cluster-Head 0 (CH_0), initie le processus de la collecte (voir la FIGURE 2.3-(A)). (ii) Collecte de données : pour chaque groupe, un véhicule CH se charge de collecter les données (voir la FIGURE 2.3-(B)). (iii) Envoi de données collectées vers l'initiateur : tous les véhicules CH envoient leurs données collectées vers le véhicule initiateur CH_0 (voir la FIGURE 2.3-(C)) en adoptant la technique de dissémination «simple flooding» [51], où chaque CH diffuse ses données et chaque véhicule

recevant ces données pour la première fois, il les rediffuse à nouveau jusqu'à atteindre la destination (véhicule initiateur, CH_0).

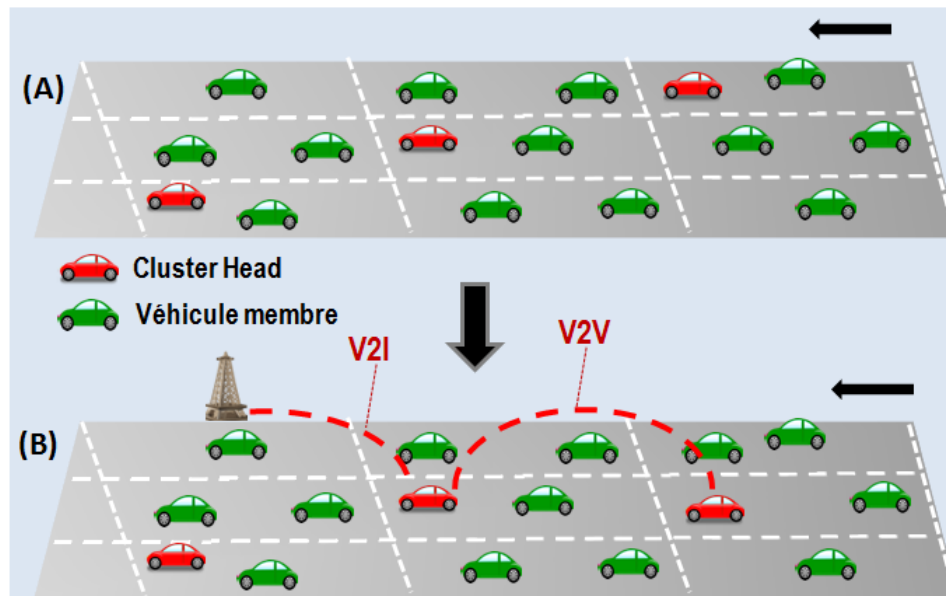


FIGURE 2.4: Les phases du CGP : (A) partitionnement de la route et l'élection des Cluster-Heads. (B) La collecte de données et l'envoi des données collectées vers le RSU via une communication V2V puis V2I.

2. **Le protocole CGP (Clustered Gathering Protocol)** : dans [15], les auteurs ont proposé le protocole CGP qui se base sur une architecture en clusters dont la segmentation de la route est statique, afin de collecter des informations en temps réel sur le trafic routier (voir la FIGURE 2.4-(A)). Par la suite, ces informations sont délivrées vers les centres de contrôle du trafic via les RSUs (voir la FIGURE 2.4-(B)).

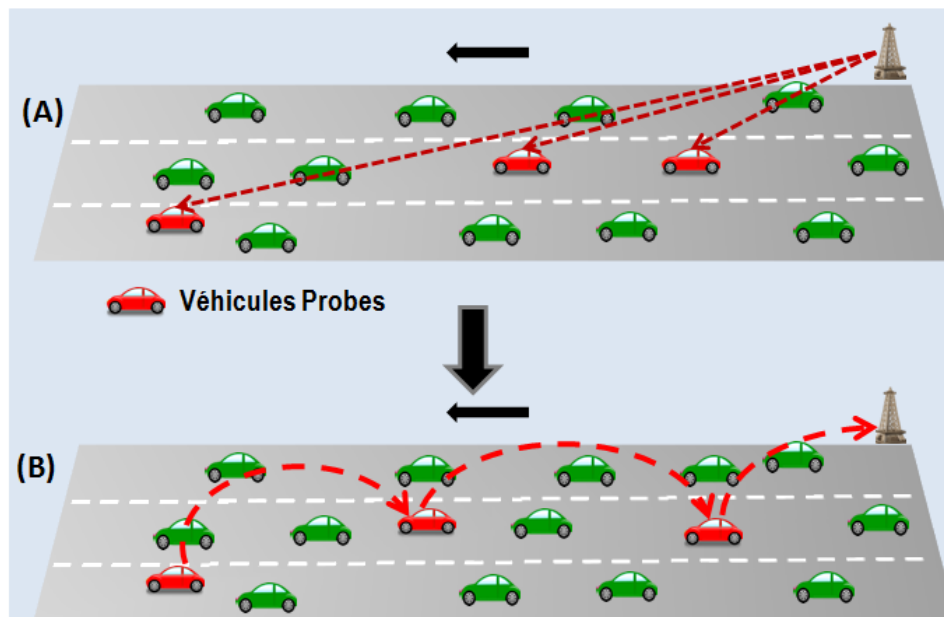


FIGURE 2.5: Les phases du Road Probing : (A) La sélection des véhicules Probes par un RSU. (B) La collecte et l'envoi des données collectées vers le RSU via une communication multi-saut (V2V).

3. **Le protocole Road Probing** [41] : ce protocole permet à un RSU d'initier le processus de la collecte en sélectionnant certains véhicules, nommés probes, qui se chargent de collecter les données du trafic routier. Ces véhicules sont sélectionnés selon leurs temps moyen du trajet sur chaque voie de la route. Le RSU favorise les véhicules qui ont circulé plus sur les voies intermédiaires. Ensuite, Chaque véhicule sélectionné par le RSU a trois responsabilités : (i) collecter des informations sur le trafic routier. (ii) collecter les données envoyées par les véhicules probes devant lui et les agréger avec ses propres données. (iii) envoyer les données agrégées vers les véhicules probes derrière jusqu'à atteindre le RSU initiateur (voir la FIGURE 2.5).
4. **Le protocole COL** [42] : à la différence du protocole Road Probing, le protocole COL permet à un véhicule d'initier le processus de la collecte, en diffusant un message au voisinage contenant les principaux paramètres de la collecte tels que, le type de données à collecter, la durée de la collecte, la profondeur de la collecte exprimée en nombre de sauts, etc. Dans ce protocole, le nombre des véhicules collecteurs augmente au fur et à mesure que le message issu de l'initiateur se propage. Ensuite, chaque véhicule informé diffuse périodiquement les données visées par la collecte à son voisinage. De plus, il remet périodiquement ses données à jour en fonction de celles qu'il a reçues et ce, en éliminant les données redondantes. Enfin, dès que la durée de la collecte est achevée,

les données collectées par l'initiateur sont considérées comme étant le résultat final de la collecte. Ce résultat est ensuite exploité localement par l'initiateur, qui peut l'envoyer aussi à d'autres véhicules ou vers un centre du trafic routier [52].

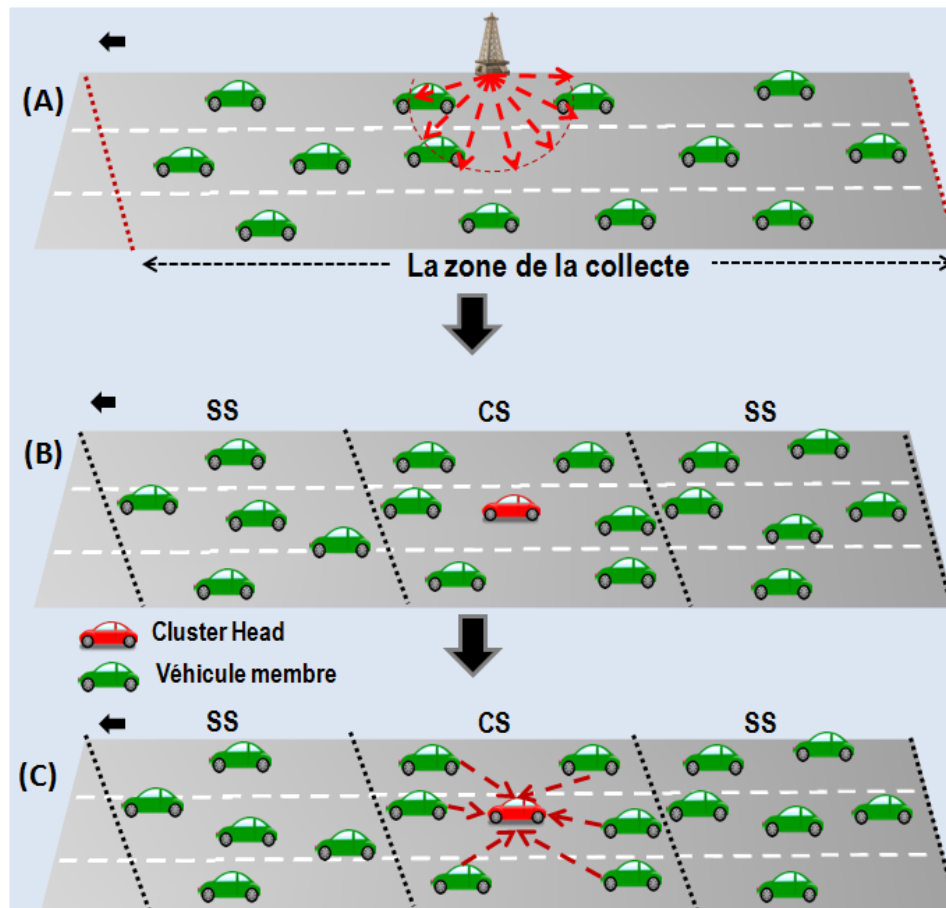


FIGURE 2.6: Les phases du CDGP : (A) Initiation du processus de la collecte par un RSU. (B) La segmentation de la zone de la collecte et l'élection des CHs. (C) La collecte de données par les CHs sur les CSs.

5. **Le protocole CDGP (Clustered Data Gathering Protocol)** : dans [20], nous avons proposé un protocole de collecte de données qui utilise une nouvelle technique d'accès au canal, appelée Dynamic Space Division Multiple Access (DSDMA). Nous déployant aussi dans ce protocole un mécanisme de retransmission des données erronées, afin d'améliorer la fiabilité et l'efficacité du processus de la collecte (voir la FIGURE 2.6). CDGP utilise une architecture en clusters et comprend deux grandes phases : (i) Initiation du processus de la collecte (voir la FIGURE 2.6-(A)), durant laquelle un RSU initie le processus de la collecte en informant les véhicules dans la zone de la collecte. Ensuite, la

zone de la collecte est partitionnée en plusieurs segments de même longueur (voir la FIGURE 2.6-(B)). Pour éviter les problèmes de collisions entre les segments adjacents lors d'une communication V2V, nous avons défini deux types de segments. Le premier type de segment *Collection Segments* (CSs) dans lesquels les véhicules Cluster-Heads (CHs) collectent les données des véhicules membres dans leurs segments. Le second type de segment *Silent Segments* (SSs), où aucune collecte n'est autorisée. (ii) La collecte de données, durant cette phase, les CHs utilisent la technique d'accès au canal DSDMA pour collecter les données des véhicules membres (voir la FIGURE 2.6-(C)).

De plus, dans le but d'améliorer l'efficacité de la collecte du protocole CDGP, en collectant le maximum de données en un minimum de temps, nous avons proposé un autre protocole de collecte, nommé Token-based Clustered Data Gathering Protocol (TCDGP) [43]. TCDGP partage avec CDGP la même phase d'initiation du processus de la collecte. Cependant, TCDGP utilise une autre technique d'accès au canal, proposée dans le contexte de ce travail, nommée Token-based Dynamic SDMA (TD-SDMA), qui consiste à générer et envoyer un paquet Token par les CHs vers les véhicules membres et attendre la réception des paquets de données.

6. **Le protocole LTE4V2X** [45, 46, 47, 53] : est un protocole de collecte qui se base sur l'infrastructure du réseau cellulaire 4G (LTE pour Long Term Evolution) dans l'environnement urbain. LTE4V2X partitionne le réseau véhiculaire en plusieurs clusters gérés et contrôlés par les points d'accès (eNodeBs) du réseau LTE. Ce protocole opère en deux phases : (i) Initialisation, durant cette phase les eNodeBs organisent le réseau véhiculaire en clusters et sélectionnent, dans chaque cluster un véhicule CH. (ii) Maintenance, durant laquelle, les eNodeBs mettent à jour les clusters, et les véhicules CHs collectent les données provenant des véhicules membres et envoient les paquets agrégés vers les eNodeBs.
7. **Le protocole ADOPEL (Adaptive Data cOlllection Protocol using rEinforcement Learning)** : dans [49], les auteurs ont proposé un protocole de collecte dans le milieu autoroutier dans lequel un véhicule initiateur collecte les données depuis ses véhicules voisins avant de les agréger. Ensuite, il les envoie vers le Centre du Contrôle de Trafic (CCT), afin de réaliser plusieurs applications telles que, l'estimation du trafic, la vitesse moyenne des véhicules, la consommation du carburant, etc. ADOPEL se concentre sur la façon dont les données agrégées sont acheminées du véhicule initiateur vers le CCT. Pour cela, il utilise une technique d'apprentissage pour sélectionner, à chaque saut, le véhicule relai le plus approprié et qui offre plus de chance pour maximiser le taux de délivrance des données agrégées.

8. **Le protocole Delay-Bounded Vehicular Data Gathering (DB-VDG)** [44, 54] : conçu pour collecter des données depuis des zones urbaines tout en respectant une contrainte temporelle (la durée de la collecte). Dans DB-VDG, un RSU initie le processus de la collecte en propageant un paquet de requête qui contient essentiellement le type de données, la durée et la zone de la collecte. Puis, tous les véhicules, informés par ce processus, doivent collecter des données depuis cette zone et les envoyer vers le RSU initiateur via une communication multi-saut. De plus, pour minimiser l'overhead¹ et respecter la contrainte temporelle de la collecte, les auteurs ont formulé un problème d'optimisation avant de le résoudre à l'aide d'un algorithme génétique.
9. **Le protocole Real-time traffic Information aware Data collection (RIDE)** : les auteurs dans [55] proposent un protocole de collecte dans une zone urbaine toute en prenant en considération la durée de la collecte et les conditions de trafic telles que le partitionnement du réseau. Dans ce protocole, un RSU initie la collecte en informant les véhicules dans une zone d'intérêt via une communication multi-saut. Ensuite, ces véhicules envoient leurs données collectées vers le RSU initiateur via aussi une communication multi-saut. De même, dans le but de répondre aux exigences de la collecte comme l'overhead de communication, les auteurs dans [56] adaptent deux protocoles de routage de données [57, 58] en relayant les données collectées de véhicules vers les RSUs.

2.4.2 Classification des protocoles de collecte de données

Les protocoles de collecte de données peuvent être classés selon le mode d'accès offert par leurs techniques d'accès au medium, qui peut être soit ordonné (structuré) ou bien aléatoire.

Les protocoles basés sur le mode d'accès au canal ordonné

Dans cette classe [16, 20, 43, 45, 46, 47], les véhicules informés par le processus de la collecte accèdent au canal d'une façon ordonnée et structurée. Cet accès ordonné peut être soit : (i) centralisé à l'aide d'un seul nœud du réseau (CH [16, 20, 43], RSU ou point d'accès [45, 46, 47]), ce qui permet de minimiser les problèmes de collisions lors d'une communication V2V, ou bien (ii) distribué entre les véhicules pour éviter la sélection du nœud central et sa maintenance dans un réseau fortement dynamique.

Généralement, les protocoles de cette classe se basent sur les deux techniques d'accès au canal SDMA et TDMA. Dans le protocole LTE4V2X [45, 46, 47], les auteurs utilisent la technique TDMA dont l'affectation des time-slots est gérée et contrôlée par les eNodeBs du réseau

1. Peut être mesuré comme étant le nombre de paquets échangés dans le réseau.

Tableau 2.2: Comparaison entre les protocoles d'accès ordonné.

Les critères		Les protocoles				
		TrafficGather [16]	LTE4V2X [45, 46, 47]	CDGP [20]	TCDGP [43]	
Les étapes de la collecte	Initiation	X	X	X	X	
	Collecte de données	X	X	X	X	
	Agrégation de données					
	L'envoi de données collectées vers l'initiateur	X	X			
L'initiateur de la collecte	Véhicule	eNodeB	RSU	RSU		
Le type de la collecte	Temps réel	X				
	Tolérante au délai		X	X	X	
Environnement de la collecte	Urbain		X			
	Autoroutier	X		X	X	
Segmentation de la zone de la collecte	Une seule zone					
	Plusieurs zones (clusters)	X	X	X	X	
Le mode d'accès au canal	Ordonné	X	X	X	X	
	Aléatoire					
	Centralisé	X	X	X	X	
	Distribué					
La technique d'accès au canal	SDMA	TDMA	SDMA dynamique	SDMA dynamique		
La retransmission des données erronées			X	X		

cellulaire LTE. Ceci limite la collecte des données seulement dans les zones couvertes par ces points d'accès. Cependant, dans le protocole TrafficGather [16] et dans nos deux protocoles précédents [20, 43], la technique SDMA est utilisée d'une manière centralisée. La différence entre ces deux travaux réside dans l'étape d'affectation des time-slots. Dans le protocole TrafficGather, le principe est de segmenter la route en plusieurs blocs et d'affecter à chaque bloc un time-slot même si le bloc est vide. Ce principe engendre un nombre élevé des time-slots perdus (blocs vides) surtout dans le cas d'un réseau clairsemé. Par contre, dans nos travaux, nous avons adopté la technique SDMA de telle sorte que les véhicules CHs affectent un nombre de time-slots correspondant au nombre de véhicules actifs dans leurs segments de routes, ce qui minimise le nombre des time-slots perdus et par conséquent, le temps d'attente de chaque véhicule avant d'accéder au canal.

À notre connaissance, nous notons qu'aucun protocole de collecte de données n'a été proposé en se basant sur une technique d'accès au canal qui offre un accès ordonné d'une façon distribué entre les nœuds du réseau. Dans le Tableau 2.2, nous synthétisons cette étude comparative entre les protocoles de cette classe, selon plusieurs critères de classifications.

Les protocoles basés sur le mode d'accès aléatoire au canal

Dans ces protocoles, les véhicules informés par la collecte accèdent au canal de façon aléatoire et distribuée entre eux, afin d'envoyer leurs données collectées vers un ou plusieurs nœuds collecteurs. Il est bien évident que les protocoles de cette classe héritent tous les principaux problèmes des techniques d'accès contention-based tels que le taux important de collisions et le délai illimité de communication, qui sont dus principalement au caractère d'accès aléatoire au canal.

Souvent, la plupart des protocoles de cette classe se basent sur le standard 802.11p qui utilise à son tour la technique CSMA/CA [15, 41, 42, 44], sauf le protocole ADOPEL où les auteurs supposent l'utilisation d'une technique d'accès au canal idéale sans aucun problème de collision. Le Tableau 2.3 illustre cette étude comparative entre les protocoles appartenant à cette classe.

2.5 Les métriques d'évaluation des protocoles de collecte de données dans les VANet

Afin de valider les protocoles de collecte dans les VANet, une étape d'évaluation de performances est indispensable et qui peut être établie à l'aide d'une étude analytique en prouvant

Tableau 2.3: Comparaison entre les protocoles d'accès aléatoire.

Les critères		Les protocoles						
		CGP [15]	DB-VDG [44]	COL [42]	ADOPEL [49]	RIDE [55]		
Les étapes de la collecte	Initiation	X	X	X	X	X		
	Collecte de données	X	X	X	X	X		
	Agrégation de données	X	X	X	X	X		
	L'envoi de données collectées vers l'initiateur	X	X		X	X		
L'initiateur de la collecte	Véhicule	RSU	Véhicule	Véhicule	RSU			
Le type de la collecte	Temps réel	X	X		X	X		
	Tolérante au délai							
Environnement de la collecte	Urbain	X	X					
	Autoroutier			X	X			
Segmentation de la zone de la collecte	Une seule zone	X	X	X	X	X		
	Plusieurs zones (clusters)							
Le mode d'accès au canal	Ordonné							
	Aléatoire	X	X		X	X		
	Centralisé							
	Distribué	X	X	X	X	X		
La technique d'accès au canal	CSMA/CA	Standard 802.11p	Standard 802.11p	Technique idéale	Standard 802.11p			
La retransmission des données erronées	X	X	X		X			

leurs corrections et en étudiant leurs complexités [19, 22]. Elle peut être aussi réalisée à l'aide de l'outil de la simulation [15, 41, 42], où on distingue deux aspects de simulation : aspect comportemental et aspect protocolaire. L'aspect comportemental est axé sur la modélisation de la mobilité des véhicules dans le réseau à l'aide d'un simulateur de mobilité tel que SUMO [59]. Dans ce contexte, les modèles de mobilité Freeway et Manhattan [60] ont été proposés pour les deux environnements autoroutier et urbain, respectivement. Le fichier de sortie d'un simulateur de mobilité est introduit par la suite à un simulateur de réseau tel que NS2 (Network Simulator) [61] et OMNet++ (Object Modular Network testbed in C++) [62]. L'objectif d'un simulateur de réseau est de modéliser l'aspect protocolaire qui consiste à modéliser les communications inter véhicules. Par conséquent, la simulation des protocoles et des mécanismes utilisés par les applications réseau.

Pour valider leurs performances, les protocoles de collecte de données sont simulés en variant certains paramètres tels que : le modèle de mobilité, la densité et la vitesse des véhicules, la portée de communication de véhicules et des RSUs.

De plus, les performances d'un protocole de collecte peuvent être évaluées par rapport à d'autres protocoles selon plusieurs métriques de simulation, parmi lesquelles on peut citer :

1. Le taux des données collectées : le pourcentage des paquets de données correctement reçus par rapport à tous les paquets de données reçus.
2. Le taux des données retransmises : le pourcentage des paquets de données erronées et qui ont été retransmis avec succès par les véhicules.
3. Le ratio des données agrégées : le rapport entre les paquets de données envoyés vers l'initiateur et ceux reçus par les nœuds collecteurs.
4. Le temps d'attente des véhicules : le délai d'attente des véhicules avant d'accéder au canal pour envoyer leurs paquets de données.
5. L'overhead des messages générés : l'overhead des paquets de données générés durant tout le processus de la collecte.
6. Le délai : le délai dans chaque phase du processus de la collecte.

2.6 Conclusion

L'objectif de ce chapitre était de présenter un état de l'art sur la collecte de données dans les réseaux véhiculaires. Pour cela, nous avons exposé les travaux de recherche proposés dans la littérature accompagnant d'une classification de ces travaux.

Comme nous avons pu le constater dans cet état de l'art, un nombre important de solutions ont été proposées dans la littérature pour la collecte de données dans les réseaux véhiculaires. Ces solutions ont été motivées principalement par la variété de dispositifs embarqués dans les nouveaux véhicules ainsi que l'évolution des systèmes de communications sans fil, ce qui a permis aussi d'envisager de nombreuses applications pour les systèmes de transport intelligents (ITS).

Par ces connaissances de base sur la collecte de données dans les réseaux Ad hoc de véhicules ainsi que les différents travaux proposés dans ce contexte, nous présentons nos contributions, dans ce contexte, dans le chapitre suivant.

Chapitre 3

Le protocole DDGP : Distributed Data Gathering Protocol for Vehicular Networks

DANS la littérature, plusieurs travaux de recherche ont été proposés dans le but de permettre à un nœud du réseau de collecter des données dans le réseau routier. En effet, un véhicule peut être considéré comme étant un collecteur mobile qui collecte des données liées à des applications tolérantes et non tolérantes au délai telles que les données liées à la sécurité routière, à la gestion du trafic et au confort des usagers de la route (voir section 2.1.2 dans chapitre 2). Dans le chapitre précédent, nous avons pu constater que les performances de ces travaux sont fortement liées à la technique d'accès au canal utilisée. Une technique d'accès au canal efficace et fiable doit : (i) minimiser le temps d'attente des véhicules avant d'accéder au canal, (ii) réduire les problèmes de collisions lors d'une communication sans-fil, et (iii) permettre la retransmission de données en cas d'erreurs.

Dans nos travaux précédents [20, 43], nous avons proposé deux protocoles de collecte CDGP et TCDGP, respectivement. Nos protocoles améliorent les performances du processus de la collecte en implémentant deux nouvelles techniques d'accès au canal accompagnées d'un mécanisme de retransmission des données erronées.

Dans ce contexte, nous avons aussi conçu deux autres protocoles de collecte nommés DDGP pour Distributed Data Gathering Protocol [22] et ECDGP comme extension de CDGP, pour Extended Clustered Data Gathering Protocol [19]. Les caractéristiques principales de nos protocoles sont : (i) ils supportent la collecte de données liées à des applications tolérantes et non

tolérantes au délai, (ii) ils se basent sur une nouvelle technique d'agrégation durant la phase de la collecte pour agréger les données collectées avant de les envoyer vers l'initiateur, (iii) ils implémentent un processus de collecte complet en prenant en considération toutes les phases nécessaires (voir section 2.3 dans chapitre 2) (iv) et ils offrent aux véhicules un accès au canal équitable et fiable grâce à nos techniques d'accès au canal. Nous notons que ECDGP utilise la même technique d'accès au canal avec CDGP (D-SDMA), tandis que DDGP utilise une nouvelle technique, nommée VL-CSMA pour Vehicular Location-based Carrier Sense Multiple Access, qui offre aux véhicules un accès au canal ordonné et distribué.

Dans ce chapitre et vu que ECDGP et DDGP partagent la plupart des phases de collecte, nous avons choisi de présenter seulement le protocole DDGP. Nous commençons par la description de ses phases. Puis, nous élaborons l'évaluation de ses performances via une étude analytique et par simulations.

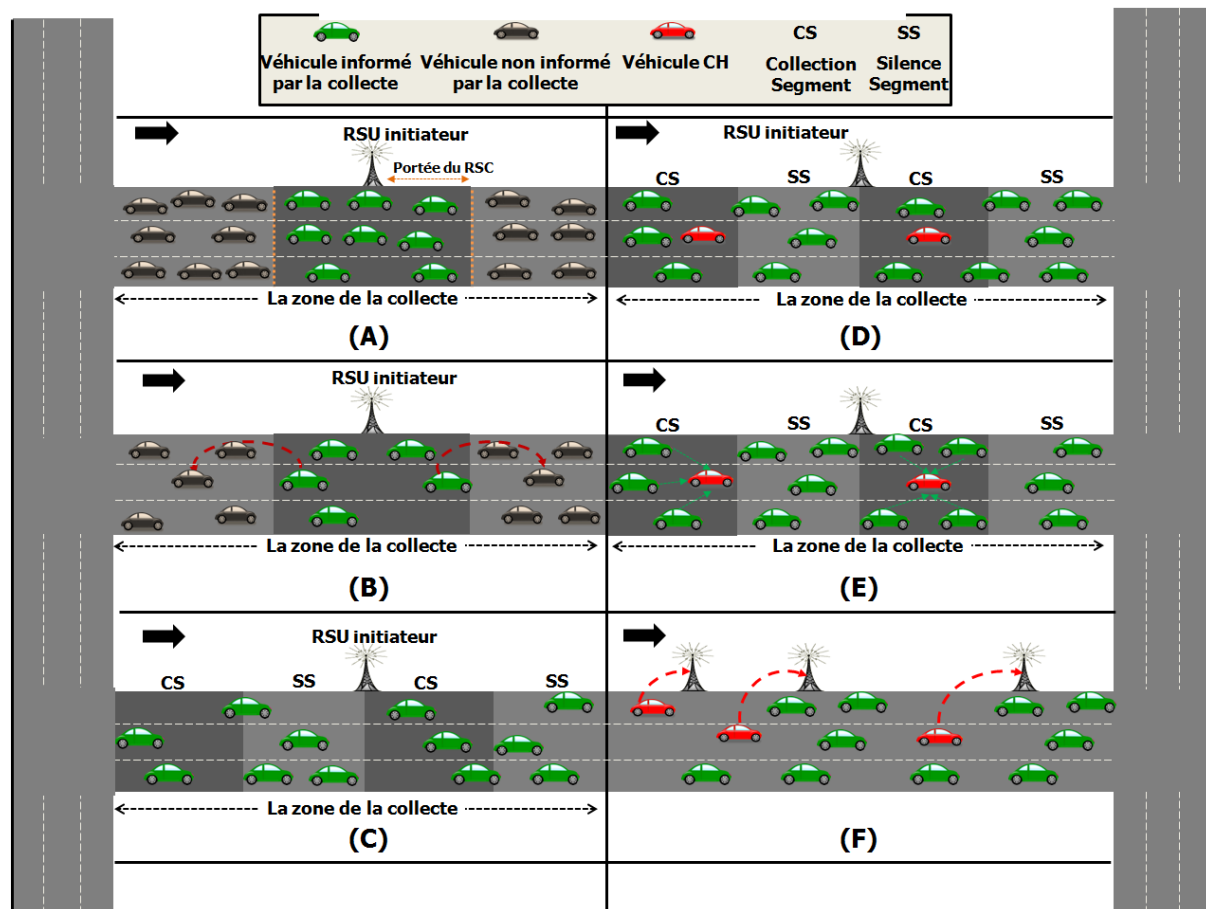


FIGURE 3.1: Un exemple illustrative du fonctionnement des deux protocoles ECDGP et DDGP. (A) Initiation du processus de la collecte. (B) La propagation du paquet Col. (C) La segmentation de la zone de la collecte. (D) La sélection du RVS (ou CHs). (E) La collecte de données par un RVS (CH). (F) L'envoi des données collectées vers l'initiateur.

3.1 Description de notre protocole DDGP

DDGP se déroule en quatre grandes phases qui sont détaillées dans les sections suivantes à travers un exemple illustratif (voir la FIGURE 3.1) : 1) initiation du processus de la collecte, 2) la collecte de données, 3) l'agrégation de données, 4) l'envoi des données collectées vers l'initiateur. Nous notons bien que DDGP partage avec ECDGP les mêmes phases d'initiation, d'agrégation et d'envoi de données collectées vers l'initiateur. Cependant, la différence entre eux réside dans la phase de collecte de données. Avant de présenter les phases de DDGP, nous commençons tout d'abord par présenter le modèle du réseau et les formats de paquets utilisés dans DDGP.

3.1.1 Modèle du réseau

Nous considérons un réseau de véhicules dans les deux environnements autoroutier et urbain. Notre réseau est composé de deux types de composants : Road Side Collectors (RSCs) et Road Vehicles (RVs). RSCs sont connectés entre eux et ils forment un Road Side Backbone (RSB). Nous supposons que les RCSs sont équipés de plusieurs interfaces radio sans-fil. Les RVs sont les véhicules qui peuvent être soit Road Vehicle Collector (RVC) qui se charge de collecter les données depuis le réseau routier, ou des Road Vehicle Sink (RVS) qui sont élus afin de collecter les données des RVCs dans le segment de la route. Dans notre système, nous considérons un segment de route droit de N voies dont la longueur est égale à L_R . Ce segment peut contenir une ou plusieurs zones de collecte (Collection Area CA), de longueur L_{CA} . Il est à noter aussi que dans le modèle de réseau utilisé, la densité du réseau (Density) représente le nombre des RVs dans la zone de la collecte (CA).

3.1.2 Format de paquets

Notre protocole DDGP utilise quatre types de paquets (voir la FIGURE 3.2).

1. **Le paquet Coll** : contient toutes les informations nécessaires pour initier le processus de la collecte (voir la FIGURE 3.2 et le Tableau 3.1).
2. **Le paquet de données (Data)** : contient les données collectées.
3. **Le paquet Acknowledgement** : pour accuser la bonne ou la mauvaise réception du paquet de données. Ce paquet contient l'ID du véhicule concerné par cet accusé.
4. **Le paquet Announcement** : annonce l'élection des Road Vehicle Sinks (RVSs) et initie la collecte de données dans chaque segment de collecte. Il contient l'ID du RVS et l'ID de l'itération de la collecte.

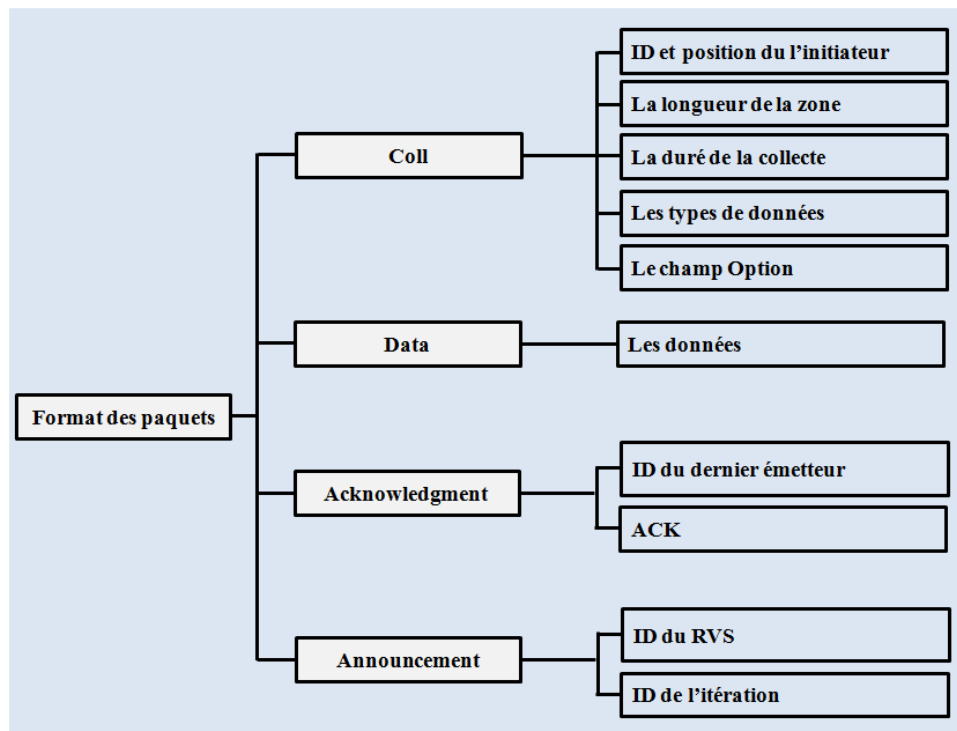


FIGURE 3.2: Les formats de paquets de DDGP.

3.1.3 La phase d'initiation du processus de la collecte

Cette première phase comprend deux étapes :

Étape de propagation du message de la collecte

Durant cette étape, un RSC initiateur informe les RVs dans la zone de la collecte par le processus de la collecte, en diffusant un paquet Coll (voir la FIGURE 3.1-(A)). Ce paquet contient principalement la longueur de la zone de la collecte, le temps de validation des données à collecter (VT) et les données à collecter (DT) (voir la FIGURE 3.2). De plus, le champ *Option* dans le paquet Coll est utilisé pour spécifier le type de données (en temps réel ou bien tolérante au délai), le sens de la collecte et le nombre des types de données (voir le Tableau 3.1). Comme la portée de communication du RSC peut ne pas couvrir toute la zone de la collecte, chaque RV, recevant un paquet Coll, déclenche un timer dont la durée est proportionnelle à la distance qui le sépare à une des deux extrémités de la zone de la collecte¹. À l'expiration du timer et si un RV n'a pas reçu un autre paquet Coll, il doit rediffuser le paquet Coll afin d'informer les autres

1. Extrémité de début si le véhicule est avant le RSC ou bien l'extrémité de fin dans le cas contraire.

Les bits Option	Valeur		Signification
R/D (1-bit)	0		Collecte en temps réel.
	1		Collecte tolérante au délai.
DC (2-bits)	0	0	Non utilisés.
	0	1	La collecte est dans le même sens avec le trafic.
	1	0	La collecte est dans le sens inverse.
	1	1	La collecte est dans les deux sens.
Num_DT (4-bits)			Le nombre de types de données.
Res (1-bit)			Bit réservé.

Tableau 3.1: Les bits du champ Option dans le paquet Coll.

véhicules qui sont dans la zone de la collecte mais hors portée de communication du RSC (voir la FIGURE 3.1-(B)).

Ce mécanisme sélectionne toujours le véhicule le plus éloigné pour rediffuser le paquet Coll. L'opération de rediffusion du paquet Coll se termine lorsque tous les véhicules dans la zone de la collecte sont informés par le processus de la collecte.

Etape de segmentation de la zone de la collecte

Après avoir informé tous les véhicules, la zone de la collecte est partitionnée en plusieurs segments de même longueur L_s (L_s est la portée de communication des RVs). De plus, pour éviter les problèmes de collisions entre les segments adjacents lors d'une communication sans-fil, nous avons défini deux types de segments, le segment *Collection Segment* (CS) dans lesquels les véhicules collectent les données et le segment *Silence Segment* (SS) où aucune opération n'est autorisée (voir la FIGURE 3.1-(C)). Pour déterminer dans quel segment il se trouve, chaque véhicule dans la zone de la collecte utilise la formule suivante :

$$Num_seg(v) = \lfloor D \div L_s \rfloor \quad (3.1)$$

Où D est la distance qui sépare le RV v situé à la position x de l'extrémité de fin de la zone de la collecte.

3.1.4 La phase de la collecte de données

Durant cette phase, un des RVs, dans chaque CS, est sélectionné, pour jouer le rôle d'un Road Vehicle Sink (RVS) (voir la FIGURE 3.1-(D)), afin de collecter les données depuis les

RVCs en utilisant notre technique VL-CSMA (voir la FIGURE 3.1-(E)). Cette phase se compose de deux étapes, la sélection d'un RVS et la collecte de données par les RVSs.

Etape de sélection du véhicule Road Vehicle Sink (RVS)

Pour supporter à la fois la collecte en temps réel et tolérante au délai, nous avons ajouté à notre algorithme d'élection proposé dans [20] des nouveaux critères d'élection permettant la sélection des RVSs selon le type de données spécifié par l'initiateur. Ce nouvel algorithme d'élection hérite tous les avantages de celui proposé dans [20], tels que :

- Il ne nécessite aucun échange de message afin de sélectionner un RVS,
- La sélection des RVSs est dynamique, c.à.d, dès que le RVS actuel quitte son segment de collecte (CS), un autre véhicule sera sélectionné comme étant un RVS.
- Il garantit toujours, qu'à tout moment au plus il y a un et un seul RVS par segment de collecte (CS).

Dans cet algorithme, un RV est élu comme RVS s'il satisfait les critères suivants :

- Le véhicule RV qui a la plus grande quantité de données dans son disque dur afin de minimiser le nombre de paquet de données échangés lors de la prochaine étape.
- Le véhicule RV qui a circulé le plus sur les voies intermédiaires afin de sélectionner le véhicule le plus stable. Pour cela, nous avons défini trois types de voies de l'autoroute, la voie la plus à gauche, la voie la plus à droite et les voies intermédiaires. Dans DDGP, chaque véhicule sauvegarde son profil de mobilité (T_{left} ; T_{right} ; T_{middle}) c'est-à-dire son temps du trajet sur les trois types de voies. Pour choisir un RVS, nous favorisons toujours le RV qui a circulé le plus sur les voies intermédiaires puisqu'il a été démontré que les RVs sur les voies intermédiaires circulent avec une vitesse stable et similaire, donc, ils garantissent une communication V2V plus stable [63].
- Le véhicule RV le plus proche à l'extrémité de début de segment de collecte (CS) puisqu'il dure le plus dans le segment.

Initialement, un RV dès qu'il arrive à un CS, il est en état New (NRV). Puis, il déclenche un Back-off dont la durée est calculée à l'aide de l'équation (3.2) :

$$T_{back-off}(RV) = Max_{back-off} + \frac{1}{Q} \frac{T_{left} + T_{right}}{T_{left} + T_{middle} + T_{right}} + Dist \quad (3.2)$$

Avec $Max_{back-off}$ est la durée maximale pour envoyer un seul paquet par chaque RVs dans un segment CS à forte densité, elle représente le temps d'attente maximum pour recevoir un paquet depuis un RVS. Ce qui permet de déterminer s'il existe un RVS dans le segment CS ou non. Q est la quantité de données collectées par le RV. $Dist$ est la distance qui sépare un RV

situé à la position x de début de segment CS et $(T_{left}; T_{right}; T_{middle})$ est le temps du trajet sur les trois types de voies (la plus à gauche, la plus à droite et les voies intermédiaires).

À l'expiration de ce Back-off, si le RV reçoit un paquet Annonceur d'un véhicule RVS, il passe à l'état RVC. Sinon, il devient le véhicule RVS et il diffuse un paquet Annonceur afin d'informer les autres véhicules RVs de son élection. La FIGURE 3.3 illustre les différentes étapes de notre algorithme d'élection.

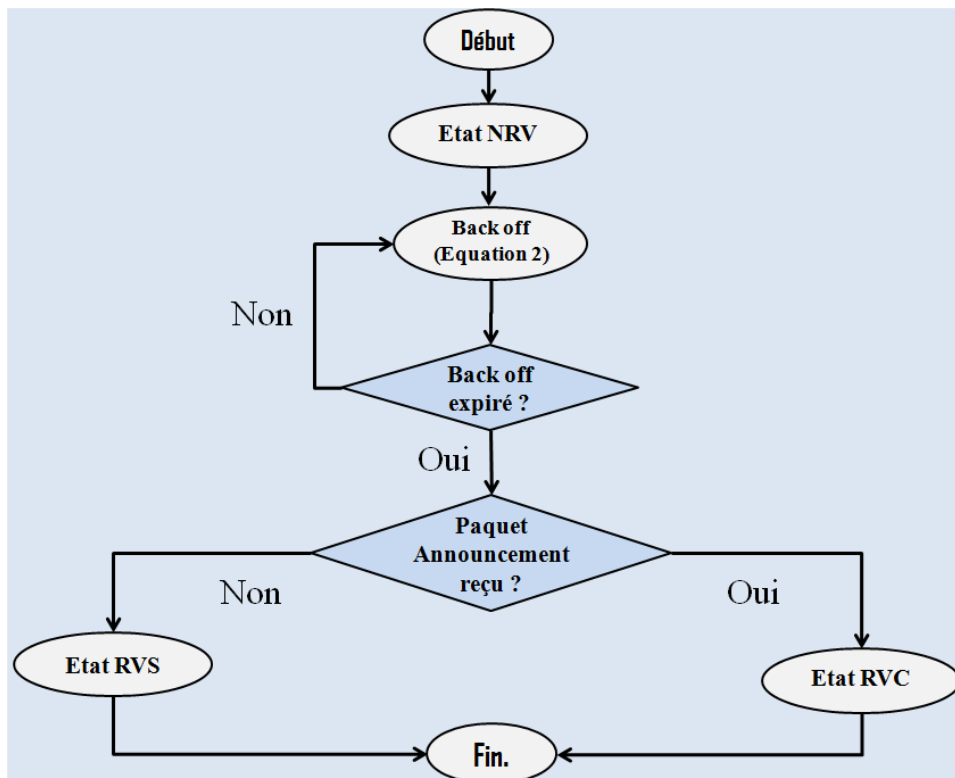


FIGURE 3.3: Notre algorithme d'élection.

Etape la collecte de données

Comme mentionné précédemment, la différence majeure entre nos deux protocoles ECDGP et DDGP réside dans la technique d'accès au canal utilisée durant cette étape. Nous présentons dans ce qui suit ces deux méthodes.

La technique d'accès au canal D-SDMA du protocole ECDGP Dans [20, 19], nous avons proposé une technique d'accès au canal, nommée D-SDMA. D-SDMA affecte des time-slots, de façon dynamique, aux véhicules qui ont des données à envoyer. L'affectation des time-slots est assurée par les RVSs en envoyant périodiquement un paquet de contrôle afin d'informer les RVCs soit par la transmission d'un nouveau paquet de données ou bien la retransmission du

paquet de données de l'itération précédente², en cas d'erreurs.

Dans D-SDMA, chaque véhicule RVC est identifié par son bloc de route depuis lequel il a envoyé ses données. Pour cela, nous segmentons chaque CS en $T = M * N$ blocs de mêmes longueurs, avec N est le nombre des voies, $M = \frac{L_s}{L_b}$ correspond au nombre des blocs par voie dans le CS et L_b est la longueur d'un bloc dans le CS qui est égale à la longueur moyenne des véhicules.

La FIGURE 3.4 illustre un exemple d'une zone de collecte partitionnée en quatre segments (deux CSs et deux SSs) et chaque CS de $N=4$ voies est divisé en $M=6$ blocs/voie et en $T=24$ blocs/CS.

Pour faire face à la forte mobilité des RVCs, chaque RVC lors de l'envoi d'un paquet de données sauvegarde son numéro de bloc courant, pour le comparer avec le prochain paquet de contrôle reçu du RVS. De plus, il sauvegarde aussi les données envoyées pour les retransmettre en cas d'erreur.

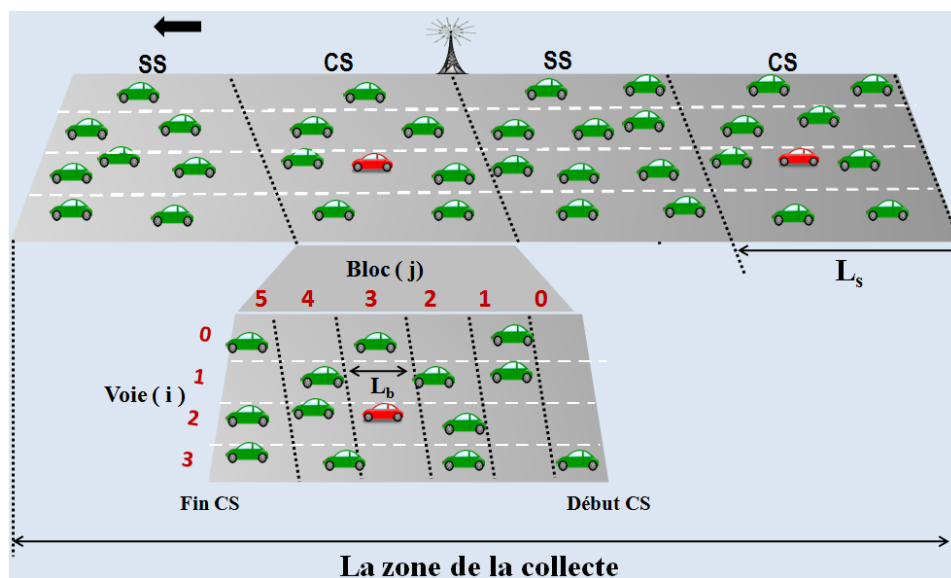


FIGURE 3.4: La segmentation des segments de la collecte (CS).

Même si D-SDMA garantit un temps fini et limité pour accéder au canal (time-slot) et améliore la fiabilité de la communication sans-fil, elle nécessite, cependant, la présence d'un nœud central (RVS) qui gère l'affectation des time-slots et une segmentation délicate de la route en plusieurs cellules (blocs).

La technique d'accès au canal VL-CSMA du protocole DDGP

2. Une itération est le temps nécessaire pour envoyer les données par tous les RVCs une et une seule fois.

À la différence de D-SDMA, la technique VL-CSMA du protocole DDGP offre aux véhicules un accès au canal ordonné, totalement distribué et ne nécessite pas une partition des segments de collecte en blocs. La technique VL-CSMA organise virtuellement chaque groupe de véhicules RVCs dans un CS sous forme d'une topologie en anneau (voir la FIGURE 3.5). Pour envoyer les données collectées vers le RVS, VL-CSMA permet à chaque RVC de déterminer son instant d'accès en utilisant les informations suivantes : (i) sa position géographique. (ii) le numéro de sa voie sur la route. (iii) la position x de début ou de fin de son segment de collecte (CS)).

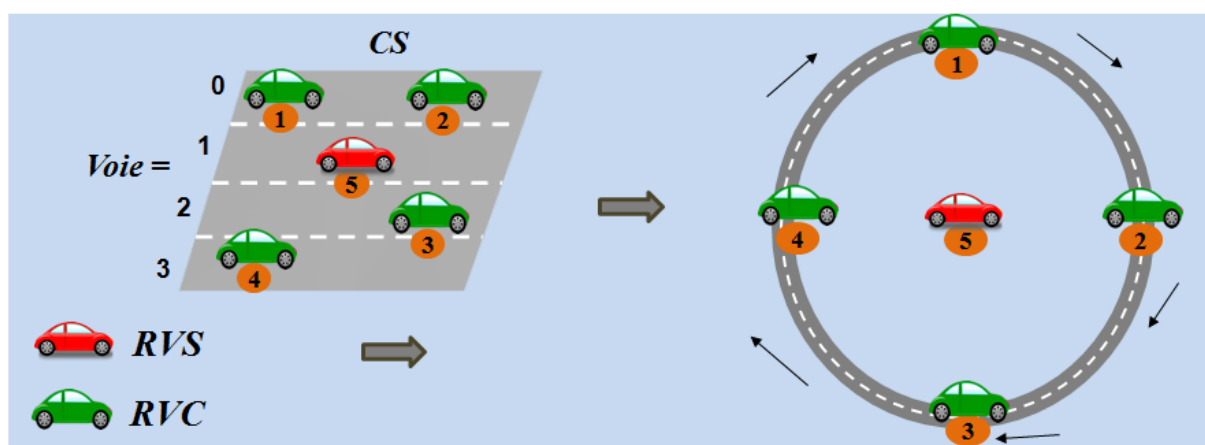


FIGURE 3.5: La topologie en anneau des véhicules RVCs.

A Calcul de back-off par les RVSs :

Après l'étape de sélection des sinks, chaque RVS diffuse un paquet Annonce pour initier la collecte de données dans les segments CSs. Pour assurer la diffusion périodique du paquet Annonce, le véhicule RVS démarre un Back-off dont la durée est calculée par l'équation suivante :

$$T_{back-off}(RVS) = (Last_Turn_index + 1)T_{Data} \quad (3.3)$$

Où $Last_Turn_index$ est l'index du dernier RV dans la topologie en anneau et T_{Data} le temps de transmission d'un paquet de données.

À la réception d'un paquet de données depuis un véhicule RVC, le sink RVS vérifie si le paquet reçu est correct ou non avant de l'accuser. Nous notons que quand le canal de communication est occupé (lors de l'envoi d'un paquet), les Back-offs de tous les RVs doivent être mises en pause durant une période NAV (Network Allocation Vector) calculée à l'aide de l'équation (3.4). À la fin de la transmission (quand le canal sera libre), les RVs reprennent

leurs Back-off, comme illustré dans la FIGURE 3.6.

$$T_{NAV}(RV) = T_{Data} + 2GI + T_{Ack} \quad (3.4)$$

Avec GI (*GuardInterval*) est le temps inter-intervalle séparant la transmission de deux trames et T_{Ack} est le temps de transmission d'un paquet Acknowledgement.

Ainsi, à l'expiration de son Back-off, chaque RVS rediffuse le paquet Annonce pour initier la prochaine itération de collecte et permettre aux véhicules RVCs de retransmettre leurs paquets de données erronés. Il est à noter que pour des contraintes temporelles, la collecte en temps réel se fait en une seule itération seulement.

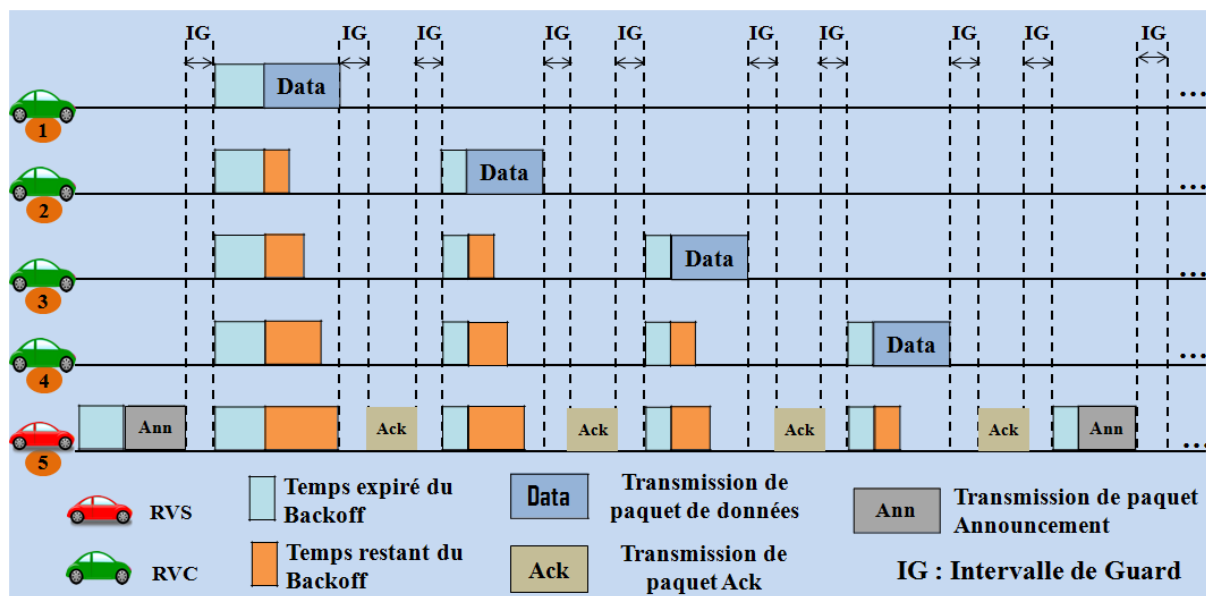


FIGURE 3.6: Le principe de la technique VL-CSMA.

B Calcul de back-off par les RVCs :

Comme montré dans les FIGURES 3.5 et 3.6, les RVCs sont organisés sous forme d'une topologie en anneau et envoient leurs données séquentiellement vers le RVS. Pour ce faire, chaque RVC détermine son tour en fonction de sa position géographique, le numéro de sa voie sur la route et la position x de début ou de fin de son segment de collecte (CS)). À la réception d'un paquet Annonce, il démarre un Back-off dont sa durée est calculée comme suit :

$$T_{back-off}(RVC) = T_{Trans} Turn_index(RVC) \quad (3.5)$$

Où

$$Turn_index(RVC) = \frac{(L_s RVC_lane_num) + (RVC_{xpos} - Startx_pos_{CS})}{Ave_RVC_length} \quad (3.6)$$

Avec $T_{Transmission}$ est le temps minimum pour recevoir les premiers bits d'un paquet de données, Ave_RVC_length est la longueur moyenne d'un RVC et L_s (Longueur de segment). À l'expiration de ce Back-off, le RVC envoie un paquet de données. De plus et pour une collecte tolérante au délai, le RVC attend la réception d'un paquet Acknowledgement depuis le RVS afin de retransmettre les données erronées ou envoyer un nouveau paquet de données lors de la prochaine itération (voir la FIGURE 3.6).

Pour le cas d'un nouveau RV arrivé à un CS, nous distinguons deux cas possibles, soit il reçoit un paquet Announcement et il déclenche un Back-off en utilisant l'équation (3.5), ou bien il reçoit un paquet de données depuis un RVC ou un paquet Acknowledgement depuis un RVS. Dans ce cas, il compare sa position dans l'anneau par rapport au RVC qui a envoyé le paquet de données. S'il est avant le RVC, il attend la réception d'un paquet Announcement qui initié une nouvelle itération de collecte et démarre un Back-off à l'aide de l'équation (3.5). Sinon, il déclenche un Back-off en utilisant soit l'équation (3.7), si les deux RVCs se trouvent dans la même voie, ou bien l'équation (3.8), s'ils se trouvent dans deux voies différentes.

$$T_{back-off}(RVC) = T_{Trans} \left(\frac{(RVC_{xposition} - Sender_{xposition})}{Ave_RVC_length} \right) \quad (3.7)$$

$$T_{back-off}(RVC) = T_{Trans} \left((Endx_pos_{CS} - Sender_{xpos}) + (RVC_{xpos} - Startx_pos_{CS}) + (L_s(RVC_lane_num - (Sender_lane_num + 1))) / Ave_RVC_length \right) \quad (3.8)$$

3.1.5 La phase d'agrégation de données collectées

L'agrégation de données est utilisée pour améliorer l'efficacité de la communication sans-fil en éliminant les données redondantes et expirées. Dans DDGP, l'agrégation de données est réalisée par les RVSS après qu'ils quittent leurs CSs et avant d'envoyer les données collectées vers l'initiateur. À cette fin, nous avons proposé un nouveau mécanisme d'agrégation où chaque entrée ; représente les données collectées dans un CS ; est identifiée par trois éléments : (i) l'identifiant du segment CS (CS-ID), (ii) l'identifiant de l'anneau ($Anneau_i$) qui représente une itération et (iii) une liste chaînée contenant les données collectées (voir la FIGURE 3.7).

RV_{range} et RSC_{range} les portées de communication des véhicules RVs et les RSCs, respectivement. Nous supposons que $RVS = \{rvs_1, rvs_2, \dots, rvs_n\}$ est le sous ensemble des véhicules RVs qui ont joué le rôle d'un Road Vehicle Sink (RVS) dans la zone de la collecte.

Preuve de correction de DDGP

La preuve de correction du DDGP peut être réalisée en deux étapes : (i) prouver que DDGP se termine en un temps fini et il ne se bloque pas, en montrant que chaque phase de DDGP s'exécute en temps fini. (ii) vérifier que le protocole DDGP fournit bien les résultats attendus, c.à.d. il collecte le type de données demandé par l'initiateur de la collecte.

Lemme 1 (la correction de DDGP) : *DDGP se termine en un temps fini et il ne se bloque pas (pas d'état de blocage).*

Preuve : on doit montrer que chaque phase de DDGP s'exécute en un temps fini ce qui peut être réalisé en trois étapes :

- i La phase d'initiation du processus de la collecte : le nœud initiateur informe les RVs dans sa portée par le processus de la collecte en diffusant un paquet Coll. Puis, certains RVs vont rediffuser ce paquet afin d'informer tous les RVs dans la zone de la collecte. Le paquet Coll ne sera pas rediffusé par les RVs qui sont en dehors de la zone de la collecte ce qui démontre la terminaison de cette phase.*
- ii La phase de la collecte de données : dans chaque CS, un véhicule RV est élu comme étant un RVS. Le back-off déclenché par les RVs (équation (3.2)) assure que notre algorithme d'élection s'exécute en un temps fini et sélectionne au plus un véhicule RV en tant que RVS. Ensuite, chaque RVS va collecter les données depuis les RVCs dans son CS avant de les agréger dans le segment SS. Donc, la terminaison de cette phase dépend de la vitesse moyenne des RVCs dans leurs CSs et elle se termine lorsque tous les RVCs quittent la zone de la collecte.*
- iii La phase d'envoi des données collectées vers l'initiateur : en supposant l'existence d'un Road Side Backbone (RSB), la dernière phase de DDGP se termine lorsque tous les RVs dans la zone de la collecte envoient leurs données collectées vers le plus proche RSC, ce qui prouve la terminaison de notre protocole DDGP en un temps fini.*

Lemme 2 (la précision et l'exactitude de DDGP) : *DDGP collecte seulement le type de données spécifié par l'initiateur.*

Preuve : on suppose que DDGP ne collecte pas le type de données spécifié initialement par l'initiateur. Durant la phase d'initiation du processus de la collecte, le nœud initiateur diffuse le

paquet Coll et informe les RVs par le type de données désiré. Puis, lors de la phase de collecte, chaque RVC envoie les données vers le RVS dont leurs types correspondent aux types spécifiés par l'initiateur (contradiction avec la supposition). Cela prouve donc le Lemme 2.

Analyse de la complexité de DDGP

Dans cette section, nous analysons la complexité de notre protocole DDGP en termes de nombre de messages échangés et de délais.

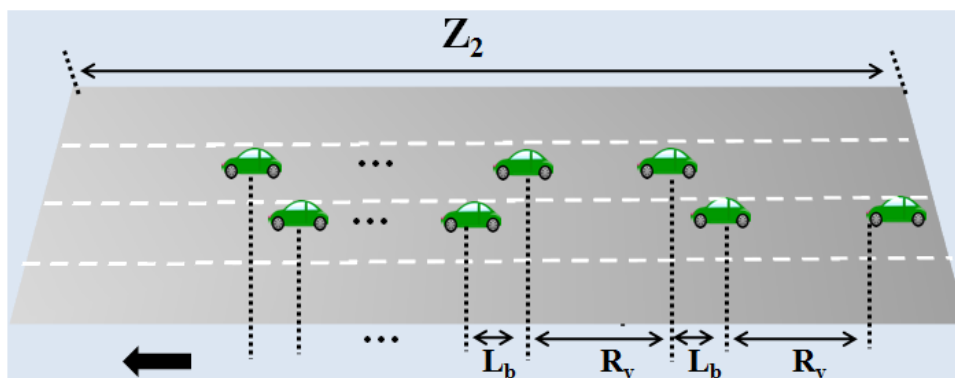


FIGURE 3.8: La retransmission du paquet Coll.

1. La complexité en termes de nombre de messages échangés : le nombre des messages échangés est la somme des messages échangés dans toutes les phases du protocole DDGP.

Lemme 3 (le nombre des messages Coll). Le nombre des paquets Coll diffusés durant la phase d'initiation est donné par :

$$Nbr_Col_msgs = 2 \frac{L_{CA}}{RV_{range} + RV_{length}} + 1 \quad (3.9)$$

Où L_{CA} est la longueur de la zone de la collecte qui est non couverte par le RSC initiateur.

Preuve : initialement, le RSC initiateur diffuse le paquet Coll. Puis, le paquet sera rediffusé par les RVs dans la zone de la collecte. Le nombre des messages est la somme du nombre des messages diffusés dans la zone non couverte par l'initiateur et celui diffusé initialement par le RSC initiateur.

Pour calculer le nombre de messages Coll rediffusés dans la zone L_{CA} et dans le pire des cas, nous distinguons deux cas possibles (voir la FIGURE 3.8). (i) Quand le prochain relai du paquet Coll est le voisin le plus proche du relai actuel, on suppose dans ce cas que la

distance entre eux est égale à $RV_{length} (L_b)$. (ii) Quand le prochain relai du paquet est le voisin le plus éloigné du relai actuel, où la distance entre les deux véhicules est égale à R_v (voir la FIGURE 3.8). En se basant sur ces deux cas, le nombre de retransmission du paquet Coll est égal à la longueur de la zone non couverte par l'initiateur divisée par la somme de RV_{length} et RV_{range} , cette valeur est multipliée par 2.

Lemme 4 (le nombre des messages Announcement). Basant sur le fait que durant cette phase et dans chaque CS, plusieurs sélections des RVSs peuvent s'effectuer et chaque RVS reçoit des paquets de données depuis les véhicules RVCs et envoie périodiquement un paquet Announcement, le nombre de messages Announcement diffusés durant cette phase est donné par :

$$Nbr_Announ_msgs = Nbr_RVS_Selection \left(1 + \frac{\max(Trac_{rv}^{cs})_{rv \in RVS}}{(Last_turn_index+1)T_{Data}} \right) \quad (3.10)$$

Avec

$$Nbr_RVS_Selection = Num_CS \left(\frac{DC_{Duration}}{\min(Trac_{rv}^{cs})_{rv \in RVS}} \right) \quad (3.11)$$

Où $Trac_{rv}^{cs}$ est la durée du trajet du véhicule RVS dans un CS après son élection, $Last_turn_index$ est l'index du dernier RV dans la topologie en anneau, $DC_{Duration}$ est la durée de la collecte, T_{Data} le temps de transmission d'un paquet de données et Num_CS est le nombre total des segments CSs dans la zone de la collecte.

Preuve : le nombre des messages Announcement échangés durant cette phase peut être obtenu en trois étapes :

- i Durant l'étape de l'élection et dans le pire des cas, le nombre des messages envoyés par les véhicules RVSs dans un CS est calculé en divisant la durée de la collecte par la durée minimale du trajet des véhicules RVSs dans les segments CSs. Ensuite, pour calculer le nombre total des messages envoyés par les RVSs, nous multiplions la valeur obtenue par le nombre total des segments CSs (voir équation (3.11)).

ii Durant l'étape de la collecte et après l'élection, chaque RVS collecte les données depuis les RVCs durant plusieurs itérations. Dans le pire des cas et pour un réseau avec une densité maximale, un RVS reçoit les paquets de données de tous les MVs dans son CS, durant son trajet. Pour calculer le nombre des messages envoyés durant la phase de collecte, nous divisons la durée maximale du trajet des véhicules RVSs dans un CS par la durée de réception des paquets de données depuis tous les RVCs, la valeur obtenue est multipliée ensuite par le nombre des élections.

iii Enfin, le nombre total des messages envoyés durant la phase de collecte est le nombre des messages envoyés durant l'étape de l'élection plus ceux envoyés durant l'étape de la collecte (voir équation (3.10)).

Lemme 5 (le nombre des messages de données (Data)). Le nombre des paquets de données est égal à la somme des paquets envoyés durant la phase de la collecte et ceux envoyés durant la phase d'envoi de données collectées vers l'initiateur :

$$Nbr_Data_msgs = Nbr_Data_Collection + Nbr_Data_Delivered \quad (3.12)$$

$$Nbr_Data_Collection = Num_RVS_Selection Num_Data \quad (3.13)$$

$$Nbr_Data_Delivered = \alpha Nbr_Data_Collection \quad (3.14)$$

Avec

$$Num_Data = Num_iter (Max_Density_Seg - 1) \quad (3.15)$$

$$Num_iter = \frac{\max(Tr_{rv}^{cs})_{rv \in RVS}}{Max_Density_Seg T_{Data}} \quad (3.16)$$

Où $0 < \alpha \leq 1$ est le taux d'agrégation des paquets de données et $Max_Density_Seg$ est le nombre maximal de la densité des RVs dans un CS.

Preuve : le nombre de messages de données envoyés peut être obtenu en deux étapes :

i Durant la phase de la collecte : dans le pire des cas et pour calculer le nombre maximum des paquets de données collectées dans un CS, nous calculons, tout d'abord, le nombre maximal des itérations, Num_iter (équation (3.16)). Ensuite, nous le multiplions par $(Max_Density_Seg - 1)$ afin de ne pas prendre en considération le nombre des paquets

Announcement (voir équation (3.15)). Enfin, le nombre des paquets de données est multiplié par le nombre de sélection des RVs (voir équation (3.13)).

ii Durant la phase de l'envoi des données vers l'initiateur : le nombre de messages envoyés durant cette dernière phase dépend du taux d'agrégation des paquets de données. Dans le pire des cas où aucune agrégation n'a été effectuée, le nombre des messages est égale au nombre de tous les paquets collectés par les RVs (voir équation (3.13)). Sinon, le nombre de messages envoyés est le nombre des messages collectés multiplié par le taux d'agrégation (voir équation (3.14)).

Theorem 1 (La complexité des messages). *Le nombre total des messages échangés dans DDGP est donné par :*

$$Msg_Complexity = Nbr_Coll_msgs + Nbr_Data_msgs + Nbr_Announ_msgs + Nbr_Ack_msgs \quad (3.17)$$

Preuve : la preuve de ce théorème est déduite par les lemmes 3, 4 et 5, sachant que le nombre de messages Ack est égal au nombre des paquets de données puisque les RVs envoient un Ack après avoir reçu chaque paquet de données.

Lemme 6 (la complexité temporelle de l'initiation). *La complexité temporelle de cette phase est donnée par :*

$$T_{init} = T_{Trans_{init}}(Coll) + (Num_retr_Coll T_{Trans_rv}(Coll)) + T_{back-off}(ns) \quad (3.18)$$

Où

$$Num_retran_Coll = 2 \frac{L_{CA}}{RV_{range} + RV_{length}} \quad (3.19)$$

Et

$$T_{back-off}(ns) = \sum_{i=1}^{Num_retran_Coll} S_i \quad (3.20)$$

Avec

$$S_i = \begin{cases} L_{CA} + RV_{range} - RV_{length} & i=1 \\ S_{i-1} - (i \bmod 2) RV_{range} - ((i-1) \bmod 2) RV_{length} & \text{Elsewhere.} \end{cases} \quad (3.21)$$

Avec $T_{Trans_{init}}(Coll)$ et $T_{Trans_{rv}}(Coll)$ sont le temps de transmission du paquet $Coll$ envoyé par le RSC et les RVs, respectivement. $T_{backoff}$ est la somme de tous les back-offs déclenchés par les RVs qui ont rediffusé le paquet $Coll$ et L_{CA} , RV_{range} et RSC_{range} sont les mêmes variables utilisées dans lemme 3.

Preuve :

le temps d'exécution de cette phase dépend fortement de la longueur de la zone de la collecte. Le meilleur cas est quand la longueur de la zone de la collecte est inférieure ou égale à la portée de communication du RSU initiateur. Dans ce cas, la complexité est réduite au temps de transmission du paquet $Coll$ $T_{init} = T_{Trans_{init}}(Coll)$. Sinon, nous prenons en considération aussi les durées des Back-offs $T_{backoff}$ plus le temps de transmission du paquet $Coll$ multiplié par le nombre de retransmission du paquet $Coll$ ($Num_{retr_Coll} T_{Trans_{rv}}(Coll)$). Pour calculer le nombre de retransmission du paquet $Coll$, nous nous sommes basés sur le même principe que celui du lemme 3.

Lemme 7 (la complexité temporelle de la collecte, de l'agrégation et de l'envoi des données collectées vers l'initiateur). La complexité temporelle de ces deux dernières phases est donnée par :

$$T_{Collection,Delivery} = \max_{rv \in RVS} (Tra_{rv}^{CA}) \quad (3.22)$$

Avec Tra_{rv}^{CA} est la durée du trajet du véhicule RV dans la zone de la collecte (CA).

Preuve : les deux dernières phases de DDGP se terminent lorsque tous les RVs informés quittent la zone de la collecte et tous les véhicules RVS envoient leurs données collectées vers le plus proche RSC. Pour calculer la complexité temporelle, nous supposons que chaque RV informé sauvegarde sa durée du trajet dans l'autoroute. En effet, le temps d'exécution de ces deux phases est la plus longue durée du trajet des RVS dans la zone de la collecte. c.à.d, la durée du dernier RV qui a joué le rôle du RVS dans la zone de la collecte.

Theorem 2 (La Complexité temporelle). Elle est équivalente à la somme de la complexité temporelle des trois phases du protocole DDGP : initiation du processus de la collecte, la collecte et l'agrégation de données et l'envoi des données collectées vers l'initiateur.

$$T_{DDGP} = T_{init} + T_{Collection,Delivery} \quad (3.23)$$

Preuve : La même que celles des lemmes 6 et 7.

3.1.8 Simulation et résultats

Cette section présente l'évaluation des performances de notre protocole DDGP par simulation dans un environnement autoroutier (Freeway).

Les paramètres de simulation

Pour valider notre protocole et évaluer ses performances par rapport à d'autres protocoles, nous avons simulé DDGP à l'aide des deux simulateurs SUMO [59] et OMNet++ [62]. Dans notre scénario de simulation, les véhicules sont distribués aléatoirement dans l'autoroute et leurs vitesses sont générées entre V_{min} et V_{max} . Nous avons fait varier la densité des véhicules entre 100 et 1000 véhicules dans une zone de collecte de 5 Km. Le Tableau 3.2 illustre les principaux paramètres de notre simulation en plus de ceux cités dans le chapitre précédent (voir section 2.5 dans chapitre 2).

Tableau 3.2: Les paramètres de notre simulation.

Paramètres	Valeurs
La longueur de la zone de la collecte	5 Km
Le nombre des voies	3
La longueur des segments (RV_{range})	100 m
La portée des RSC	500 m
Time-Slot	5 ms
La taille des parquets de données	1500 bytes
La longueur des la route	15 Km

Dans notre simulation, nous avons choisi d'évaluer les performances de DDGP en le comparant avec le protocole CGP et TrafficGather pour les applications en temps réel (RTD) et seulement TrafficGather pour les applications tolérantes au délai (DTD). De plus, puisque le protocole TrafficGather supporte seulement les applications RTD, nous l'avons adopté pour les applications DTD par l'exécution du même protocole mais durant plusieurs itérations. Nous avons choisi d'établir cette étude comparative afin de prouver que : (i) notre protocole DDGP est plus performant que les autres protocoles en termes de certaines métriques de simulation. (ii) La fiabilité du processus de la collecte de données grâce au mécanisme de retransmission de données erronées. (iii) L'efficacité de la technique d'agrégation de DDGP. (iv) les performances de DDGP pour les applications RTD et DTD.

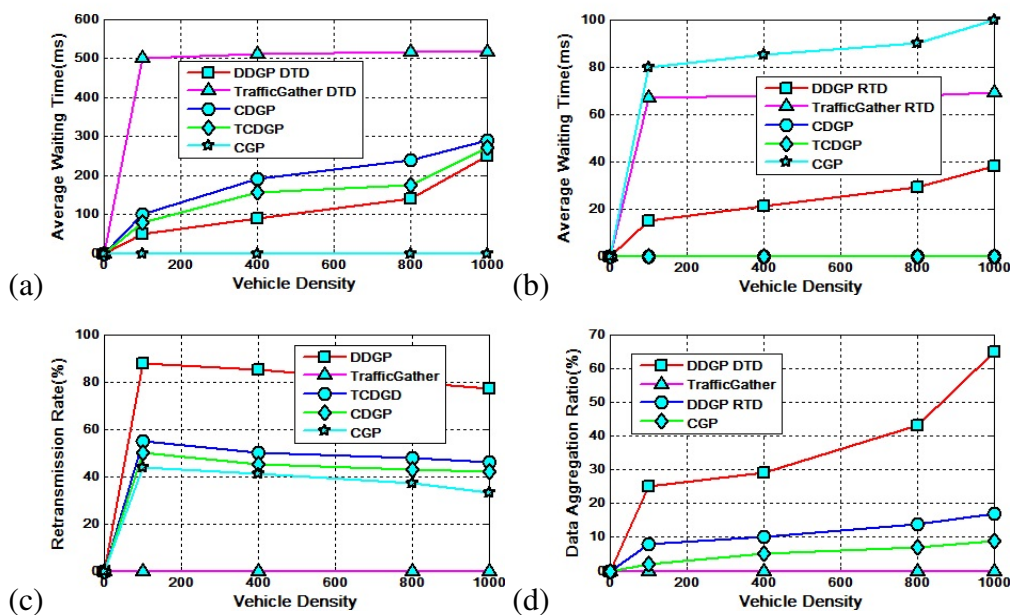


FIGURE 3.9: Evaluation des performances de DDGP dans un environnement autoroutier. (a) le temps d'attente moyen pour les applications DTD. (b) le temps d'attente moyen pour les applications RTD. (c) Le taux de retransmission des données pour les applications DTD. (d) Le taux d'agrégation de données.

Résultats et discussions

Dans cette section, nous présentons les résultats obtenus pour les métriques : le temps d'attente moyen des véhicules, le taux de retransmission des données, le taux d'agrégation de données et l'overhead de la phase de collecte en termes de délai.

Les FIGURES 3.9-(a) et (b) illustrent le temps d'attente moyen pour les applications DTD et RTD, respectivement. Nous remarquons que DDGP est plus performant par rapport à tous les autres protocoles et pour les deux applications DTD et RTD. Ceci est dû à l'utilisation de notre technique d'accès au canal VL-CSMA qui offre aux véhicules un accès au canal distribué et à tour de rôle, c.à.d. VL-CSMA minimise le temps d'attente des véhicules par rapport à la technique SDMA utilisée dans CDGP, TCDGP et TrafficGather, et à la technique CSMA/CA utilisée dans le protocole CGP. Nous notons aussi que pour DTD, le protocole TrafficGather a le temps d'attente le plus long par rapport aux protocoles CDGP et TCDGP à cause de l'utilisation de la technique SDMA d'une façon statique, c.à.d. elle affecte un Time-Slot (TS) à chaque bloc du segment même si le bloc est vide. Dans ce cas, un grand nombre des TSs peut être réservés en vain ce qui augmente le temps d'attente moyen de véhicules dans ce protocole, spécialement dans un réseau à faible densité (moins de 300 véhicules). D'un autre côté, les

protocoles CDGP et TCDGP utilisent la technique SDMA de façon dynamique, ce qui permet aux véhicules CHs d'affecter un nombre des TSs égale au nombre des véhicules membres et qui ont des données à envoyer. Ce qui diminue le temps d'attente de véhicules par rapport à SDMA statique. Cependant, le temps d'attente de TrafficGather est inférieur à celui de CGP pour les applications RTD. Ce qui peut être expliqué par l'utilisation de la technique CSMA/CA par CGP qui est connue par son délai de communication illimité [64, 65].

Le taux de retransmission de DDGP, CDGP, TCDGP et CGP en fonction de la densité est illustré dans la FIGURE 3.9-(c). Nous observons que DDGP a le plus grand Taux de Retransmission (TR) (plus de 88%), ce qui peut être justifié par le fait que le TR de CGP dépend essentiellement des critères de sélection des CHs dans CGP, qui sélectionne toujours le véhicule le plus proche de l'extrémité de fin du segment CS comme CH, ce qui diminue la durée de vie du CH dans le segment CS. Autrement dit, ces CHs ont une grande probabilité de quitter leurs segments CSs avant qu'ils reçoivent tous les paquets de données retransmis par les véhicules MVs, spécialement dans un réseau à forte densité (plus de 500 véhicules). Le même cas peut apparaître dans CDGP et TCDGP où les véhicules CHs envoient leurs accusés de réception aux véhicules MVs à la fin de l'itération de collecte. Cependant dans DDGP, les RVSS envoient un Ack (positif ou négatif) directement après la réception de chaque paquet de données (voir la FIGURE 3.6).

La FIGURE 3.9-d montre le Taux d'Agrégation (TA) de données pour les applications DTD et RTD. Nous remarquons que le TA de DDGP est proportionnel à la densité de véhicules (25%, 29%, 43%, et 65% pour les densités 100, 400, 800, et 1000 véhicules, respectivement). Cela montre bien que l'efficacité de notre technique d'agrégation est proportionnelle à la quantité de données collectée. Autrement dit, notre technique d'agrégation est plus efficace quand la quantité de données collectée est grande. Cependant, le TA pour RTD diminue à 8%, 10%, 14%, et 17% pour 100, 400, 800, et 1000 véhicules, respectivement, car la quantité de données collectée pour les applications RTD diminue, ce qui confirme aussi notre justification sur notre technique d'agrégation. De plus, le taux d'agrégation de DDGP pour RTD est supérieur à celui de CGP, ce qui revient à l'utilisation d'une simple technique d'agrégation dans CGP basée sur le calcul de la moyenne, ce qui prouve aussi l'efficacité de notre technique d'agrégation par rapport à celle du protocole CGP.

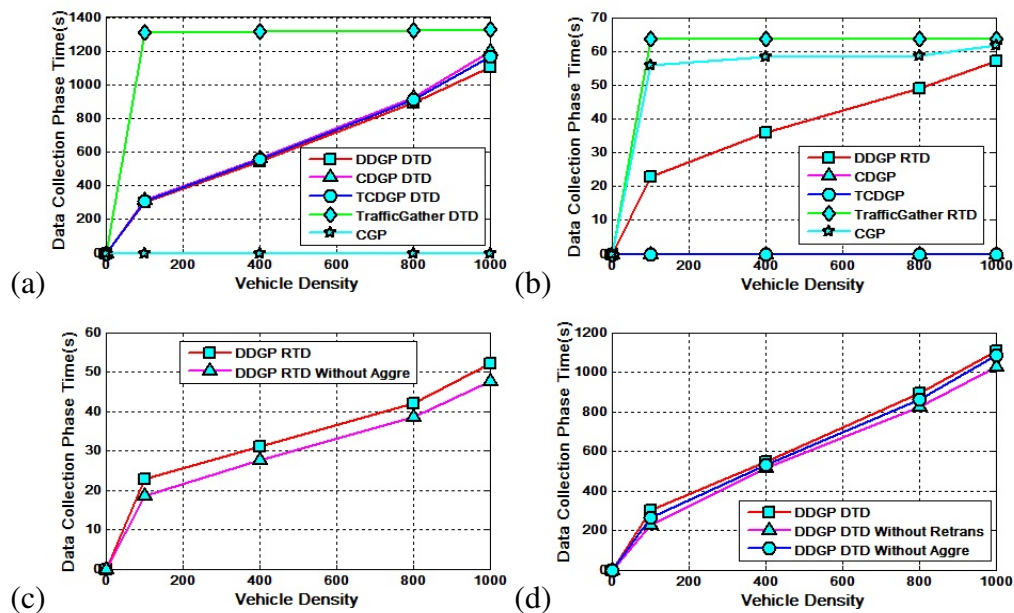


FIGURE 3.10: Evaluation des performances de DDGP dans un environnement autoroutier. (a) Le délai de la phase de la collecte pour les applications DTD. (b) Le délai de la phase de la collecte pour les applications RTD. (c) Le délai de la phase de la collecte pour les applications RTD entre DDGP et DDGP sans agrégation. (d) Le délai de la phase de la collecte pour les applications DTD entre DDGP, DDGP sans agrégation, DDGP sans retransmission.

Les FIGURES 3.10-(a) et (b) illustrent le délai de la phase de collecte de données pour les applications RTD et DTD. Nous remarquons que le protocole TrafficGather a le plus grand délai, ce qui est dû à l'utilisation de la technique SDMA d'une façon statique qui augmente le nombre des TSs perdus (comme expliqué avant) et augmente donc la durée de la phase de collecte. Nous remarquons aussi que DDGP est plus performant que CDGP et TCDGP pour les applications DTD et CGP pour les applications RTD. Comme justifié avant, le délai de communication illimité de CSMA/CA dans CGP, le délai et l'overhead de messages introduit par SDMA dynamique causent l'augmentation de la durée de la phase de collecte.

Le délai introduit par nos techniques d'agrégation et de retransmission pour les applications DTD et RTD sont illustrés dans les FIGURES 3.10-(c) et (d), respectivement. Pour DTD, notre mécanisme de retransmission introduit un grand délai par rapport à notre technique d'agrégation. Cela peut être justifié par le fait que notre mécanisme ait besoin d'utiliser la bande passante afin d'accuser la réception des paquets de données depuis les RVCs. Par contre, notre technique d'agrégation ne nécessite aucune communication sans-fil. De plus, nous déduisons aussi depuis la FIGURE 3.10-(d) que notre technique d'agrégation introduit un délai négligeable (au plus 4s), ce qui prouve aussi l'efficacité de notre technique en terme de délai.

Enfin, nous déduisons que notre protocole DDGP fournit une collecte de données efficace grâce à : (i) Notre technique d'accès au canal VL-CSMA qui offre aux véhicules un accès au canal distribué et fiable en permettant la retransmission de données erronées. L'avantage majeur de VL-CSMA c'est qu'elle ne génère aucun overhead de messages pour organiser l'accès au canal. (ii) Notre technique d'agrégation qui élimine toutes les données redondantes et non-désirées. Par conséquent, elle réduit l'overhead de messages envoyés vers l'initiateur.

3.2 Conclusion

Dans ce chapitre, nous avons présenté notre protocole nommé DDGP. Ce protocole offre un processus de collecte complet implémentant toutes les phases principales (initiation du processus de la collecte, la collecte, l'agrégation des données et l'envoi des données collectées vers l'initiateur). Il garantit la fiabilité du processus de la collecte grâce à l'intégration d'un mécanisme de retransmission en cas d'erreurs et améliore l'efficacité de la collecte par l'utilisation de la technique d'accès au canal (VL-CSMA) et une technique d'agrégation de données proposées aussi dans le contexte de ce travail.

Nous avons évalué ses performances analytiquement en prouvant sa correction et en analysant ses complexités en termes de temps et de nombre de messages échangés, et à travers la simulation en le comparant avec d'autres protocoles.

Les résultats obtenus ont confirmé l'efficacité de DDGP à minimiser le temps d'attente des véhicules avant de pouvoir accéder au canal, à agréger les données collectées en éliminant celles non utiles et à améliorer le taux de retransmission des données erronées et donc la fiabilité du processus de la collecte.

Chapitre 4

Revue de la littérature sur l'offre et la consommation de services dans les réseaux véhiculaires

LES nouveaux véhicules sont de plus en plus dotés des capacités de communication sans fil, des dispositifs de détection : capteurs, radars, caméras, . . . , d'un système de localisation (GPS, Global Positionning System), de grandes capacités de stockage, de traitement et d'énergie illimitée. Néanmoins, ces véhicules passent plusieurs heures soit en état de circulation ou en état stationné, dans des parkings. Pendant cette durée, leurs ressources sont inexploitées et non utilisées. Récemment, l'idée d'exploiter et d'offrir ces ressources de véhicules en tant que services à d'autres véhicules a donné naissance au concept de Cloud véhiculaire [17].

Dans ce contexte, plusieurs travaux de recherche ont été proposés dont l'objectif est de permettre, d'une part, aux véhicules fournisseurs d'offrir leurs ressources en tant que services et d'autre part, aux véhicules consommateurs de découvrir les véhicules fournisseurs et leurs services afin de les consommer.

À travers ce chapitre, nous allons passer en revue les travaux effectués dans le domaine du Cloud véhiculaire, dans lequel se situe notre deuxième problématique traitée dans cette thèse. Nous commençons par donner une idée sur le passage du Cloud Computing au Cloud véhiculaire. Puis, nous présentons les concepts liés au Cloud véhiculaire ainsi que ses défis. Nous terminons ce chapitre par une discussion sur les travaux proposés et nos objectifs de recherche.

4.1 Du Cloud Computing au Cloud véhiculaire

Indéniablement, la technologie de l'Internet se développe de manière exponentielle depuis sa création. Le Cloud Computing (CC) ou l'informatique en nuage reflète l'exploitation des ressources matériels et logiciels des serveurs informatiques distants à travers un réseau, généralement Internet [66]. Ces serveurs sont loués à la demande, le plus souvent par tranche d'utilisation selon des critères de qualités (puissance, capacité de stockage, bande passante, etc.). Divers fournisseurs comme Google, Amazon, IBM offrent une vaste gamme de services en Cloud Computing [67, 68, 69]. Ces services sont généralement répartis en trois classes [70] : (i) *Infrastructure en tant que service (IaaS, infrastructure as a service)* : consiste à offrir des machines virtuelles sur lesquelles le consommateur peut installer un système d'exploitation et des applications, sans devoir acheter du matériel informatique. (ii) *Plateforme en tant que service (PaaS, Platform as a service)* : fournit aux consommateurs un ensemble d'outils, bibliothèques et des langages de programmation afin de générer, déployer et exécuter leurs propres applications. (iii) *Logiciel en tant que service (SaaS, software as a service)* : où des applications sont mises à la disposition des consommateurs.

Avec l'émergence des applications mobiles, le Cloud Computing Mobile (CCM) est introduit comme étant une intégration du Cloud Computing dans l'environnement mobile et qui peut être défini par une infrastructure où le stockage et le traitement de données se produisent en dehors de l'équipement mobile. Ce type de Cloud permet d'accéder à des services qui sont fournis par l'environnement du Cloud Computing à tout moment et n'importe où à partir des appareils mobiles tels que des smartphones et des tablettes [71].

Par ailleurs, le Cloud véhiculaire représente une classe particulière du CCM qui combine le concept du CC et les réseaux véhiculaires. Il offre la possibilité à chaque véhicule disposant de ressources matérielles et logicielles suffisantes, d'offrir et de louer ces ressources en tant que services à d'autres véhicules, et donc d'agir comme étant un serveur du Cloud mobile [17].

Collectivement, les véhicules représentent une infrastructure considérable, en termes de stockage et de traitement. Cependant, les capacités de véhicules ne se limitent pas seulement à ces ressources (comme dans le CC classique), mais aussi aux équipements de détection (capteurs, radars, caméras...). En effet, les véhicules sont des plateformes d'observation idéales de l'environnement routier qui peuvent collecter et stocker des informations avec plus de détail (meilleur que l'infrastructure de détection déployées au bord de la route). De plus, les

informations collectées ont une pertinence locale. Par exemple, un conducteur peut avoir des informations sur les restaurants les plus proches directement depuis les véhicules voisins, au lieu d'accéder à Internet qui est long en termes de délai et cher en termes de coût. Ainsi, la capacité de collecte et la pertinence locale des informations, représentent les avantages majeurs de Cloud véhiculaire par rapport au CC traditionnel.

4.2 Les types du Cloud véhiculaire

Pour former un Cloud véhiculaire, nous distinguons deux types possibles [17, 72, 73] :

- A. **Cloud véhiculaire Statique (Stationnaire)** : dans plusieurs scénarios, le Cloud véhiculaire peut agir comme étant un CC classique. Par exemple, les ressources de véhicules stationnés dans le parking d'une entreprise forment un Cloud véhiculaire stationnaire et ses services peuvent être consommés par les véhicules consommateurs stationnés dans le même parking, par l'entreprise elle-même ou bien par d'autres clients.
- B. **Cloud véhiculaire mobile** : où l'offre et la consommation de services s'effectuent dans un environnement purement mobile. Comme nos travaux de recherche se portent sur ce type du Cloud véhiculaire, nous le présentons en détail dans les sections suivantes.

4.3 Les services du Cloud véhiculaire mobile

Plusieurs services peuvent être offerts dans le Cloud véhiculaire mobile [74, 72]. Nous décrivons ici ces services tout en soulignant qu'il peut en avoir d'autres qui pourraient être ajoutés ultérieurement en fonction des capacités de véhicules fournisseurs et les besoins des consommateurs :

- Réseau en tant que service (NaaS, Network as a Service)

Certains véhicules intelligents ont une connexion permanente à Internet (par exemple, via RSU ou un réseau cellulaire), tandis que d'autres ne l'ont pas. Cela permet aux véhicules intelligents, possédant un accès à Internet, d'offrir leurs bandes passantes à d'autres véhicules.

- Stockage en tant que Service (STaaS, SStorage as a Service)

Les véhicules qui ont une grande capacité de stockage peuvent fournir ce service à d'autres véhicules, qui peuvent avoir besoin de stockage supplémentaire pour faire fonctionner leurs applications, tout en spécifiant la durée et la capacité de stockage et le prix unitaire de stockage.

- Information en tant que Service (INaaS, INformation as a Service)

Les conducteurs peuvent avoir besoin des informations spécifiques telles que, des informations sur les conditions du trafic (congestion, trafic fluide, accidents, etc.). Au lieu d'accéder à Internet pour obtenir ces informations, ils peuvent les demander directement depuis les autres véhicules sur la route, qui ont déjà collecté ces informations.

- Coopération en tant que service (CaaS, Cooperation as a Service)

Les véhicules fournisseurs peuvent offrir leurs coopérations en tant que service afin de réaliser une application spécifique telles que, la collecte de données, la détection des nœuds malveillants dans le réseau, la surveillance de l'état de santé du conducteur, etc.

4.4 Les architectures du Cloud véhiculaire mobile

Dans [75], les auteurs ont proposé une classification des architectures du Cloud véhiculaire mobile :

- **Cloud Ad hoc Véhiculaire (CAV)**

Dans cette architecture, un véhicule offre ses ressources en tant que services (INaaS, STaaS, NaaS, ...) à d'autres véhicules consommateurs, via une communication V2V (voir la FIGURE 4.1).

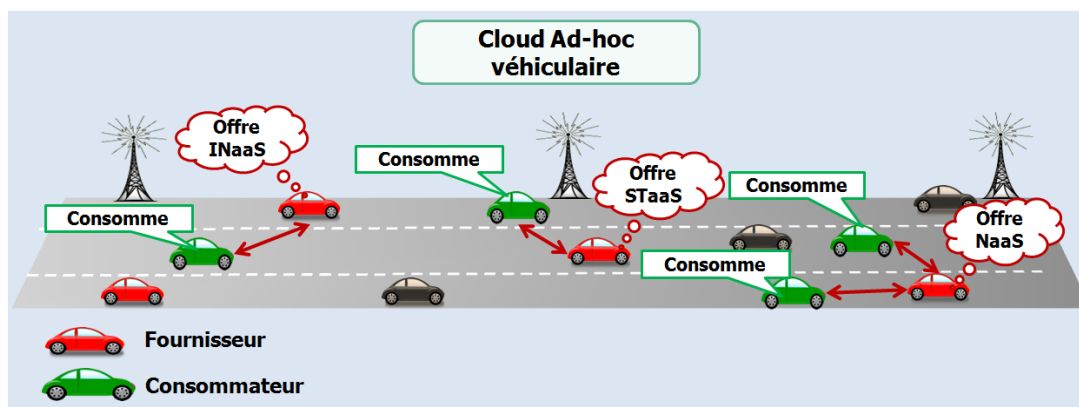


FIGURE 4.1: Le Cloud Ad hoc Véhiculaire.

- **Véhicules utilisant le Cloud Computing (VuC)**

Les véhicules utilisent et consomment les services offerts par le Cloud Computing classique (voir la FIGURE 4.2). Dans cette architecture, les RSUs agissent comme étant des passerelles pour les véhicules consommateurs des services du Cloud Computing traditionnel.

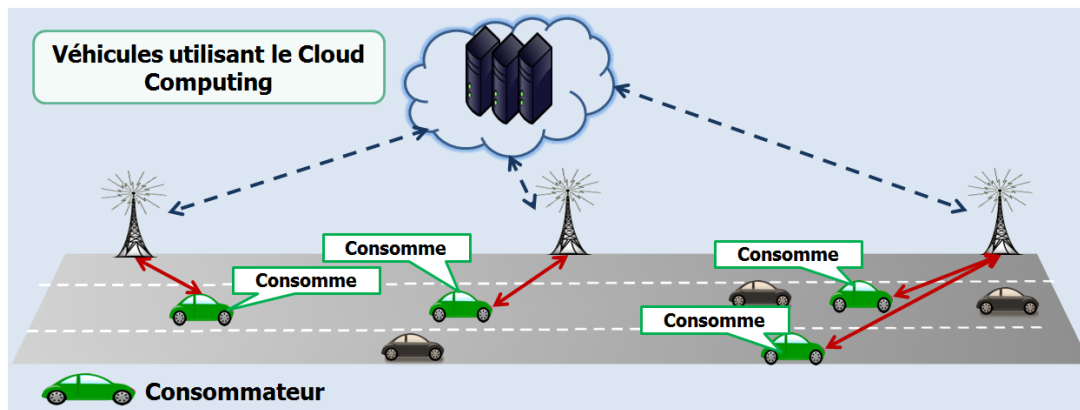


FIGURE 4.2: Les véhicules utilisant le Cloud Computing.

– **Cloud hybride**

L'architecture hybride est la combinaison entre VAC et VuC. Dans cette architecture, les véhicules, qui se déplacent sur la route, peuvent consommer des services soit depuis d'autres véhicules ou bien depuis le Cloud Computing classique à travers les RSUs.

4.5 Les défis de conception d'un protocole d'offre et de consommation de services dans le CAV

Par rapport au CC classique, les fournisseurs de services sont des nœuds mobiles qui peuvent rejoindre ou quitter le Cloud à tout moment. Par conséquent, deux défis majeurs sont posés lors de la conception d'un protocole d'offre et de consommation de services dans CAV :

- *Les défis liés aux réseaux véhiculaires* : ils découlent des caractéristiques des réseaux véhiculaires telles que la connectivité intermittente et le partitionnement du réseau, qui sont dus principalement à la forte mobilité et la diversité de la densité de véhicules (voir section 2.2, chapitre 2).
- *Les défis liés à l'application de l'offre et de consommation de service* : le Cloud véhiculaire offre aux véhicules utilisateurs la possibilité de consommer des services depuis les véhicules voisins, au lieu d'accéder à Internet. Néanmoins, il y a plusieurs défis à relever lors de la conception d'un protocole dans le Cloud véhiculaire, tels que :
 - **La découverte des véhicules fournisseurs et leurs services** : vue la dynamique du réseau véhiculaire, les véhicules consommateurs doivent tout d'abord découvrir les véhicules fournisseurs et leurs services avant de pouvoir en demander.

- **La sélection du service** : après avoir découvert les services offerts, plusieurs fournisseurs proposent des services similaires. Il devient alors nécessaire à l'utilisateur de choisir le fournisseur qui convient le mieux à ses besoins.
- **La consommation de service** : le protocole doit fournir un mécanisme de communication afin de permettre aux utilisateurs de consommer leurs services directement depuis les fournisseurs.

4.6 Les acteurs d'un protocole d'offre et de consommation de service dans CAV

Les acteurs sont les entités qui composent le Cloud véhiculaire à savoir : les fournisseurs de service, les consommateurs de services et le répertoire du Cloud véhiculaire.

1. Les fournisseurs : se sont tous les véhicules cherchant à offrir leurs ressources en tant que services à d'autres véhicules.
2. Les consommateurs : tous les véhicules qui cherchent à consommer des ressources depuis les véhicules fournisseurs.
3. Le répertoire du Cloud véhiculaire : c'est une entité intermédiaire entre les fournisseurs et les consommateurs, dans laquelle les fournisseurs stockent leurs offres et à partir de laquelle les consommateurs découvrent les services offerts. Le choix de cette entité doit être établi de façon optimale afin, d'une part, de donner plus de visibilité aux services offerts et, d'autre part, de faciliter l'opération de découverte de services par les véhicules utilisateurs.

4.7 Les étapes d'un protocole d'offre et de consommation de service dans CAV

Généralement, un protocole d'offre et de consommation de service comprend trois grandes phases (voir la FIGURE 4.3) :

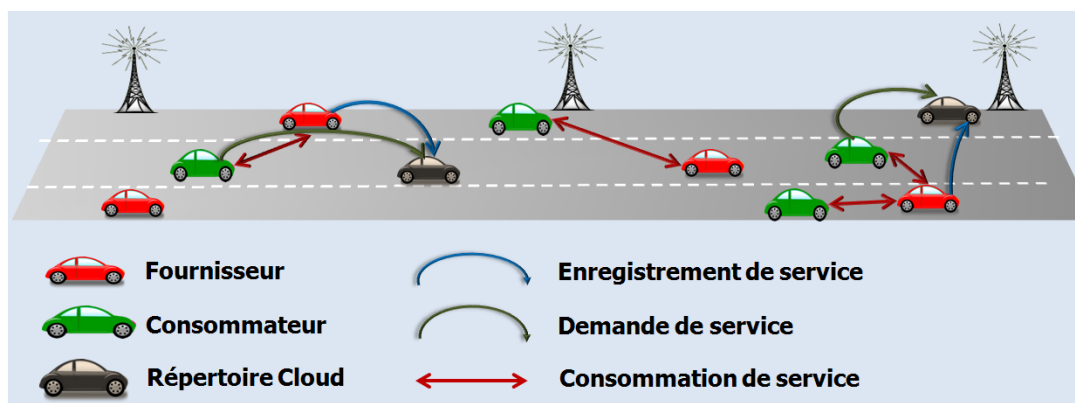


FIGURE 4.3: Les étapes d'un protocole d'offre et de consommation de services.

1. Enregistrement de service par les véhicules fournisseurs : durant cette première étape, les véhicules fournisseurs stockent leurs offres dans le répertoire Cloud. Cependant, ils doivent tout d'abord déterminer leurs services ainsi que leurs attributs.
2. Demande de service par les véhicules consommateurs : lorsqu'un véhicule a besoin de consommer un service, il envoie sa requête vers le répertoire de Cloud le plus proche. Cette demande contient essentiellement le(s) service(s) demandé(s) et les préférences de cet utilisateur.
3. Consommation de service : c'est la dernière étape durant laquelle le véhicule utilisateur sélectionne le fournisseur le plus adéquat en fonction de ses préférences avant de lui envoyer directement sa requête pour consommer le service demandé.

4.8 Les travaux connexes au Cloud véhiculaire

Récemment, plusieurs études de recherche ont été effectuées dans le contexte du Cloud véhiculaire. Ces travaux peuvent être répartis en deux grandes classes : (i) Architectures et protocoles d'offres et de consommation de service [23, 24, 25, 26, 27, 2, 28, 29]. (ii) L'approvisionnement de ressources du Cloud véhiculaire qui gère l'allocation de ressources et la sélection de service ou bien du nœud fournisseur le plus adéquat, pour chaque demande de consommation de service [30, 31].

4.8.1 Les architectures du Cloud véhiculaire

Pour améliorer le confort et la sécurité des conducteurs, les auteurs dans [24] proposent une autre architecture nommée Vehicular Cloud ou bien V-Cloud. Elle comprend trois couches :

Vehicular Cyber-Physical System (VCPS), V2V, et V2I. La couche VCPS est composée de deux types de capteurs, les capteurs internes du véhicule qui se chargent de fournir des informations liées au véhicule telles que la localisation géographique et les capteurs des téléphones intelligents qui sont utilisés dans le véhicule pour contrôler l'état de la santé du conducteur.

Un système de Cloud assisté pour la conduite autonome, appelé Carcel System, a été aussi proposé dans [26]. Ce système collecte les informations fournies par les capteurs des véhicules et par les capteurs de l'infrastructure fixe afin d'assister les véhicules à déterminer des itinéraires les plus sécurisés en évitant les obstacles et les autres véhicules, et des itinéraires plus efficaces en détectant les événements inattendus tels que les accidents et les congestions.

Une autre architecture nommée Vehicular Cloud for Road-side scenarios (VCR) a été proposée dans [25]. VCR vise à offrir différents types de services dans le réseau véhiculaire. Elle définit deux modes de communication sans-fil : public vehicular cloud (V2I) communication, qui permet aux véhicules d'accéder au Cloud Computing traditionnel à travers les RSUs et private vehicular cloud (V2V), à travers laquelle un véhicule peut consommer les services d'un véhicule fournisseur dans le Cloud véhiculaire.

Une architecture nommée VANet-Cloud a été proposée dans [23], qui permet aux utilisateurs du VANet (les conducteurs et les passagers) de découvrir et consommer les services de Cloud soit depuis le Cloud Computing classique tels que Software as a Service (SaaS), Infrastructure as a Service (IaaS) et Platform as a Service (PaaS), ou bien depuis le Cloud véhiculaire. Cette architecture est composée essentiellement de trois couches (voir la FIGURE 4.4) : (i) La couche Client : c'est la couche inférieure de ce modèle qui représente les utilisateurs finaux. (ii) La couche de communication : assure la connexion entre les clients et les deux Clouds (le Cloud Computing et le Cloud véhiculaire). (iii) La couche du Cloud : représente les services offerts par le Cloud Computing ainsi que par le Cloud véhiculaire.

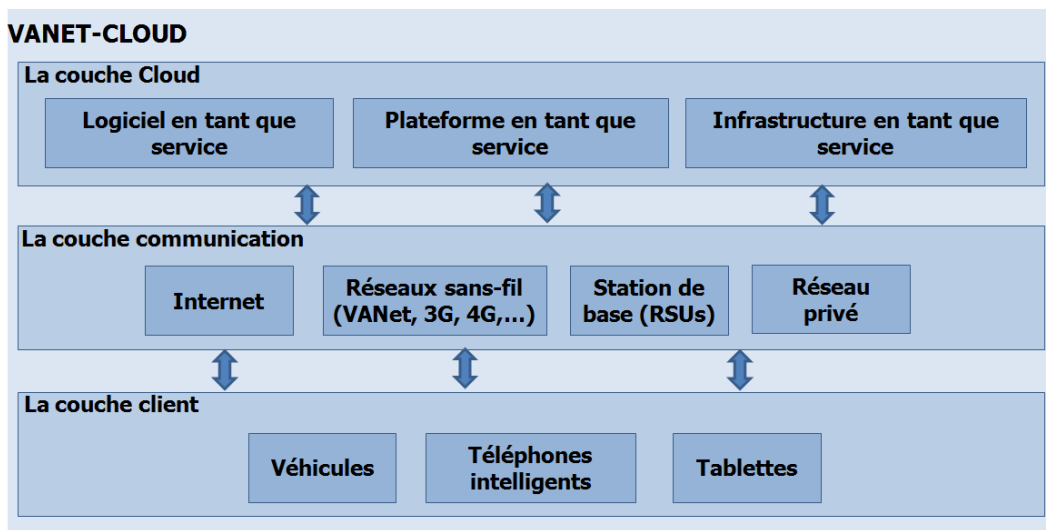


FIGURE 4.4: L'architecture VANET-CLOUD.

4.8.2 Les protocoles d'offre et de consommation de service

Dans [27], les auteurs ont proposé le protocole CROWN qui est conçu pour assurer l'offre et la consommation des services de véhicules fournisseurs, nommés « STAR ». Ce protocole se base sur les RSUs comme étant des répertoires de Cloud et se déroule en trois grandes phases (voir la FIGURE 4.5) : (i) Les véhicules STAR enregistrent leurs services ainsi que leurs niveaux de qualités auprès des plus proches RSUs. (ii) Les véhicules consommateurs découvrent les services offerts auprès des RSUs. (iii) La consommation de service depuis les véhicules fournisseurs en utilisant un protocole de routage proposé par les mêmes auteurs [76].

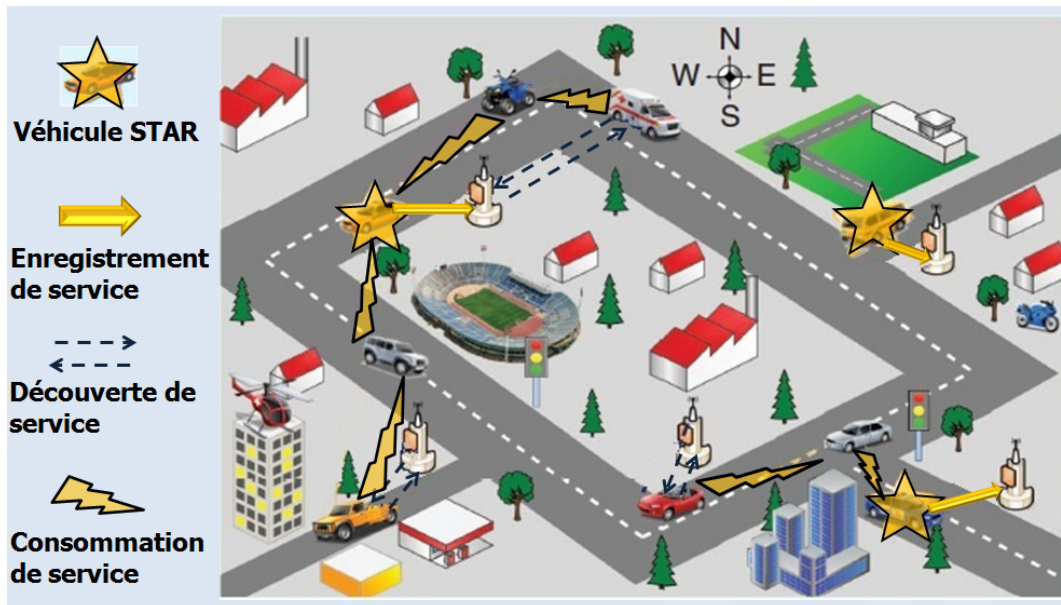


FIGURE 4.5: Le déroulement du protocole CROWN.

Dans [2], les bus publics agissent comme étant des passerelles mobiles afin d'offrir aux véhicules consommateurs l'accès au CC classiques (voir la FIGURE 4.6). Ce protocole se base sur une architecture en trois couches. La couche supérieure « niveau 1 », représente les centres de gestion de VANET ainsi que l'infrastructure de Cloud Computing. La couche intermédiaire « niveau 2 » est constituée uniquement des passerelles mobiles qui sont les bus publics. La couche inférieure « niveau 3 » contient tout autre type de véhicule consommateur (privé et semi-public (taxis)).

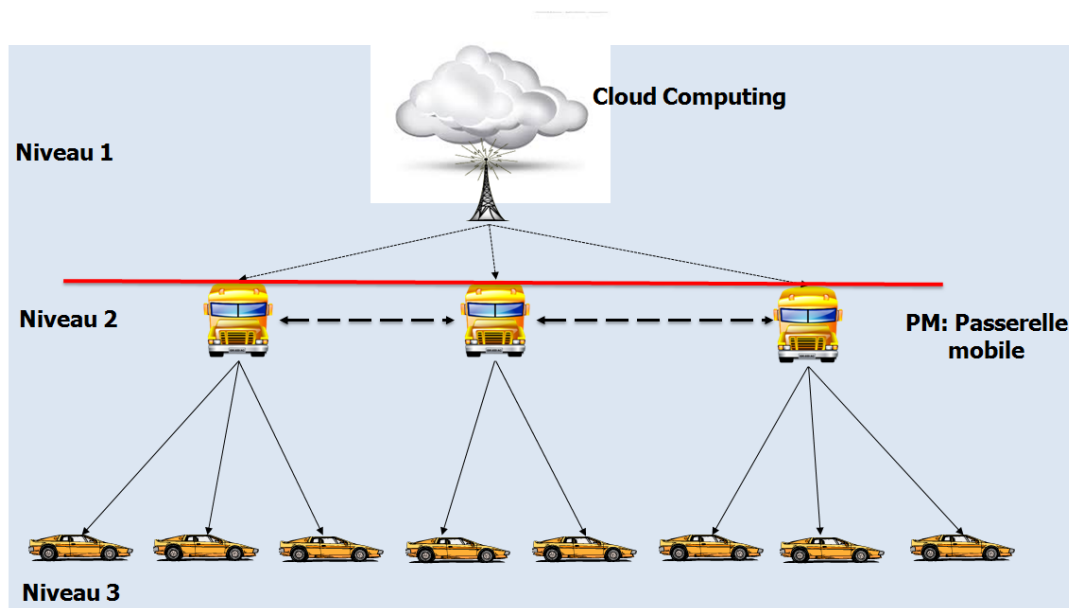


FIGURE 4.6: Les niveaux du protocole proposé dans [2].

Dans [28, 31], les auteurs ont proposé un protocole de formation du Cloud véhiculaire mobile, nommé FcVcA (A Fuzzy clustering-based Vehicular cloud Architecture). Ce protocole se base sur une architecture en clusters en utilisant un nouvel algorithme d'élection des véhicules Cluster-Heads (CHs) basé sur la logique floue [77]. Les véhicules CHs jouent aussi le rôle de contrôleurs qui sont responsables de la création et de la maintenance du Cloud véhiculaire. De plus, quand un véhicule a besoin de consommer un service, il envoie sa requête vers le plus proche véhicule CH qui sélectionne le véhicule fournisseur le plus adéquat en se basant sur une technique Q-learning [78]. La sélection du véhicule fournisseur est effectuée selon plusieurs critères tels que la vitesse moyenne, la bande passante, les ressources de calcul et de stockage.

Dans [30, 79, 80], les auteurs ont proposé plusieurs modèles d'ordonnancement pour l'allocation de ressources des véhicules fournisseurs. En considérant un parking d'un aéroport, les services offerts dans le Cloud véhiculaire sont les ressources de calcul des véhicules stationnés, qui arrivent et quittent l'aéroport à tout moment. Pour éviter le départ d'un véhicule fournisseur pendant qu'il offre ses ressources, ces modèles se basent sur le principe d'affecter la même tâche à plusieurs fournisseurs (affectations redondantes) et gèrent aussi l'affectation des tâches d'un véhicule fournisseur à un autre, en cas de départ.

En considérant un réseau de véhicules stationnés dans un parking, les auteurs dans [81,

82] proposent une technique de sélection de services qui prend en considération le niveau de qualité offert par chaque fournisseur ainsi que les préférences des véhicules utilisateurs. Cette technique implémente un système de négociation entre les fournisseurs et les consommateurs à travers une entité de confiance intermédiaire qui se charge aussi de garantir l'opération de paiement.

4.9 Discussion et objectifs de recherche

Dans cette étude, nous avons effectué une recherche approfondie et détaillée sur le Cloud véhiculaire. Ce concept est devenu un domaine très attractif de recherche pour exploiter les ressources des véhicules. Le Tableau 4.1 résume une étude comparative entre les travaux de recherche, que nous avons présenté dans la section précédente, selon trois grands critères, la classe à laquelle il appartient chaque travail (Architectures et protocoles d'offre et de consommation de service ou approvisionnement de ressources dans le Cloud véhiculaire), le type de Cloud véhiculaire considéré et les acteurs du Cloud véhiculaire et le service offert.

Ce tableau illustre bien que la plupart des travaux se concentrent sur des architectures du Cloud véhiculaire qui visent à faciliter la mise en œuvre de ce nouveau concept. Chaque architecture se compose d'un ensemble de modules (couches) dont chacun a un rôle bien défini. Nous remarquons aussi que peu de travaux ont été proposés dans le contexte des protocoles d'offre et de consommation de service. Les protocoles CROWN et FcVcA permettent la consommation de service dans un environnement purement Ad hoc (CAV) alors que le protocole proposé dans [2] offre aux véhicules la possibilité de consommer les services du Cloud Computing classique (VuC). Nous notons aussi que peu de travaux ont été proposés pour l'approvisionnement des ressources. Ces travaux se concentrent sur le Cloud véhiculaire statique (stationnaire).

Dans ce contexte, notre objectif de recherche est double. Premièrement, concevoir et proposer des protocoles d'offre et de consommation de service dans le CAV en optimisant les délais de découvert et de consommation de service. Deuxièmement, proposer des techniques de sélection de service dans le Cloud véhiculaire mobile qui permettent de sélectionner le service le plus adéquat en fonction des préférences des utilisateurs et les niveaux de qualité fournis par les fournisseurs.

Tableau 4.1: Comparaison entre les travaux proposés dans le Cloud véhiculaire.

	La Classe du travail			Le type de Cloud véhiculaire			Les acteurs du Cloud véhiculaire			Les services	
	Archi- -tecture	Proto- -cole	Approvi- -sionnement de ressources	Statique	Mobile		Fourni- -sieur	Conso- -mmateur	Réper- -toire		
					CAV	VuC	Hybr- -ide				
VANET- CLOUD [23]	X						X				SaaS, IaaS, PaaS, STaaS, NaaS, CaaS, INaaS.
VCR [25]	X						X				
VCPS [24]	X				X						INaaS
Carcel [26]	X				X						
CROWN [27]		X			X				Véhicule	RSU	STaaS, NaaS, CaaS, INaaS
Buses as gateways [2]		X				X			Cloud computing	Cloud Computing	SaaS, IaaS, PaaS
FcVcA [28, 31]		X	X		X				Véhicule	Véhicule	STaaS, NaaS, CaaS, INaaS
Modèles d'ordonnan- cement [30, 79, 80]			X	X					Véhicule	Véhicule	
Technique de négo- tiation [81, 82]			X	X					Véhicule	Véhicule	

4.10 Conclusion

L'objectif de ce chapitre était de présenter une vue d'ensemble du Cloud véhiculaire et d'explorer l'état de l'art dans ce contexte. En effet, le Cloud véhiculaire est un concept récent qui représente une extension du Cloud Computing classique dans les réseaux véhiculaires. L'apparition de ce nouveau concept a été motivée principalement par la grande variété de ressources dont les nouveaux véhicules disposent.

Cependant, même si l'on peut considérer le Cloud véhiculaire comme un cas spécifique du Cloud computing, les solutions existantes pour ce dernier ne sont pas applicables comme telle aux Cloud véhiculaire, à cause de leurs caractéristiques particulières. Il existe donc de nombreuses problématiques de recherche dans le domaine du Cloud véhiculaire. Dans cette thèse, nous nous intéressons à celles liées à l'offre et la consommation de service en état mobile et la sélection de service (fournisseur) en fonction niveaux de qualité offerts.

Chapitre 5

Nos protocoles d'offres et de consommation de services dans le Cloud véhiculaire

CE chapitre présente nos deux protocoles d'offre et de consommation de services, DCCS-VC (Discovering and Consuming Cloud Services in Vehicular Cloud) [32] et RCS-VC (Renting out and Consuming Services in Vehicular Cloud) [33].

Le protocole DCCS-VC dépend essentiellement sur les bus publics qui agissent comme étant des répertoires du Cloud en formant un index dynamique pour les véhicules consommateurs et fournisseurs. Nous avons choisi d'utiliser les bus publics comme répertoires pour les deux raisons suivantes : (i) leurs prédictibilités en temps et en espace dans l'environnement urbain, (ii) ils couvrent la plupart des zones urbaines et donc ils améliorent la connectivité du réseau. Nous notons que notre contribution comprend aussi une nouvelle technique de localisation des véhicules fournisseurs que nous allons détailler dans les sections suivantes.

Cependant, le protocole RCS-VC étend l'infrastructure du réseau de quatrième génération (4G) LTE-A [34] pour l'exploiter comme étant un répertoire de Cloud. Long Term Evolution-Advanced (LTE-A) est l'évolution la plus déployée des réseaux de téléphonie mobile que nous avons choisi pour différentes raisons : (i) LTE-A offre un grand débit de communication allant jusqu'à 1 Go et supporte la forte mobilité des nœuds. (ii) la portée de communication des BS d'environ 1 Km, ce qui le rend adapté aux réseaux véhiculaires. (iii) les nouveaux véhicules sont dotés de plusieurs interface radio telles que 802.11p et LTE (comme dans l'architecture CALM [10]). (iv) LTE-A offre une technique de localisation des nœuds mobiles très efficace à

travers le concept Tracking Area (voir sous-section 5.2.1).

5.1 DCCS-VC : Discovering and Consuming Cloud Services in Vehicular Cloud

Dans cette section, nous présentons notre protocole DCCS-VC.

5.1.1 Vue générale sur le protocole DCCS-VC

Le protocole DCCS-VC permet aux véhicules consommateurs de découvrir les véhicules fournisseurs, de connaître et de consommer leurs services. Dans DCCS-VC, nous utilisons les bus publics comme des répertoires du Cloud pour les services offerts.

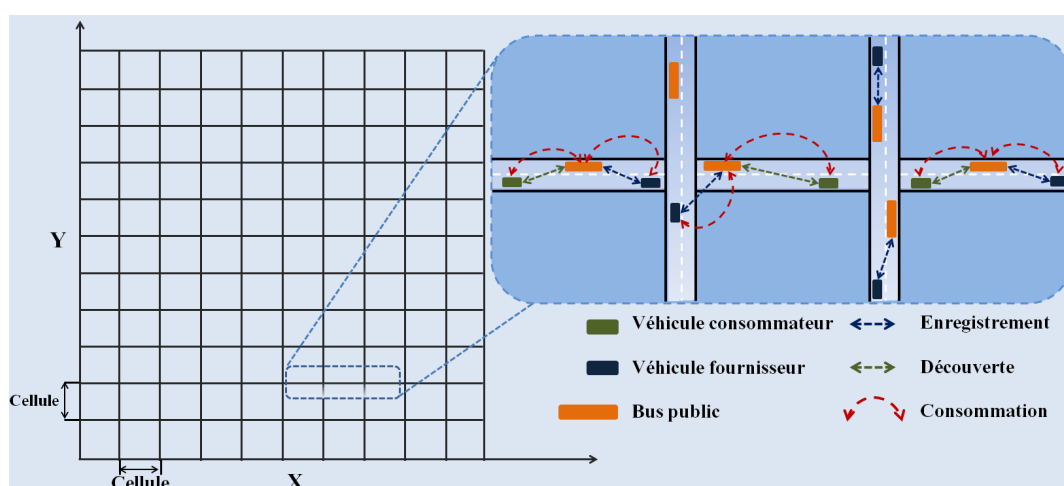


FIGURE 5.1: Vue générale sur DCCS-VC.

De plus, nous proposons une nouvelle technique de localisation des véhicules fournisseurs à base d'une grille, nommée Grid-based Tracking Cell (GTC), qui se base aussi sur les bus publics. Pour cela, nous partitionnons chaque chemin de bus en plusieurs cellules, chacune ayant un identifiant unique. Les cellules sont regroupées en Tracking Bus Path (TBP) et chaque TBP possède également un identifiant unique TBP Identity (TBPI). Puis, quand un véhicule fournisseur enregistrera ses services auprès d'un bus public, il sera informé par son TBPI. En d'autres termes, il sera informé par toutes les cellules couvertes par ce bus public. Par la suite, si le véhicule fournisseur quitte son TBP, c.à.d, il se retrouve dans une cellule qui n'appartient

pas à son TBP, il doit enregistrer à nouveau ses services auprès du bus le plus proche afin de recevoir son nouveau TBPI. La FIGURE 5.1 illustre une vue générale de l'environnement dans lequel DCCS-VC fonctionne.

5.1.2 Les phases du protocole DCCS-VC

Dans cette sous-section, nous exposons en détails les trois grandes phases de notre protocole DCCS-VC à savoir : l'enregistrement de services, la demande de services et la consommation de services. La FIGURE 5.2 montre les différentes étapes de fonctionnement du protocole DCCS-VC que nous les détaillons dans les sous-sections suivantes.

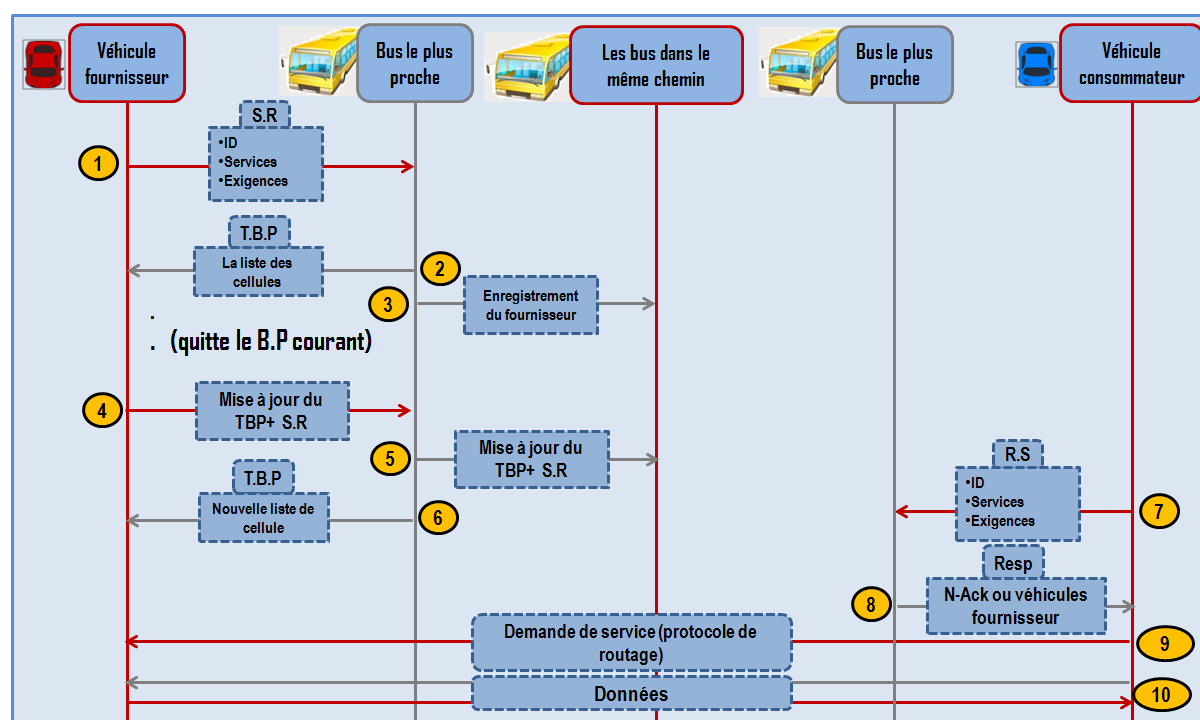


FIGURE 5.2: Les étapes de fonctionnement du protocole DCCS-VC.

Enregistrement de services par un véhicule fournisseur

Quand un véhicule fournisseur décide de participer au Cloud véhiculaire, il doit tout d'abord définir ses services offerts et leurs niveaux de qualité (leurs attributs). Les attributs des services dans Cloud véhiculaire peuvent être classés en deux groupes : les attributs communs et les attributs spécifiques. Par exemple, si le véhicule fournisseur offre le stockage en tant que service (STaaS), il doit définir sa capacité maximale de stockage et la durée de stockage par consom-

Tableau 5.1: Les attributs de services de Cloud véhiculaire.

Les services	Les attributs spécifiques		Les attributs communs		
			Coût du service	Disponibilité	Type
NaaS	Bande passante	Délai	Par heure	Par heure /jour/ semaine.	Service d'Internet
STaaS	Capacité maximale	Durée de stockage par utilisateur	Par unité de stockage		Type de données à stocker
INaaS et ENaaS	Capacité maximale d'information		Par unité d'information		Type d'Information

mateur comme des attributs spécifiques et le coût de stockage par heure, la disponibilité de service et le type de données à stocker comme des attributs communs.

Le Tableau 5.1 montre les attributs communs et spécifiques pour chaque type de service. Après avoir défini ses services, le fournisseur envoie un paquet nommé Service Registration (SR) vers le bus le plus proche (voir étape 1, la FIGURE 5.2). Le paquet SR comprend essentiellement l'identifiant du véhicule fournisseur, les services offerts et leurs attributs. Puis, le bus public envoie un paquet Tracking Bus Path (TBP) vers le véhicule fournisseur pour lui indiquer toutes les cellules qu'il couvre (voir étape 2, la FIGURE 5.2). Nous notons qu'après l'enregistrement de services, le bus informe tous les bus partageant le même chemin de cet enregistrement (voir étape 3, la FIGURE 5.2). De plus, quand le véhicule fournisseur quitte son TBP courant, il doit enregistrer à nouveau ses services et mettre à jour son TBP (voir étape 4, la FIGURE 5.2).

Demande de services par un véhicule consommateur

Quand un véhicule a besoin de consommer un service, il formule tout d'abord un paquet Request for Service (RS) et il l'envoie vers le bus le plus proche (voir étape 7, la FIGURE 5.2). Le paquet RS contient les services désirés et leurs préférences. Par exemple, pour le service de stockage (STaaS), il contient la capacité de stockage maximale, la durée de stockage maximale, le coût maximum de stockage, le type de données à stocker, etc. Le Tableau 5.2 illustre les préférences communes et spécifiques d'un véhicule consommateur pour chaque type de service.

À la réception d'un paquet RS, le bus recherche dans son répertoire tous les véhicules fournisseurs qui peuvent satisfaire cette requête. Par exemple, si le service de stockage est

Tableau 5.2: Les préférences d'un véhicule consommateur pour chaque type de service.

Les services	Les préférences spécifiques		Les préférences communes		
			Coût du service	Disponibilité	Type
NaaS	Min bande passante	Max délai	Par heure	Par heure /jour/ semaine.	Service d'Internet
STaaS	Capacité maximale	Max Durée de stockage	Par unité de stockage		Type de données à stocker
INaaS et ENaaS	Capacité maximale d'information		Par unité d'information		Type d'Information

demandé, le bus recherche tous les véhicules fournisseurs offrant le service STaaS du type de données spécifié, qui ont la capacité et la durée du stockage supérieur à celle spécifiée par le véhicule consommateur, etc. Enfin, si le bus ne trouve aucun véhicule fournisseur, il envoie un paquet RESponse (RESp) contenant un Ack négatif. Sinon, il envoie la liste des véhicules fournisseurs satisfaisant cette requête (voir étape 8, la FIGURE 5.2). Nous notons bien que le bus peut envoyer plusieurs paquets RESp vers le même véhicule consommateur, si ce dernier a demandé plusieurs services.

La consommation de services

À la réception d'un Ack positif, le véhicule consommateur sélectionne le véhicule fournisseur le plus adéquat selon ses préférences. Par exemple, pour le STaaS, le véhicule consommateur sélectionne le véhicule fournisseur qui offre le minimum du coût de stockage ou bien celui qui offre plus de capacité et de durée de stockage. Par la suite, le véhicule consommateur envoie un paquet Request for Ressources (RR) vers le véhicule fournisseur sélectionné. Ce paquet comprend toutes les informations nécessaires pour commencer la consommation de service telles que, le mode de paiement, l'ordre des paquets de données, la technique utilisée pour assurer la sécurité et la fiabilité des données (voir étapes 09 et 10, la FIGURE 5.2).

Pour exécuter les deux dernières étapes du protocole DCCS-VC d'une façon efficace, nous avons adopté un protocole de routage qui se base sur les bus public proposé dans [83]. Ce protocole transmet les paquets de données via les bus seulement. Dans DCCS-VC, nous utilisons les bus public comme des relais pour acheminer les paquets RRs depuis les véhicules consommateurs vers les véhicules fournisseurs et les paquets de données depuis les véhicules fournisseurs vers les véhicules consommateurs. De plus, nous limitons la rediffusion des paquets entre les

Tableau 5.3: Les critères de qualité des bus public.

Critères	Définition	Type
N_{PV}	Nombre d'offres stockées dans le bus	Entier
$A_{storage}$	Espace de stockage disponible dans le bus	Double (Gb)
S_{rate}	Le taux de fournisseurs qui ont réussi à offrir leurs services à travers ce bus	Double (%)
B_{Cost}	Le coût d'enregistrement de service dans ce bus par unité de stockage	Double (\$/Mb)
TBP	Tracking Bus Path (voir sous-section 5.1.1)	Tableau des entiers

bus qui partagent le même TBP afin d'éviter le problème d'inondation du réseau (broadcast storm problem) [84]. Autrement dit, à la réception d'un paquet RR ou bien de données par un bus, il doit vérifier s'il partage le même TBP avec l'émetteur ou non, afin de rediffuser ou bien supprimer le paquet, respectivement.

5.1.3 Sélectionner le meilleur répertoire de Cloud dans DCCS-VC

Dans DCCS-VC, chaque véhicule fournisseur doit publier son offre auprès d'un bus public pour le partager avec les véhicules consommateurs. Cependant, en se basant sur les informations en temps réel sur les bus telles que les horaires et les différentes lignes, le véhicule fournisseur peut découvrir plusieurs bus dans son voisinage. Sélectionner le bus public le plus adéquat, afin de donner une meilleure visibilité aux services Cloud et donc de faciliter l'opération de la découverte par les consommateurs, représente un des défis à relever dans DCCS-VC. Pour cela, nous avons étendu notre protocole DCCS-VC par l'ajout d'une méthode de sélection des bus publics multicritères. Nous avons nommé cette extension FDCCS-VC (Fast Discovering and Consuming Cloud Services in Vehicular Cloud) [85]. Quand un fournisseur découvre plusieurs bus publics dans son voisinage, il leur envoie un paquet Cloud Directory Discovering (CDD). Puis, à la réception d'un CDD, chaque bus répond par l'envoi d'un paquet Cloud Directory Service (CDS) qui comprend les valeurs de ses critères de qualité.

Pour sélectionner le bus le plus adéquat, nous appliquons la méthode de la somme pondérée SAW¹ (pour Simple Additive Weighting) [86]. Nous nous sommes basés sur les quatre critères de qualité présentés dans le Tableau 5.3. Ces critères sont numérotés de 1 à 4, avec $1=N_{PV}$, $2=A_{Storage}$, $3=S_{Rate}$, $4=B_{Cost}$. Nous notons que le critère TBP est utilisé seulement pour favoriser les bus avec lesquels le fournisseur partage une partie du même TBP. Nous considérons un ensemble de bus candidats $C_{Bus} = \{B_1, B_2, B_3, \dots, B_n\}$ qui peuvent agir comme étant des ré-

1. est une méthode multicritère caractérisée par une agrégation additive des critères par sommation pondérée.

pertoires de Cloud pour un véhicule fournisseur. Après avoir considéré les bus candidats et leurs critères de qualité, nous obtenons la matrice de décision $Q = (Q_{i,j}; 1 \leq i \leq n; 1 \leq j \leq 4)$. Le processus de sélection d'un bus comprend deux grandes phases :

1. La phase de normalisation des critères : cette phase a pour objectif de conserver la proportionnalité entre les valeurs avant de les combiner. Nous appliquons les équations (5.1) et (5.2) pour normaliser respectivement les critères à minimiser tels que le coût d'enregistrement de services (B_{Cost}) et ceux à maximiser comme le taux des fournisseurs qui ont réussi à offrir leurs services à travers ce bus (S_{Rate}) :

$$V_{ij} = \begin{cases} \frac{Q_j^{max} - Q_{ij}}{Q_j^{max} - Q_j^{min}} & (Q_j^{max} - Q_j^{min} \neq 0) \\ 1 & (Q_j^{max} - Q_j^{min} = 0) \end{cases} \quad (5.1)$$

$$V_{ij} = \begin{cases} \frac{Q_{ij} - Q_j^{min}}{Q_j^{max} - Q_j^{min}} & (Q_j^{max} - Q_j^{min} \neq 0) \\ 1 & (Q_j^{max} - Q_j^{min} = 0) \end{cases} \quad (5.2)$$

Q_j^{max} et Q_j^{min} sont, respectivement, les valeurs maximales et minimales du critère de qualité j dans la matrice Q , c.à.d, $Q_j^{max} = \text{Max}(Q_{ij})$ et $Q_j^{min} = \text{Min}(Q_{ij})$ $1 \leq i \leq n$. Q_{ij} est la valeur du critère de qualité.

Après avoir normalisé les valeurs des critères, nous obtenons une autre matrice $V = (V_{i,j}; 1 \leq i \leq n; 1 \leq j \leq 4)$, dans laquelle chaque ligne correspond à un bus candidat tandis que chaque colonne correspond aux valeurs d'un des quatre critères considérés.

2. Mise en œuvre de la méthode Somme Pondérée : pour calculer le score final de chaque bus candidat, nous utilisons la formule suivante :

$$\text{Score}(B_i) = \sum_{j=1}^4 (V_{ij} * W_j) \quad (5.3)$$

Avec $W_j \in [0, 1]$ et $\sum_{j=1}^4 W_j = 1$ représente le poids du critère j mesurant son importance par rapport aux autres critères du point de vue véhicule fournisseur. Pour un

ensemble de bus candidats, le véhicule fournisseur sélectionne le bus qui offre le score maximum.

5.1.4 Simulation et résultats

Dans cette section, nous évaluons les performances de DDCS-VC dans l'environnement urbain (Manhattan) par simulation.

Les paramètres de simulation

Pour valider notre protocole et évaluer ses performances par rapport à d'autres protocoles, nous l'avons simulé à l'aide des deux simulateurs SUMO [59] et OMNet++ [62].

Nous avons simulé notre protocole en utilisant un modèle de Manhattan de $9 \times 9 \text{ Km}^2$. Nous avons partitionné cette map en 5 clusters, où dans chacun nous avons fait varier la densité des bus entre 20 et 60. Autrement dit, nous avons 5 TBP différents. La densité des véhicules varie entre 300 et 1500 véhicules, nous considérons aussi trois densités de véhicules fournisseurs : un quart, un tiers et un demi de la densité des véhicules sachant que la valeur utilisée par défaut de la densité de véhicules fournisseurs est un tiers de la densité de véhicules. Les principaux paramètres de notre simulation sont illustrés dans le Tableau 5.4.

Tableau 5.4: Les paramètres de simulation.

Paramètres	Valeurs
Temps de simulation	1000s
Zone de simulation	9x9 Km ²
Portée de communication	500 m
Débit	18 Mb/s
Densité des bus par TBP (PBD)	20, 40, 60 bus
Densité des véhicules fournisseurs (P)	1/4, 1/3 , 1/2 de la densité de véhicules
Taille des paquets Registration, TBP, Request, Response	128 octets
Taille des paquets de données	[1-5] Ko
Taille de la file d'attente des véhicules fournisseurs	5 véhicules consommateurs (CV)
Le nombre maximum de services offerts par véhicule fournisseur (PV)	3 services
Le nombre maximum de demande de services par véhicule consommateur (CV)	2 services

Durant les 300 premières secondes de simulation, chaque véhicule fournisseur doit enregistrer un ou plusieurs services auprès du bus le plus proche. Ensuite, les véhicules consommateurs commencent à envoyer leurs demandes de services. À la réception d'un paquet RR par le véhicule fournisseur depuis un véhicule consommateur, si le véhicule fournisseur peut satisfaire la requête, il répond positivement à la demande du véhicule consommateur. Sinon, il répond par un Ack négatif. Pour évaluer DCCS-VC, nous nous sommes basés sur les métriques de simulation suivantes :

1. Le Délai de Découverte de services (DD, Discovering Delay) : mesure la durée entre l'envoi d'un paquet Request for Service (RS) et la réception d'un paquet de réponse par un bus.
2. Le Délai de Consommation de services (CD, Consuming Delay) : mesure la durée entre l'envoi d'un paquet Request for Resources et la réception de tous les paquets de données que nous avons limité à trois paquets.
3. Les acquittements positifs et négatifs (Pos-ACK et Neg-ACK) : le nombre des acquittements positifs et négatifs envoyés par un bus.

4. Trafic des Véhicules (VT, Vehicle Traffic) : le trafic généré, envoyé et reçu par un véhicule.
5. Le Délai de bout en bout des paquets de données (E2ED, End to End Delay) : mesure la durée moyenne durant laquelle les paquets de données sont acheminés depuis les véhicules fournisseurs vers les véhicules consommateurs, à travers les bus.

De plus, afin de prouver l'efficacité de notre protocole, nous l'avons comparé avec d'autres protocoles : (i) Broadcast DCCS-VC dans lequel nous avons remplacé le protocole de routage de DCCS-VC par un autre protocole de diffusion. Dans ce protocole et durant la dernière phase du DCCS-VC, chaque paquet diffusé par un véhicule (ou bien un bus) sera rediffusé par ses voisins jusqu'à ce que le paquet soit reçu par la destination. (ii) le protocole CROWN [27] en termes de DD, CD, VT et E2ED. (iii) le protocole B-CROWN [27] dans lequel toutes les opérations du CROWN sont remplacées par des opérations de diffusion.

Résultats et discussion

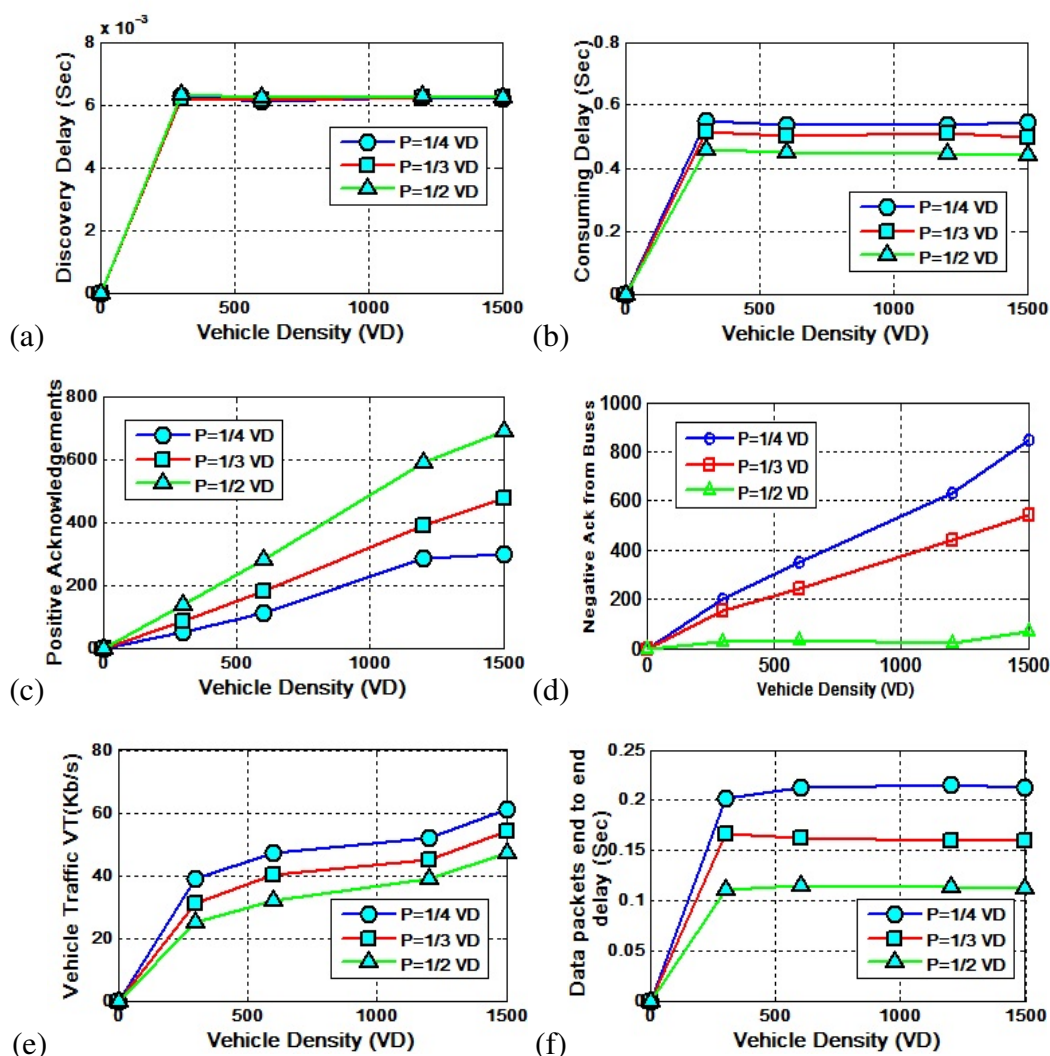


FIGURE 5.3: Evaluation des performances du protocole DCCS-VC dans l'environnement urbain en fonction de la densité des véhicules fournisseurs. (a) Le Délai de Découverte de services (DD). (b) Le Délai de Consommation de services (CD). (c) Les acquittements positifs. (d) Les acquittements négatifs. (e) Le trafic des véhicules (VT). (f) Le délai de bout en bout des paquets de données (E2ED).

Dans la FIGURE 5.3, nous étudions les performances de DCCS-VC en termes de DD, CD, Pos-ACK, Neg-ACK, VT et E2ED quand nous faisons varier la densité des véhicules fournisseurs (P). La FIGURE 5.3-a montre que DCCS-VC a des performances stables et constantes quand la densité de véhicule (VD) et celle des véhicules fournisseurs (P) augmente, cela est justifié par le fait que pour découvrir un service de Cloud dans DCCS-VC, un véhicule consommateur doit envoyer sa requête directement vers le bus le plus proche. Autrement dit, l'opération

de découverte de services ne dépend pas de la densité de véhicules, mais de la présence des bus dans le voisinage. De même, DCCS-VC offre un CD stable et constant quand on augmente VD. Cependant, le CD diminue lorsqu'on augmente P, cela est expliqué par le fait que lorsqu'on augmente le nombre de véhicules fournisseurs la probabilité de trouver et consommer leurs services par les véhicules consommateurs augmente aussi, ce qui permet de diminuer le CD, d'augmenter le Pos-ACK (voir la FIGURE 5.3-c) et de réduire le Neg-ACK (voir la FIGURE 5.3-d).

Les variations du trafic des véhicules (VT) et de délai de bout en bout des paquets de données (E2ED) en fonction de la densité sont illustrés dans les FIGURES 5.3-(e)-(f), respectivement. Nous remarquons que DCCS-VC génère un VT élevé et qui est proportionnel à la densité de véhicules (VD), car plus de densité implique la présence de plus de consommateurs ce qui cause l'augmentation du VT. Nous remarquons aussi que VT diminue quand on augmente P, cela est dû principalement au fait que l'augmentation de P augmente la probabilité des consommateurs à trouver un fournisseur plus proche, ce qui réduit le VT généré et le E2ED (voir la FIGURE 5.3-(f)).

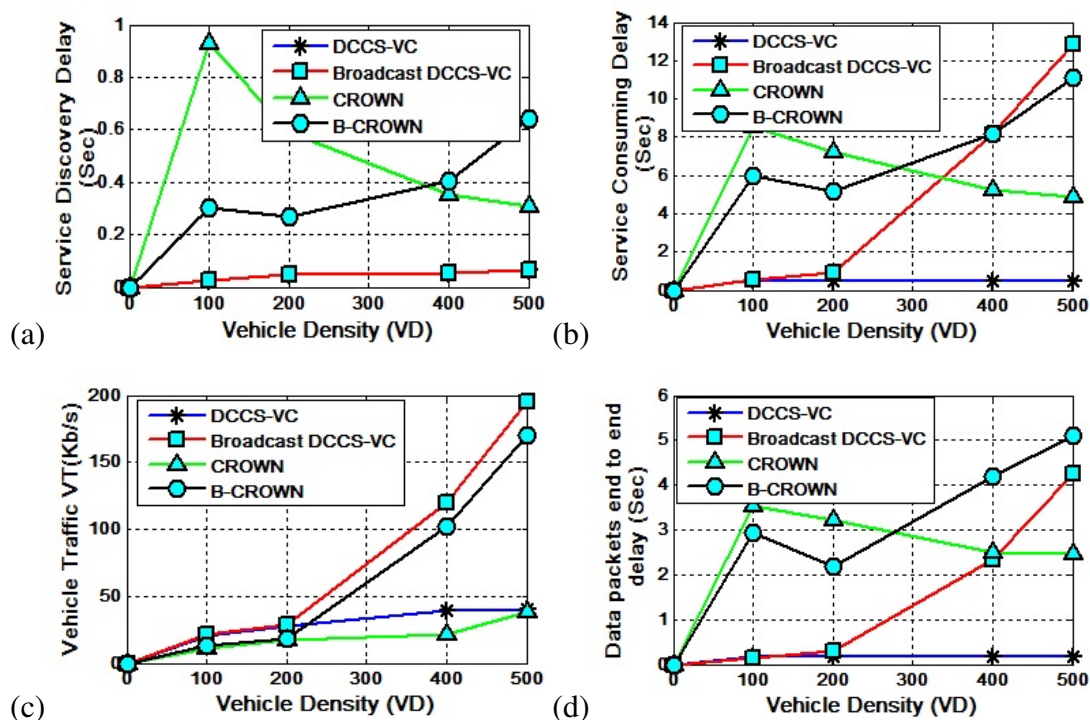


FIGURE 5.4: Evaluation des performances du protocole DCCS-VC dans l'environnement urbain par rapport aux autres protocoles. (a) Le Délai de Découverte de services (DD). (b) Le Délai de Consommation de services (CD). (c) Le Trafic des Véhicules (VT). (d) Le délai de bout en bout des paquets de données (E2ED).

La FIGURE 5.4-a montre que DCCS-VC et Broadcast DCCS-VC fournissent le même DD qui est inférieur à ceux de CROWN et B-CROWN. Cela peut être justifié que pour découvrir un service dans CROWN et B-CROWN, ils utilisent un protocole de routage, tandis que dans DCCS-VC et Broadcast DCCS-VC une communication sans-fil à un seul saut est nécessaire (comme nous l'avons expliqué dans le cas de la FIGURE 5.3-a). Nous remarquons aussi que le DD de CROWN diminue lorsque nous accroissons VD alors que le DD de B-CROWN est proportionnel à la densité VD. Selon le protocole de routage utilisé dans [27], la présence d'un grand nombre de véhicule dans le voisinage permet aux paquets d'être acheminés rapidement. Cependant, le comportement de B-CROWN se base essentiellement sur la diffusion de paquets et surtout dans le cas de la forte densité de véhicules. De même, le CD et le E2ED de DCCS-VC sont inférieurs à tous les autres protocoles (voir la FIGURE 5.4-b et d, respectivement), pour deux raisons : (i) l'architecture en grille du réseau limite la découverte et la consommation de services seulement entre les véhicules dans le même TBP. (ii) Le protocole de routage adopté [83] limite aussi la propagation des paquets seulement entre les bus dans le même TBP. D'une autre part, nous remarquons aussi que Broadcast DCCS-VC fournit le même CD et

Tableau 5.5: Le pourcentage d'enregistrement des fournisseurs avec succès.

	20 bus par TBP	40 bus par TBP	60 bus par TBP
P=1/4 VD	60%	70%	75%
P=1/3 VD	58%	65%	75%
P=1/2 VD	50%	59%	67%

E2ED avec DCCS-VC, pour une densité de véhicules inférieure à 200 véhicules. Après cette valeur, le CD et le E2ED de Broadcast DCCS-VC augmentent exponentiellement à cause de l'adoption d'un mécanisme de diffusion.

De la FIGURE 5.4-c, nous observons que les deux protocoles Broadcast DCCS-VC et B-CROWN atteignent le plus grand VT par rapport aux autres protocoles, ce qui est dû essentiellement au mécanisme de diffusion de paquets utilisé dans ces deux protocoles.

Pour évaluer notre technique de sélection des bus comme répertoire de Cloud, le Tableau 5.5 illustre le pourcentage des véhicules fournisseurs qui ont réussi à enregistrer leurs services dans les bus les plus adéquats respectant leurs préférences. Pour chaque niveau de densité des véhicules fournisseurs, nous remarquons bien que ce pourcentage accroit lorsqu'on augmente le nombre des bus dans chaque TBP. Cela montre que la probabilité qu'un fournisseur trouvera un bus, qui répond à ses préférences, dépend essentiellement de la densité des bus dans le réseau.

5.2 RCS-VC : Renting out and Consuming Services in Vehicular Clouds

Dans cette section, nous exposons notre protocole RCS-VC qui se base sur l'infrastructure du réseau cellulaire LTE-A. Avant de présenter les phases de RCS-VC, nous introduisons tout d'abord le réseau LTE-A.

5.2.1 Long Term Evolution-Advanced (LTE-A)

LTE-Advanced est une norme du réseau mobile de 4^e génération qui a été définie par l'organisme de normalisation 3GPP (3rd Generation Partnership Project) [87]. LTE-A offre un grand débit de communication et il se caractérise par une architecture Full-IP. L'architecture de LTE-A est divisé en deux grandes parties : le réseau d'accès et le cœur ou bien le noyau du réseau (voir la FIGURE 5.5).

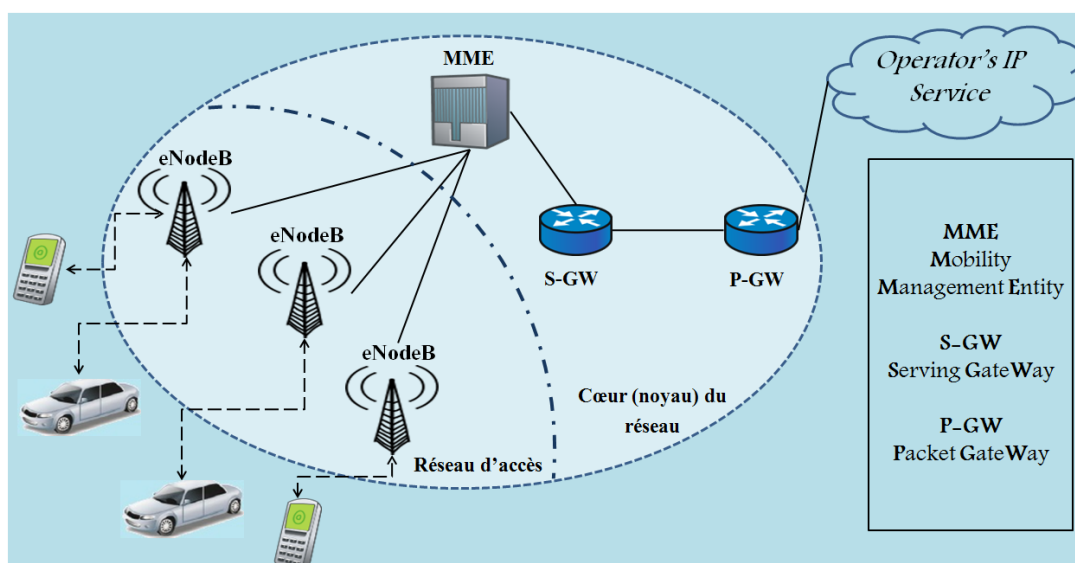


FIGURE 5.5: L'architecture de LTE-A.

Le réseau d'accès est composé des eNodeBs qui se connectent avec les nœuds mobiles (UE, User Equipment) tandis que le noyau du réseau est composé de trois nœuds : Mobility Management Entity (MME) qui est responsable des fonctions de contrôle telles que l'authentification des UEs, la mobilité des nœuds et la gestion de sécurité, Serving Gateway (S-GW) qui achemine les paquets des UEs et Packet Gateway (P-GW) qui permet la communication avec l'architecture IP.

Pour gérer la mobilité des nœuds, LTE-A se base sur le concept de Tracking Area List (TAL) qui permet au MME de suivre le mouvement des UEs. Le MME est connecté à un grand nombre d'eNodeBs (cellules) qui sont groupées en TALs. Lorsqu'un UE s'enregistre au réseau LTE, le MME lui informe par son TAL. Autrement dit, le MME informe le UE par toutes les cellules qu'il couvre. Ensuite, si le UE se retrouve dans une cellule qui n'appartient pas à son TAL, il doit mettre à jour son TAL via le nouveau MME. Lorsque le réseau LTE a besoin de localiser le UE, il lit la liste de dernières cellules où l'UE a effectué un enregistrement [88].

5.2.2 Vue générale sur le protocole RCS-VC

RCS-VC permet aux véhicules clients de découvrir les véhicules fournisseurs afin de consommer leurs services. Dans RCS-VC, nous étendons l'infrastructure de LTE-A pour l'exploiter comme étant un répertoire de Cloud. Pour cela, nous ajoutons un nouveau serveur nommé Vehicular Services Directory (VSD) pour stocker les informations liées aux véhicules fournisseurs et leurs services. Pour faire face à la forte mobilité des véhicules et localiser les véhicules four-

nisseurs, nous connectons le serveur VSD au MME qui est l'entité responsable de la mobilité des nœuds (voir sous-section 5.2.1). La FIGURE 5.6 montre une vue générale sur l'environnement dans lequel le protocole RCS-VC opère.

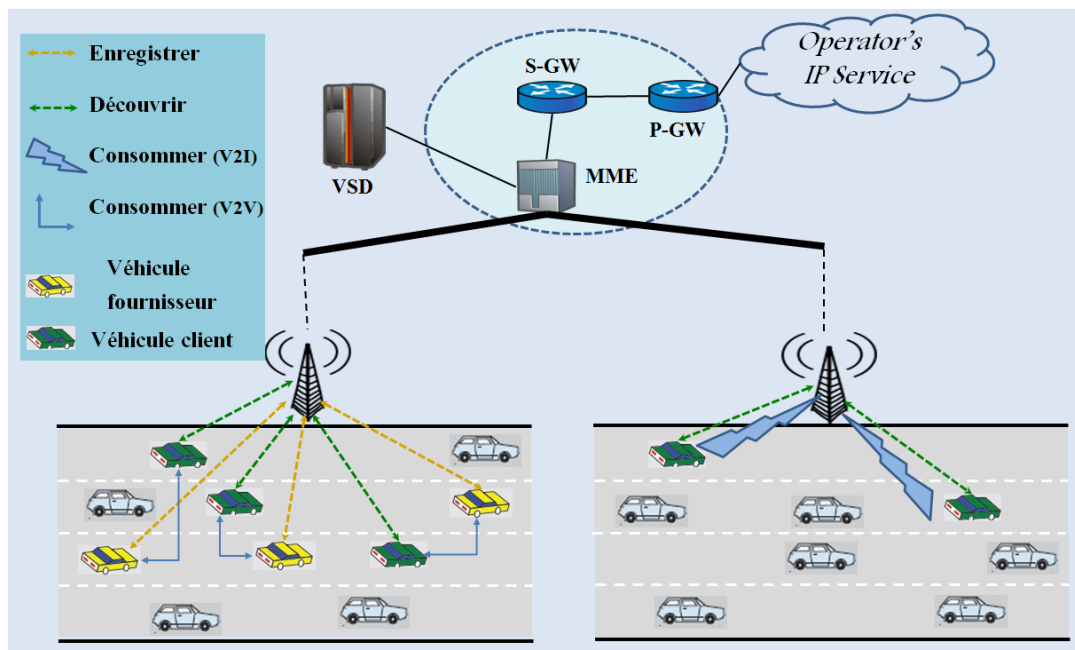


FIGURE 5.6: Une vue générale sur RCS-VC.

5.2.3 Les phases du protocole RCS-VC

La FIGURE 5.7 montre les différentes étapes de fonctionnement du protocole RCS-VC que nous détaillons dans les sous-sections suivantes.

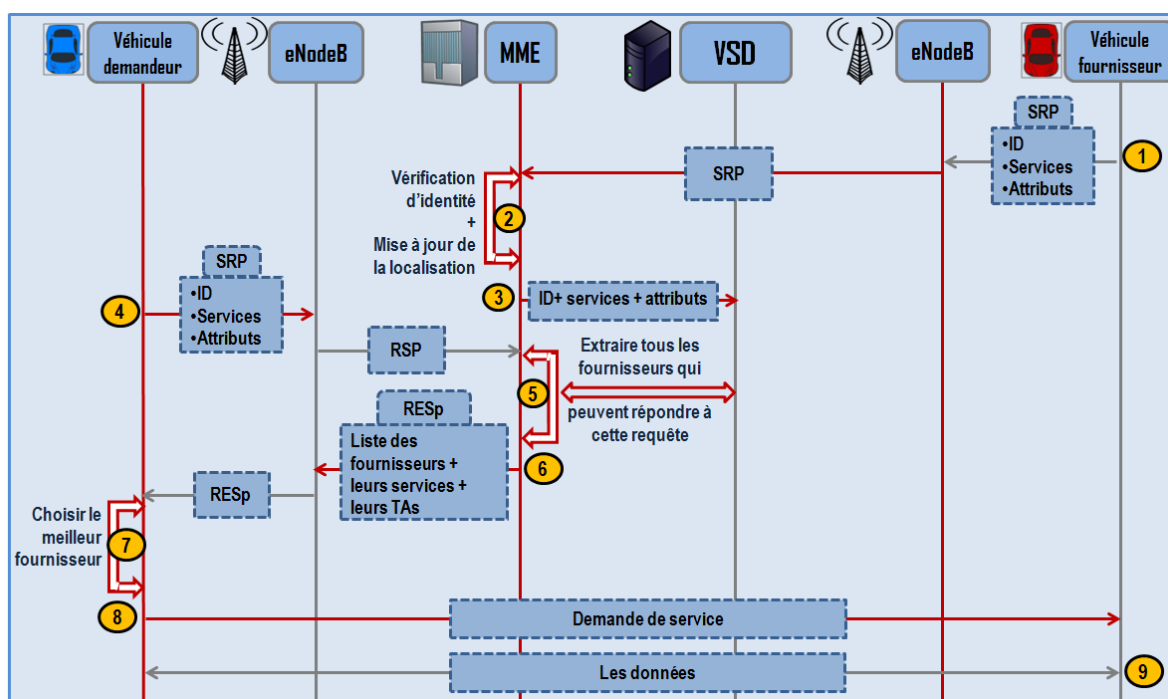


FIGURE 5.7: Les étapes de fonctionnement du protocole RCS-VC.

L'enregistrement de services par un véhicule fournisseur

Comme pour le protocole DCCS-VC, quand un véhicule décide d'agir comme étant un véhicule fournisseur, il doit tout d'abord définir les services qu'il offre ainsi que leurs attributs communs et spécifiques (voir le Tableau 5.1 1).

Après avoir défini ses services, le véhicule fournisseur envoie un paquet nommé Service Registration Packet (SRP) vers son MME à travers son eNodeB (voir étape 1, la FIGURE 5.7). Le MME ensuite vérifie l'identité du véhicule fournisseur avant de stocker ses services offerts dans le serveur VSD (voir étapes 2 et 3, la FIGURE 5.7). Le paquet SRP comprend essentiellement l'identifiant du véhicule fournisseur, les services offerts et leurs exigences (attributs).

La demande de services par un véhicule consommateur

Quand un véhicule a besoin de consommer un service, il formule tout d'abord un paquet Request for Service Packet (RSP) et il l'envoie vers son MME via son eNodeB (voir étape 4, la FIGURE 5.7). Le paquet RSP contient les services désirés et leurs préférences (voir le Tableau 5.2).

Ensuite, le MME cherche tous les véhicules fournisseurs qui peuvent satisfaire cette requête

(voir étape 5, la FIGURE 5.7) et envoie un paquet RESp vers le véhicule demandeur qui contient soit un acquittement négatif ou bien tous les véhicules fournisseurs candidats, leurs offres ainsi que leurs TALs (voir étape 6, la FIGURE 5.7).

La consommation de services

À la réception d'un Ack positif, le véhicule consommateur sélectionne le véhicule fournisseur le plus adéquat selon ses préférences (voir étape 7, la FIGURE 5.7). Par exemple, le véhicule fournisseur qui offre le service désiré avec minimum du coût. Par la suite, le véhicule consommateur envoie un paquet Request for Ressources Packet (RRP) directement vers le véhicule fournisseur sélectionné. Ce paquet comprend toutes les informations nécessaire pour commencer la consommation de services telles que, le mode de paiement, l'ordre des paquets de données, la technique utilisée pour assurer la sécurité et la fiabilité des données (voir étape 8, la FIGURE 5.7).

Le véhicule consommateur envoie le paquet RRP vers le fournisseur soit via une communication V2V ou bien via le réseau LTE-A. Si les deux véhicules (consommateur et fournisseur) partagent le même TA, le consommateur envoie son paquet via V2V. Sinon, il l'envoie à travers le réseau cellulaire LTE-A. Ainsi, les paquets de données seront acheminés de la même façon des paquets RRP (voir étape 9, la FIGURE 5.7).

Nous notons que pour une communication V2V, nous utilisons un mécanisme de diffusion optimisé où seul le véhicule le plus éloigné dans la direction de destination est toujours sélectionné comme le prochain relai.

5.2.4 Simulation et résultats

Dans cette section, nous évaluons les performances de RCS-VC à travers la simulation.

Les paramètres de simulation

Pour évaluer les performances de RCS-VC, nous l'avons simulé à l'aide du simulateur OMNet++ [62] et l'outil SimuLTE [89] qui a été développé afin d'évaluer les performances du réseau LTE sous le simulateur OMNet++.

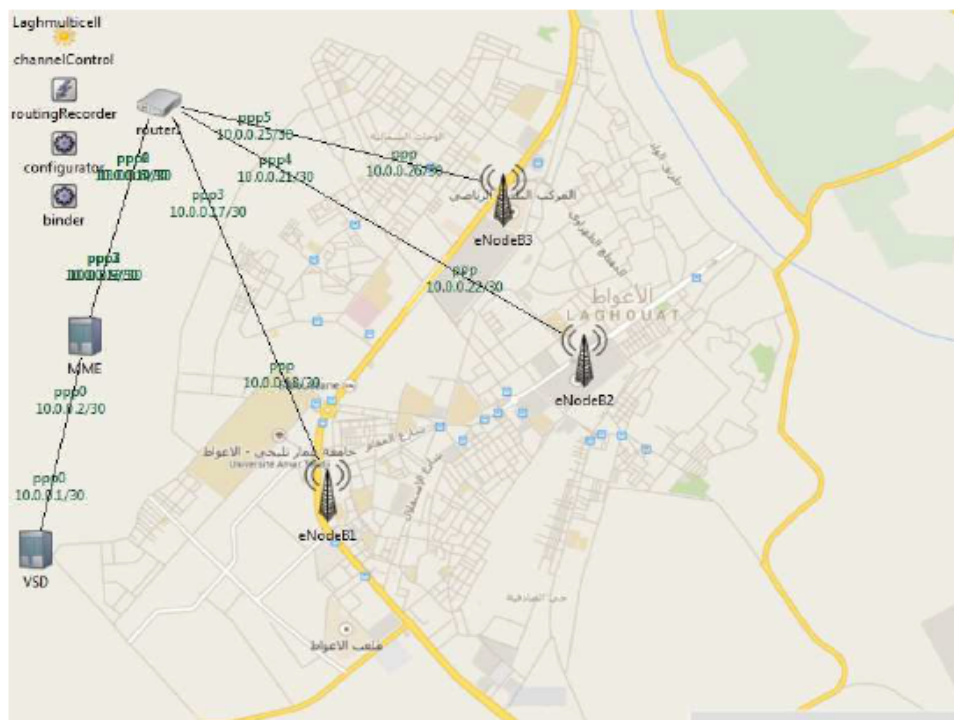


FIGURE 5.8: La localisation des trois eNodeBs dans la ville de Laghouat.

Nous avons simulé RCS-VC en utilisant la carte de la ville de Laghouat qu'on a couvert par trois eNodeB (voir la FIGURE 5.8). Dans notre simulation, nous avons fait varier la densité des véhicules consommateurs (CVD) entre 100 et 500 et nous avons testé trois valeurs de densité des véhicules fournisseurs P : 10, 20 et 40 véhicules. Les principaux paramètres sont illustrés dans le Tableau 5.6.

Tableau 5.6: Les paramètres de simulation.

Paramètres	Valeurs
Temps de simulation	120s
Portée de communication de LTE	1 Km
Portée de communication de 802.11p	500 m
Débit de LTE	100 Mb/s
Débit de 802.11p	27 Mb/s
Vitesse de véhicules	[20, 120] km/h
Densité des véhicules Consommateurs (CVD)	[100, 500] véhicules
Densité des véhicules fournisseurs (P)	10, 20 et 40 véhicules
Taille des paquets Registration, Request, Response	100 octets
Taille des paquets de données	[1-5] Ko
Taille de la file d'attente des véhicules fournisseurs	5 véhicules consommateurs (CV)
Le nombre maximum de services offerts par véhicule fournisseur (PV)	3 services
Le nombre maximum de demande de services par véhicule consommateur (CV)	2 services

Durant les 30 premières secondes de la simulation, les véhicules fournisseurs envoient leurs SRPs vers les MMEs. Chaque fournisseur choisit aléatoirement d'offrir un ou plusieurs services. Ensuite, les véhicules consommateurs commencent à envoyer leurs requêtes vers les MMEs. Enfin, lorsqu'un véhicule consommateur envoie un paquet RRP au fournisseur sélectionné, ce dernier commence à servir le consommateur si sa file d'attente est vide, soit il envoie un acquittement négatif au consommateur si sa file d'attente est pleine. Pour évaluer RCS-VC, nous nous sommes basés sur les métriques de simulation suivantes :

1. Le Délai de Découverte de services (DD, Discovering Delay) : mesure la durée entre l'envoi d'un paquet Request for Service Packet (RSP) et la réception d'un paquet de réponse.
2. Le Délai de Consommation de services (CD, Consuming Delay) : mesure la durée entre l'envoi d'un paquet Request for Resources Packet (RRP) par un véhicule consommateur

vers un véhicule fournisseur et la réception de tous les paquets de données que nous avons limité à trois paquets.

3. Les acquittements positifs et négatifs (Pos-ACK et Neg-ACK) : le nombre d'acquittements positifs et négatifs.
4. La durée d'attente dans la file d'attente (QD, Queing Delay) : la durée moyenne pendant laquelle la demande d'un véhicule consommateur reste dans la file du véhicule fournisseur. De plus, nous comparons RCS-VC avec les deux protocoles CROWN et B-CROWN [27] en termes de DD et CD.

Résultats et discussions

Dans la FIGURE 5.9, nous étudions les performances de RCS-VC en termes de DD, CD et QD quand on varie la densité des véhicules fournisseurs (P). La FIGURE 5.9-a montre que DD est proportionnel au CVD ce qui est dû essentiellement à l'augmentation de nombre de véhicules consommateurs (CVD). Cependant, le DD diminue lorsque P accroit. La présence de plusieurs véhicules fournisseurs implique que les véhicules consommateurs auront plus de chance à trouver les services requis, ce qui permet de réduire le DD, d'augmenter le nombre des Pos-ACK (voir la FIGURE 5.9-d) et de diminuer le nombre des Neg-ACK (voir la FIGURE 5.9-e).

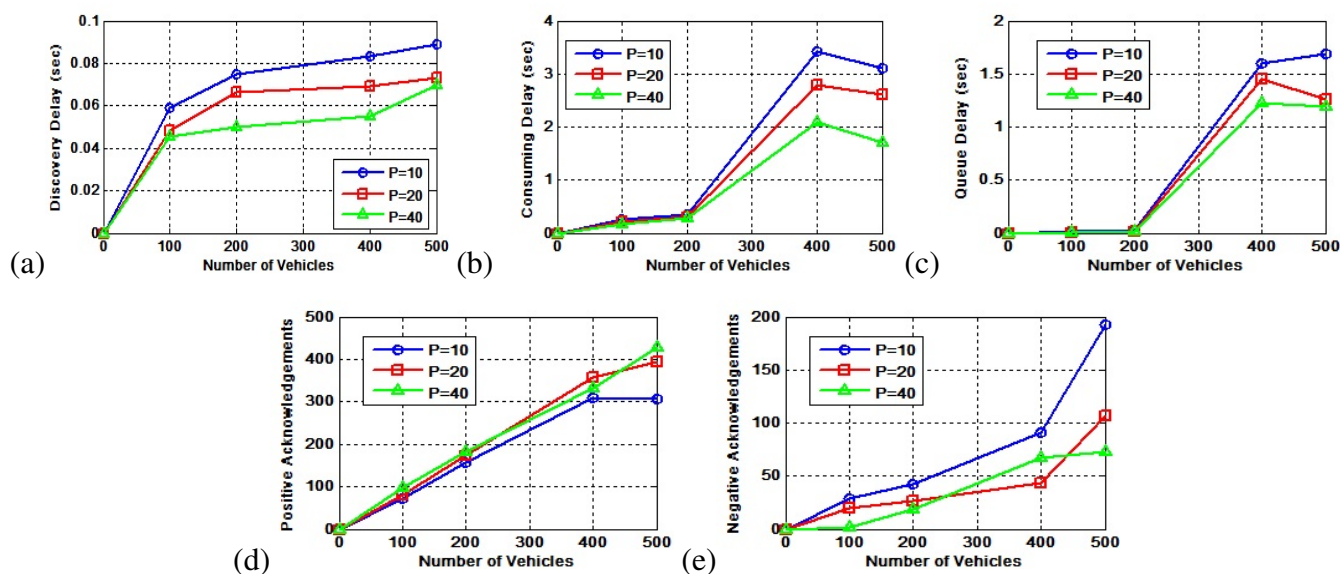


FIGURE 5.9: Evaluation de performance du protocole RCS-VC. (a) Le Délai de Découverte de services (DD). (b) Le Délai de Consommation de services (CD). (c) La durée d’attente dans la file d’attente (QD). (d) Le nombre des acquittements positifs. (e) Le nombre des acquittements négatifs.

Pour CD, RCS-VC offre des performances stable jusqu’au niveau de densité CVD=200 véhicules. Ensuite, le CD augmente à de grandes valeurs (voir la FIGURE 5.9-b), cela est causé par l’utilisation de la communication V2V durant la dernière phase du RCS-VC. Donc, dans un réseau dense (plus de 200 véhicules), un véhicule consommateur a une grande probabilité d’atteindre le véhicule fournisseur et de communiquer avec lui via une communication V2V. Par conséquent, la communication V2V augmente le CD par rapport à la communication via le réseau LTE qui offre un grand débit.

Nous remarquons aussi que le CD affecte fortement le QD (voir la FIGURE 5.9-c). Pour une densité CVD inférieur à 200 véhicules, un véhicule consommateur au maximum reste 0.02s dans la file d’attente. Mais pour une densité supérieure à 200 véhicules, le QD atteint des grandes valeurs ce qui dû essentiellement à l’augmentation du CD.

Les FIGURES 5.10-a et b montrent que le DD et le CD du protocole RCS-VC sont inférieures à ceux des deux protocoles CROWN et B-CROWN, ce qui est justifié par le grand débit de communication du réseau LTE-A (100 Mb/s) par rapport à celui de CROWN et B-CROWN (10 Mb/s).

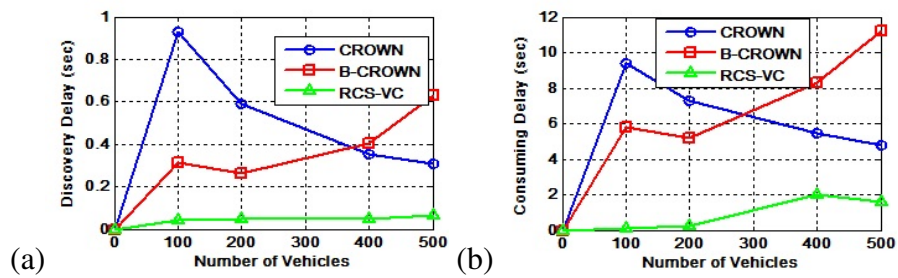


FIGURE 5.10: Evaluation de performance du protocole RCS-VC par rapport aux autres protocoles. (a) Le Délai de Découverte de services (DD). (b) Le Délai de Consommation de services (CD).

Enfin, nous déduisons que notre protocole RCS-VC offre des performances stables même avec l'augmentation du nombre de véhicules consommateurs (CVD). Cependant, les performances de RCS-VC durant la dernière phase dépendent fortement du mécanisme de communication utilisé (V2V ou LTE-A). Nous pouvons aussi déduire que RCS-VC est plus performant que les protocoles CROWN et B-CROWN en termes de DD et CD, grâce au débit élevé de LTE-A par rapport à celui de VANet.

5.3 Conclusion

Dans ce chapitre, nous avons présenté nos deux protocoles de découverte et de consommation de services dans les réseaux véhiculaires, dans lequel les véhicules peuvent agir en tant que fournisseurs de Cloud et offrir leurs services à d'autres véhicules.

Nous avons aussi décrit comment les bus public et l'infrastructure du LTE-A peuvent être utilisés comme étant des répertoires de Cloud afin de faciliter la découverte et la consommation de services.

Nous avons validé nos protocoles à travers plusieurs scénarios de simulation, en les comparant avec les protocoles CROWN et B-CROWN. Les résultats de simulation ont montré que nos protocoles offrent des performances stables même avec l'augmentation du nombre de véhicules consommateurs. De plus, ils sont plus performants que les protocoles CROWN et B-CROWN en termes de délais de découverte et de consommation de services.

Chapitre 6

Nos modèles de sélection de services dans le Cloud véhiculaire

DANS le Cloud véhiculaire, il y a des paramètres critiques de qualité de service (QoS, Quality of Service) à prendre en compte lors du traitement d'une demande de service par un fournisseur. Par exemple, le temps de réponse et la fiabilité de consommation de service. Cependant, plusieurs véhicules fournisseurs offrent des services similaires mais avec des qualités de services différentes. Il devient alors difficile aux consommateurs de choisir les fournisseurs qui conviennent le mieux à leurs besoins.

Dans ce contexte, nous avons proposé deux modèles de sélection de services dans le Cloud véhiculaire prenant en considération les exigences de chaque acteur (consommateur et fournisseur) afin de déterminer le meilleur fournisseur en termes de qualité de service [90, 91]. Le premier modèle se base sur la logique floue pour permettre à chaque bus public de sélectionner le véhicule fournisseur le plus adéquat dans notre protocole DCCS-VC, au lieu d'envoyer aux consommateurs la liste de tous les véhicules fournisseurs [91]. Tandis que le deuxième modèle se base sur la théorie de jeux, nommé GSS-VC pour Game theory-based approach for Service Selection in Vehicular Cloud [90]. GSS-VC est un modèle générique qui permet aux véhicules consommateurs de sélectionner les meilleurs fournisseurs sur la route.

Dans ce chapitre, nous présentons nos modèles de sélection de services. Mais avant de présenter chaque modèle, nous commençons tout d'abord par donner une vue générale sur la logique floue et la théorie de jeux et leurs concepts de base.

6.1 La sélection de service dans DCCS-VC fondée sur la logique floue

Dans cette section, nous présentons une vue générale de la logique floue. Par la suite, nous décrivons notre modèle de sélection de services dans le protocole DCCS-VC qui se base sur cette logique.

6.1.1 Introduction à la logique floue

La logique floue est une extension de la logique booléenne créée par Lotfi Zadeh qui est fondée sur la théorie mathématique des ensembles flous [92]. Elle remplace la valeur de vérité d'une proposition à choisir dans {vrai, faux} par un degré de vérité, à choisir, généralement, dans l'intervalle $[0, 1]$.

La logique floue permet la modélisation des imperfections des données et se rapproche de la flexibilité du raisonnement humain. Un des intérêts de la logique floue pour formaliser le raisonnement humain est que les règles sont énoncées en langage naturel telles que : *"Si le feu est rouge et si ma vitesse est élevée et si le feu est proche alors je freine fort"* ou bien *"Si le feu est rouge et si ma vitesse est faible et si le feu est loin alors je maintiens ma vitesse"*.

La logique floue est aussi appelée logique linguistique car elle se base sur des valeurs de vérité qui sont des mots du langage courant : élevé, loin, si loin, près de, grand, petit. ...

Les ensembles flous

En logique booléenne, l'appartenance d'un élément à un ensemble est booléenne. Les ensembles flous permettent en revanche, de déterminer le degré d'appartenance d'un élément à un ensemble. Un ensemble flou A d'un univers du discours U est caractérisé par une fonction d'appartenance $\mu_A : U \rightarrow [0, 1]$ où $\mu_A(x)$ est le degré d'appartenance de l'élément x à A (voir la FIGURE 6.1) [92].

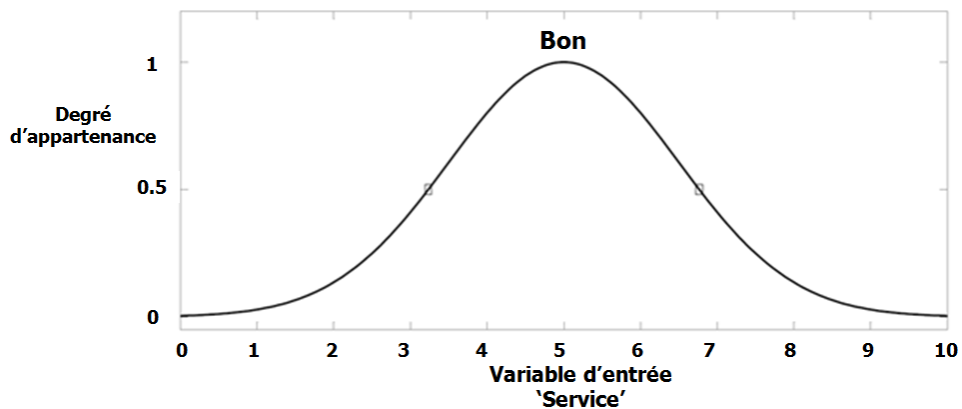


FIGURE 6.1: Fonction d'appartenance caractérisant le sous-ensemble 'bon' de la qualité du service.

La fonction d'appartenance permet de définir des systèmes flous en langage naturel en faisant le lien entre logique floue et variable linguistique.

Variables linguistiques

En logique floue, les systèmes flous sont représentés par des variables linguistiques. Une variable linguistique est une variable dont les valeurs sont des mots ou des phrases utilisées couramment dans une langue naturelle ou un langage artificiel [92]. Par exemple, soit V une variable (qualité du service dans un restaurant, qualité de la nourriture, montant du pourboire, etc.), X la plage de valeurs de la variable (par exemple, entre 0 et 30 DA pour le pourboire) et T_V un ensemble flou fini ou infini. Une variable linguistique correspond au triplet (V, X, T_V) . La FIGURE 6.2 montre un exemple de la variable linguistique qualité de service avec $V = \text{qualité de service}$, $X = [0, 10]$ et $T_V = \{\text{mauvais}, \text{bon}, \text{excellent}\}$.

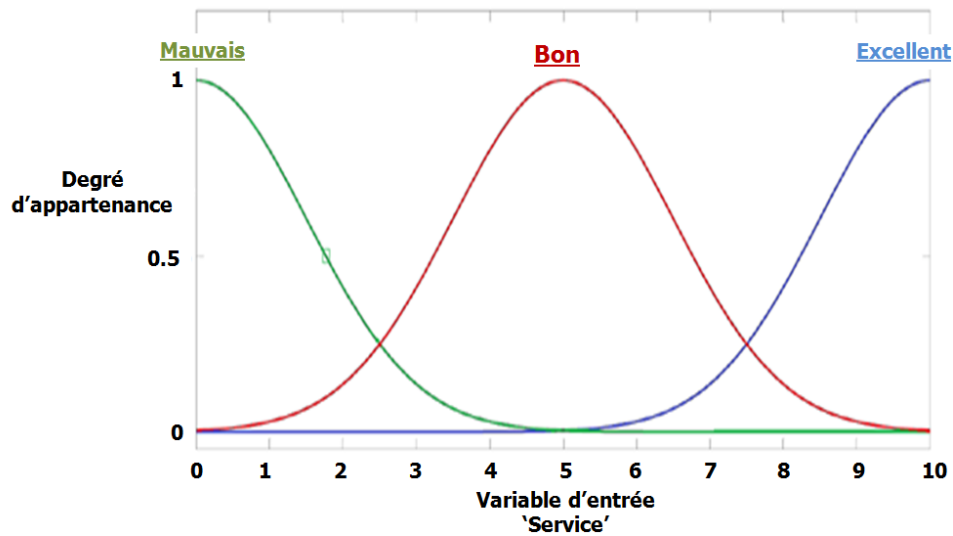


FIGURE 6.2: Variable linguistique qualité de service.

Système flou

L'objectif d'un système flou est de transformer les données d'entrée en résultats à partir de l'évaluation d'un ensemble des règles. De toute façon, un système flou (voir la FIGURE 6.3) comprend trois étapes : a) Fuzzification, b) Inférence et c) Défuzzification.

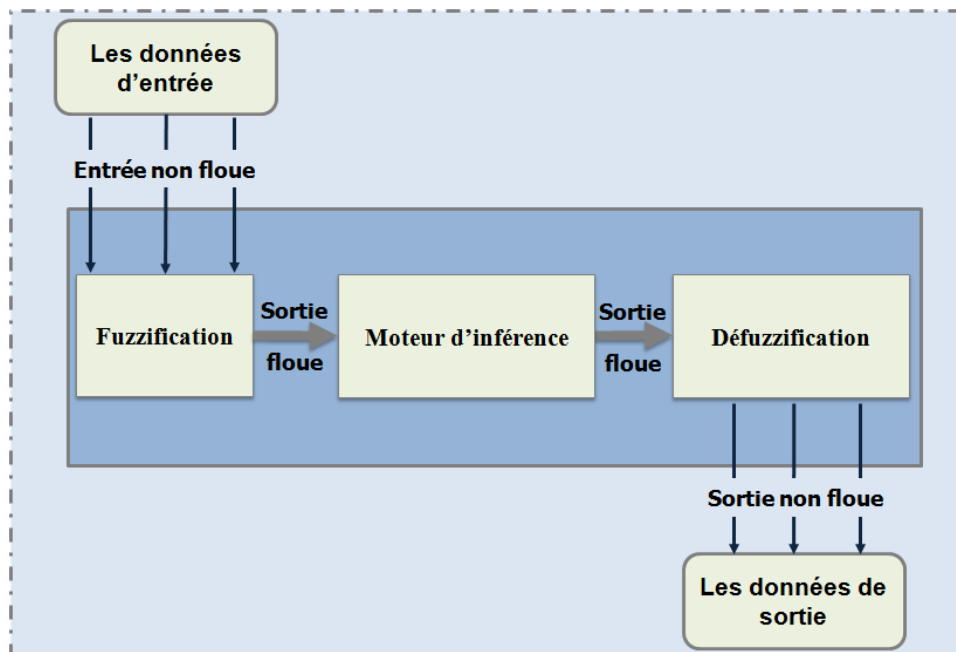


FIGURE 6.3: Les étapes d'un système flou.

- a) L'étape de fuzzification consiste à caractériser les variables linguistiques utilisées dans le système flou. Il s'agit donc d'une transformation des entrées réelles en une partie floue définie sur un espace de représentation lié à l'entrée (ensemble flou). Durant l'étape de la fuzzification, chaque variable d'entrée et de sortie est associée à des ensembles flous.
- b) La deuxième étape est le moteur d'inférence, qui est un mécanisme permettant de condenser l'information d'un système au travers d'un ensemble de règles défini pour la représentation d'un problème quelconque. Les règles sont exprimées en langage naturel en utilisant les variables linguistiques. Une règle floue a cette forme : Si $x \in A$ et $y \in B$ alors $z \in C$, avec A , B et C des ensembles flous. Par exemple, *si le service est mauvais ou la nourriture est exécration alors le pourboire est faible* » ou « *si le service est excellent ou la nourriture est délicieuse alors le pourboire est élevé* ».
- c) La troisième étape est la défuzzification, cette opération est l'inverse de la fuzzification et permet de transformer les sorties floues de l'inférence en une valeur non floue comme réponse finale du système. Il existe plusieurs méthodes de défuzzification [93] : méthodes de maximum, somme-prod, moyenne pondérée, moyenne des maximums. Cependant, la méthode la plus utilisée est celle de la détermination du centre de gravité de l'ensemble flou.

Application de la logique floue

La logique floue a été utilisée dans plusieurs domaines d'applications [94, 95]. Nous pouvons citer :

- Commande floue : un des champs d'application les plus actifs qui permet d'obtenir une loi de commande souvent efficace, sans devoir faire appel à des développements théoriques importants. Nous citons comme exemples : l'automatisation de métro de Sandai en 1988 et d'une usine de fabrication de papier au Portugal en 1992, dans le domaine de la robotique, les systèmes de ventilation et de climatisation, etc.
- Classification et reconnaissance de forme : elle consiste à regrouper des objets en catégories les plus homogènes possibles (contenant des objets similaires). La classification intervient dans plusieurs domaines d'applications tels que : la reconnaissance vocale, l'analyse d'images (médicales, radar, télédétection), la reconnaissance de cibles (domaine militaire).
- Aide à la décision multicritères : il s'agit de choisir la meilleure solution ou la solution optimale parmi tout un ensemble de solutions, par exemple la gestion de préférence (plus grand, plus rapide, moins cher, etc).

6.1.2 Description du modèle flou de sélection de service dans DCCS-VC

Dans cette sous-section, nous décrivons notre système flou que nous avons défini pour sélectionner le meilleur service (fournisseur) dans Cloud véhiculaire.

Les données d'entrée et de sortie et leurs représentations floues

Suite à une demande d'un véhicule consommateur, l'objectif de notre modèle est d'évaluer l'éligibilité du service de chaque fournisseur d'être sélectionné par le consommateur. Pour cela, nous considérons les critères de qualité suivants comme données d'entrée :

1. Débit : représente le nombre de paquets de données délivrés par le véhicule fournisseur dans un intervalle de temps (en seconds).
2. Réputation (q_{Rep}) : dépend de l'historique d'offre des services aux véhicules consommateurs. Ce critère est déterminé par les bus publics en utilisant la formule suivante :

$$q_{Rep} = \frac{\sum_{i=1}^n R_i}{n}, R_i \in [0, 1] \quad (6.1)$$

Avec R_i représente l'opinion des véhicules consommateurs sur le service s du fournisseur et n est le nombre total des véhicules consommateurs.

3. Le rapport d'offre de service avec succès (q_{SER}) : c'est la probabilité que le consommateur consomme correctement son service. Ce taux est calculé par :

$$q_{SER} = \frac{N(s)}{K} \quad (6.2)$$

Avec $N(s)$ est le nombre de fois que le s du VF a été consommé correctement et K est le nombre de fois que le service s a été offert.

4. La durée d'exécution (q_{ED}) : le délai en secondes de consommation de service qui est la somme du délai de traitement et le délai de transmission des paquets de données du service s :

$$q_{ED} = T_{Process}(s) + T_{Trans}(s) \quad (6.3)$$

5. Le coût d'exécution (q_{EP}) : c'est le prix de service proposé par le fournisseur.

Enfin, le système flou calcule le degré de sélection du service comme donnée de sortie. La FIGURE 6.4 montre les fonctions d'appartenance des variables linguistiques utilisées.

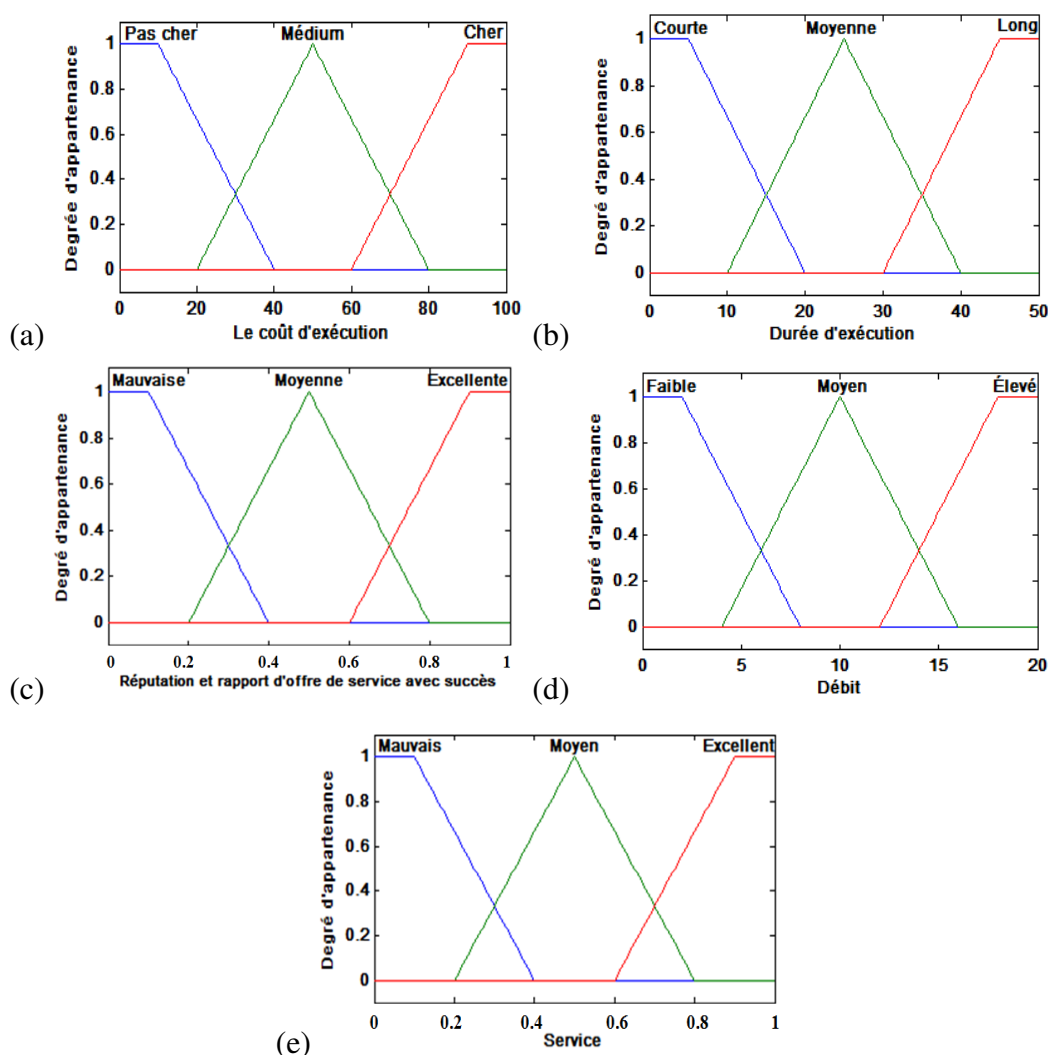


FIGURE 6.4: Les fonctions d'appartenance des variables linguistiques : (a) le coût d'exécution, (b) la durée d'exécution, (c) réputation et rapport d'exécution avec succès, (d) débit, (e) service

Définition des règles floues

Après avoir défini les ensembles flous de chaque variable, nous avons construit la base des règles sur lesquelles nous nous sommes basés afin d'évaluer chaque service. Il est bien clair que le nombre de règles dépend du nombre de variables (d'entrée et de sortie) [96]. Ces règles floues permettent de représenter les relations entre les variables d'entrée et de sortie en utilisant les opérateurs *et*, *ou*, *not*, *etc.*, de la façon suivante :

- **Si** (Débit est Élevé) **et** (Réputation est Excellente) **et** (Rapport d'offre avec succès est Excellent) **et** (Durée d'exécution est courte) **et** (cout d'exécution est Pas cher) **Alors** (service est Excellent).

Dans ce système flou, nous avons 5 variables linguistiques d'entrée et chacune possède trois ensembles flous. Par conséquent, nous avons défini 243 règles floues, $3^5=243$, (voir le Tableau 6.1).

Tableau 6.1: Ensemble des règles.

	Débit	Réputation	Rapport d'offre avec succès	Durée	Coût	Service
1	Faible	Mauvaise	Mauvais	Longue	Pas cher	Mauvais
2	Faible	Mauvaise	Mauvais	Longue	Moyen	Mauvais
3	Faible	Mauvaise	Mauvais	Longue	Cher	Mauvais
4	Faible	Mauvaise	Mauvais	Moyenne	Pas cher	Mauvais
5	Faible	Mauvaise	Mauvais	Moyenne	Moyen	Mauvais
..
..
..
50	Faible	Excellente	Excellent	Moyenne	Moyen	Moyen
51	Faible	Excellente	Excellent	Moyenne	Pas cher	Moyen
52	Faible	Excellente	Excellent	Moyenne	Cher	Mauvais
..
241	Moyen	Excellente	Excellent	Courte	Pas cher	Moyen
242	Élevé	Mauvaise	Excellent	Courte	Pas cher	Excellent
243	Élevé	Excellente	Excellent	Courte	Pas cher	Excellent

Par la suite, nous avons simplifié l'ensemble des règles en favorisant certaines variables linguistiques par rapport à d'autres, par exemple : si la réputation est mauvaise, le service est mauvais quelque soit l'état des autres variables. Sur cette base, nous avons réduit le nombre des règles précédentes en seulement 42 règles (voir le Tableau 6.2).

Tableau 6.2: Ensemble de règles obtenues.

	Débit	Réputation	Rapport d'offre avec succès	Durées	Coût	Service
1		Mauvaise	Mauvais			Mauvais
2		Mauvaise	Moyen			Mauvais
3		Mauvaise	Excellent			Mauvais
4		Moyenne	Mauvais		Pas cher	Mauvais
5		Moyenne	Mauvais		Moyen	Mauvais
6		Moyenne	Mauvais		Cher	Mauvais
7		Moyenne	Moyen		Pas cher	Moyen
8		Moyenne	Moyen		Moyen	Mauvais
9		Moyenne	Moyen		Cher	Mauvais
10		Moyenne	Excellent		Pas cher	Moyen
11		Moyenne	Excellent		Moyen	Moyen
12		Excellente	Mauvais		Pas cher	Moyen
13		Excellente	Mauvais		Moyen	Mauvais
14		Excellente	Mauvais		Cher	Mauvais
15		Excellente	Moyen		Pas cher	Excellent
16		Excellente	Moyen		Moyen	Moyen
17		Excellente	Moyen		Cher	Mauvais
18		Excellente	Excellent		Pas cher	Excellent
19		Excellente	Excellent		Moyen	Excellent
20		Excellente	Excellent		Cher	Moyen
21		Moyenne	Excellent		Cher	Mauvais
22	Élevé			Longue		Mauvais
23	Moyen			Longue		Mauvais
24	Faible			Longue		Mauvais
25	Faible			Courte	Pas cher	Excellent
26	Faible			Courte	Moyen	Excellent
27	Faible			Courte	Cher	Mauvais
28	Faible			Moyenne	Pas cher	Moyen
29	Faible			Moyenne	Moyen	Moyen
30	Faible			Moyenne	Cher	Mauvais
31	Moyen			Courte	Pas cher	Excellent
32	Moyen			Courte	Moyen	Excellent
33	Moyen			Courte	Cher	Mauvais
34	Moyen			Moyenne	Pas cher	Moyen
35	Moyen			Moyenne	Pas cher	Moyen
36	Moyen			Moyenne	Cher	Mauvais
37	Élevé			Courte	Pas cher	Excellent
38	Élevé			Courte	Moyen	Excellent
39	Élevé			Courte	Cher	Moyen
40	Élevé			Moyenne	Pas cher	Moyen
41	Élevé			Moyenne	Moyen	Moyen
42	Élevé			Moyenne	Cher	Mauvais

Défuzzification

La défuzzification permet d'avoir un résultat numérique non flou à partir de la sortie de l'inférence (ensemble flou). Pour ce faire, nous nous sommes basés sur la méthode de défuzzification du centre de gravité de cet ensemble. Par exemple, si le résultat obtenu du service

est moyen (voir la FIGURE 6.5), nous calculons le centre de gravité de la forme représentant l'ensemble flou moyen et nous obtiendrons la valeur 0.5.

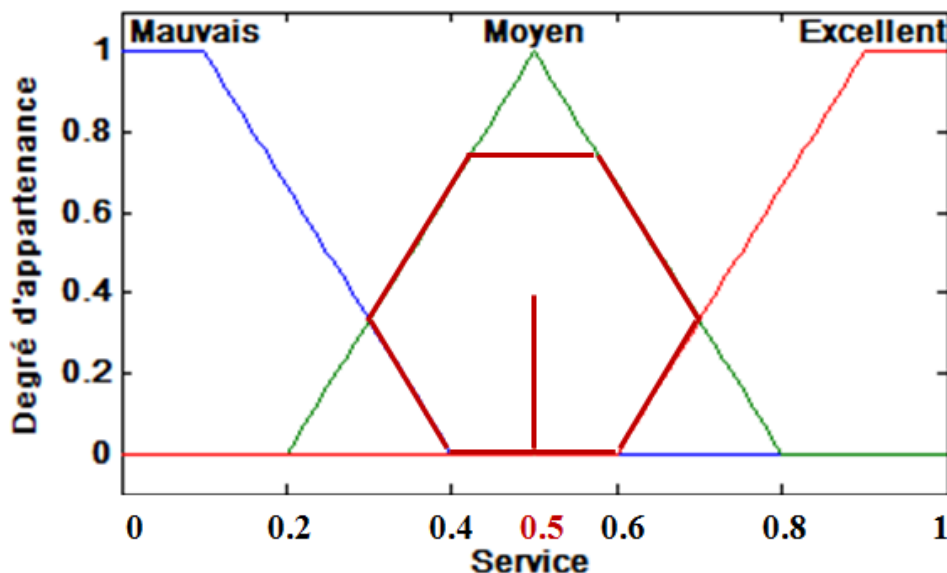


FIGURE 6.5: La défuzzification de l'ensemble flou Moyen.

Simulation et résultats

Dans cette section, nous évaluons les performances de notre modèle de sélection par simulation. Les paramètres de simulation Pour évaluer notre modèle de sélection, nous avons utilisé les mêmes simulateurs ainsi que les mêmes paramètres que nous avons utilisés pour valider le protocole DCCS-VC (voir sous-section les paramètres de simulation du protocole DCCS-VC dans chapitre 5).

Nous avons comparé notre modèle de sélection avec la technique de sur-classement SAW qui, pour chaque service offert, calcule la moyenne des critères de qualité multipliés par leurs coefficients (poids) affectés par le consommateur. Pour cela, nous nous sommes concentrés sur trois critères à savoir : la réputation, le coût et la durée d'exécution. De plus, nous avons simulé trois scénarios différents dans chacun nous favorisons un critère par rapport aux deux autres. Autrement dit, le consommateur lui affecte le poids le plus fort par rapport aux deux autres critères (voir le Tableau 6.3).

Tableau 6.3: Scénarios de simulation.

	Réputation	Coût d'exécution	Durée d'exécution
Scénario 1	Poids faible	Poids fort	Poids faible
Scénario 2	Poids fort	Poids faible	Poids faible
Scénario 3	Poids faible	Poids faible	Poids fort

Résultats et discussions Le Tableau 6.4 montre les fournisseurs sélectionnés et leurs critères de qualité dans chaque scénario. Ces résultats sont obtenus après avoir sélectionné aléatoirement un bus public dans lequel nous faisons varier la densité des véhicules fournisseurs (VF).

Dans les scénarios 1 et 3, nous remarquons que pour une densité de 5 et 10 VFs, la technique de la logique floue et la technique SAW sélectionnent le même fournisseur, alors que pour une densité de 15 et 20 VFs, SAW est meilleure en terme de coût et de durée, respectivement, dans les scénarios 1 et 3, tandis que notre modèle est meilleur en terme de réputation et durée d'exécution dans le scénario 1 et en termes de réputation et coût d'exécution dans le scénario 3. Par contre, dans le scénario 2, les deux techniques sélectionnent le même fournisseur pour tous les niveaux de densité des VFs.

Ces résultats peuvent être justifiés par le fait que dans SAW le consommateur doit favoriser certains critères par rapport aux autres. Par conséquence, cette technique sélectionne les fournisseurs dont seulement les valeurs des critères favorisés sont meilleures. Par contre, la logique floue sélectionne les fournisseurs de services dont les valeurs de la plupart de leurs critères sont meilleures.

Tableau 6.4: Résultats de simulation

Scénario	Méthode	Densité des VFs	L'ID de VF sélectionné	Réputation	Durée	Coût
Scénario 1	Logique floue	5	6	10	23	14
	SAW		6	10	23	14
	Logique floue	10	10	7	27	10
	SAW		10	7	27	10
	Logique floue	15	16	10	24	12
	SAW		10	7	27	10
	Logique floue	20	16	10	24	12
	SAW		10	7	27	10
Scénario 2	Logique floue	5	6	10	23	14
	SAW		6	10	23	14
	Logique floue	10	10	7	27	10
	SAW		10	7	27	10
	Logique floue	15	16	10	24	12
	SAW		16	10	24	12
	Logique floue	20	16	10	24	12
	SAW		16	10	24	12
Scénario 3	Logique floue	5	6	10	23	14
	SAW		6	10	23	14
	Logique floue	10	10	7	27	10
	SAW		11	4	21	17
	Logique floue	15	16	10	24	12
	SAW		16	10	24	12
	Logique floue	20	16	10	24	12
	SAW		20	7	21	19

6.2 La sélection de service fondée sur la théorie des jeux

Dans cette section, nous présentons une introduction sur la théorie des jeux avant d'exposer notre modèle de sélection des véhicules fournisseurs basé sur cette théorie, baptisé, GSS-VC pour Game theory-based approach for Service Selection in Vehicular Cloud.

6.2.1 Introduction à la théorie des jeux

La théorie des jeux est un outil mathématique qui permet d'analyser formellement des situations dans lesquelles un ensemble d'agents "*joueurs*" interagissent. L'objectif de la théorie des jeux est de déterminer l'action optimale pour chacun des joueurs, de prédire l'équilibre du jeu et de trouver comment peut-on aboutir à une situation optimale. Le résultat du jeu dépend alors conjointement des actions prises par chaque joueur [97, 98, 99].

Formellement, un jeu est défini par :

- Un ensemble $N = \{1, \dots, n\}$ de joueurs,
- Pour chaque joueur i un ensemble de stratégies $S_i = \{s_1, \dots, s_n\}$,

- Pour chaque joueur i une fonction de gain $u_i : S_1 \dots S_n \rightarrow \mathbb{R}$, qui à chaque ensemble de stratégies associe les gains du joueur i .
- Nous notons s un profil de stratégies $\{s_1, \dots, s_n\}$ où $\forall i, s_i \in S_i$, et s_{-i} le profil s des stratégies autres que celles du joueur i : $s_{-i} = \{s_1, \dots, s_{i-1}, s_{i+1}, \dots, s_n\}$.

Exemples de jeux

La théorie des jeux a été appliquée pour résoudre plusieurs types de jeux [100, 101], parmi lesquels l'on peut citer :

- Le jeu de dilemme du prisonnier : deux suspects d'un crime majeur sont détenus dans deux cellules séparées. Si les deux suspects se taisent, ils seront chacun condamnés à un an de prison. Si seulement l'un dénonce l'autre, il sera libéré et utilisé en tant que témoin contre l'autre qui sera condamné à 10 ans de prison. Enfin, si les deux dénoncent, ils passeront chacun 5 ans en prison.
- Le jeu de Pierre feuille ciseaux : c'est un jeu d'enfant à deux joueurs qui doivent choisir simultanément une action : pierre, feuille ou ciseaux. Il y a un cycle entre ces trois symboles : la pierre casse les ciseaux, les ciseaux coupent la feuille et la feuille recouvre la pierre. Si les deux joueurs choisissent le même symbole, alors ils font match nul. Sinon, le joueur ayant le symbole le plus fort gagne et l'autre perd une.
- Le jeu du challenger un monopole : un joueur en position de monopole, le "titulaire", est confronté à la possible entrée d'un challenger (le challenger peut, par exemple, être une firme désirant entrer sur un marché régi par un monopole ou un homme politique désirant prendre le contrôle d'un parti. Le challenger peut entrer ou non. S'il entre, le "titulaire" peut accepter ou se battre.
- Le jeu d'allocation de coûts : trois villes voisines (A, B et C) sont en contrat avec une société pour réaliser des adductions d'eau. Le projet revient à 10 (millions d'euros) pour chaque municipalité prise séparément. Pour des raisons géographiques, le constructeur propose des coûts (réduits) de respectivement 16, 17 et 18 pour des contrats communs entre A et B, A et C, et B et C. Le contrat impliquant les trois villes a un coût de 24. Comment les coûts devraient-ils être répartis entre les trois villes ?

Types des jeux

En théorie de jeux, les jeux peuvent être classés en fonction de leurs approches de résolution [97, 98, 99]. Les classes les plus courantes sont :

- **Jeux coopératifs et jeux non coopératifs** : un jeu coopératif permet d'étudier et d'ana-

lyser les situations où les agents peuvent se regrouper en coalitions (sous-ensembles de joueurs) afin de coopérer. Ce type de jeu couvre plusieurs situations réelles, notamment dans les domaines de l'économie et de la politique : allocation et partage de coûts, distribution de gains, exploitation de ressources communes, etc. En revanche, un jeu non coopératif modélise les interactions stratégiques entre différents joueurs qui ne cherchent pas à se coordonner. Les jeux de dilemme du prisonnier, de pierre feuille ciseaux et du challenger un monopole sont des jeux non coopératifs.

- **Jeux simultanés et jeux séquentiels** : dans ce cas, les joueurs décident de leurs actions soit au même temps ou conditionnellement à ce qu'ont joué les autres joueurs. Par exemple, le jeu de dilemme du prisonnier et le jeu pierre-feuille-ciseaux sont des jeux simultanés, alors que le jeu d'échecs et le jeu de monopole et challenger sont des jeux séquentiels.
- **Jeux à somme nulle et jeux à somme non nulle** : jeu à somme nulle est un jeu à deux joueurs dans lequel le gain de l'un des deux joueurs représente exactement le coût de l'autre joueur. Le jeu pierre-feuille-ciseaux est un jeu à somme nulle. Le dilemme du prisonnier n'est pas un jeu à somme nulle (dans certains cas, les deux prisonniers peuvent perdre).

Représentations des jeux

Les jeux coopératifs sont généralement présentés sous la forme de fonctions caractéristiques alors que les jeux non coopératifs sont représentés sous forme normale ou sous forme extensive [97, 98, 99].

- **Forme normale** : cette forme est très utile pour décrire des situations dans lesquelles les joueurs jouent en même temps. Un jeu sous forme normale est défini par une liste de joueurs, un ensemble des stratégies possibles pour chacun des joueurs et les préférences de chacun des joueurs sur l'ensemble des combinaisons stratégiques possibles. Pour un jeu à deux joueurs avec un ensemble fini de stratégies, comme le dilemme du prisonnier, il est courant de le représenter sous sa forme normale à l'aide d'une matrice des gains. Il s'agit d'un tableau à double-entrée qui énumère sur chaque côté les stratégies possibles des joueurs. Chaque cellule de ce tableau note le couple de gains des deux joueurs. Le Tableau 6.5 illustre la matrice des gains du jeu le dilemme du prisonnier où chaque joueur décide soit de Coopérer (C) ou de Ne pas Coopérer (NC).
- **Forme extensive** : la forme extensive est l'un des moyens les plus simples pour représenter un jeu. Il s'agit d'un modèle où les joueurs choisissent séquentiellement leurs actions, jusqu'à l'étape où le jeu est déclaré terminé.

Tableau 6.5: Exemple d'une matrice de gain.

	C	NC
C	(-5,-5)	(0,-10)
NC	(-10,0)	(-1,-1)

Tout jeu sous forme extensive est représenté par un arbre de décision (graphe connexe sans cycle) où :

- Chaque nœud terminal correspond un résultat du jeu,
- Chaque nœud non terminal est associé au joueur qui décide, c'est à son tour de jouer,
- Chaque arc représente chacune des actions que ce joueur peut entreprendre à ce point du jeu.

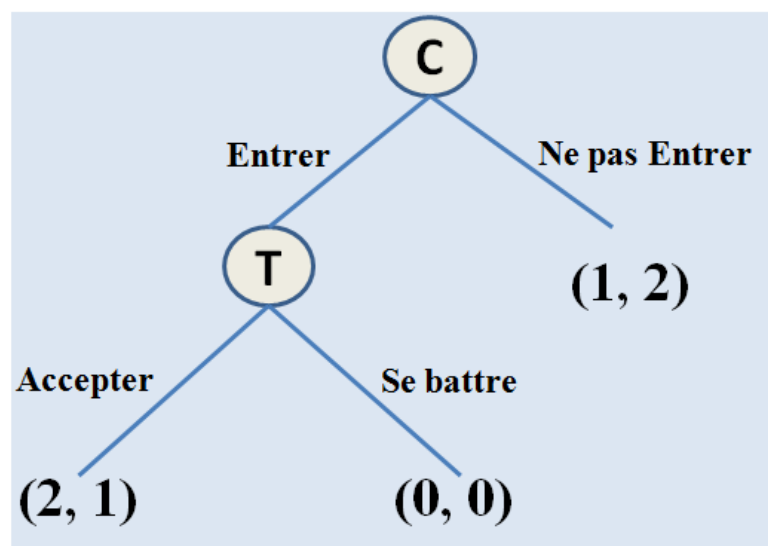


FIGURE 6.6: Jeu sous forme extensive.

La FIGURE 6.6 représente l'arbre de décision du jeu monopole et challenger où les préférences du challenger peuvent être représentées par la fonction du gain u_1 pour laquelle : $u_1(\text{Entrer}, \text{Accepter}) = 2$, $u_1(\text{Ne pas entrer}) = 1$ et $u_1(\text{Entrer}, \text{Se battre}) = 0$. De même, les préférences du titulaire peuvent être formalisées par la fonction du gain u_2 avec : $u_2(\text{Ne pas entrer}) = 2$, $u_2(\text{Entrer}, \text{Accepter}) = 1$ et $u_2(\text{Entrer}, \text{Se battre}) = 0$.

- Jeux sous forme caractéristiques : Il s'agit d'une forme de jeu coopératif dans laquelle le jeu est noté $G = (N, v)$, où :
 - N est l'ensemble des joueurs.
 - v est la fonction caractéristique, elle associe à chaque coalition $S \subseteq N$ la valeur $v(S)$, c'est-à-dire le gain (ou le coût) obtenu par la coalition S .

Par exemple, le jeu d'allocation des coûts est un jeu à trois joueurs $N = \{A, B, C\}$ et la valeur de chaque coalition S est définie par l'économie de coût réalisée après avoir signé un contrat commun. Nous obtenons ainsi $v(\{A\}) = v(\{B\}) = v(\{C\}) = 0$, $v(\{A, B\}) = 4$, $v(\{A, C\}) = 3$, $v(\{B, C\}) = 2$ et $v(N) = 6$.

Équilibre de Nash

Plusieurs concepts de solutions ont été définis pour résoudre chaque type de jeu. Il s'agit principalement de la répartition du gain entre les joueurs. L'équilibre de Nash représente le concept de solution le plus utilisé dans la théorie des jeux [102]. La notion d'équilibre de Nash est une situation telle qu'aucun joueur n'a intérêt à dévier (seul) de la situation obtenue.

Formellement, un équilibre de Nash est un profil de stratégies $s^* = (s_1^*, s_2^*, \dots, s_n^*)$ tel que pour tout joueur i et pour toute stratégie $s' \in S_i$: $u_i(s_i^*, s_{-i}^*) \geq u_i(s', s_{-i}^*)$. Par exemple, le jeu dilemme du prisonnier a un unique équilibre de Nash en stratégies (NC, NC) .

De plus, pour les jeux finis, c'est-à-dire les jeux pour lesquels l'ensemble des stratégies de chacun des joueurs est fini, nous distinguons deux variantes de l'équilibre de Nash :

- A. **Équilibre de Nash en stratégie pure** : c'est l'équilibre d'un jeu dans lequel les joueurs choisissent une stratégie de manière déterministe.
- B. **Équilibre de Nash en stratégie mixte** : c'est l'équilibre dans lequel les joueurs choisissent de jouer les différentes stratégies possibles de manière probabiliste. Autrement dit, une stratégie mixte du joueur i est une distribution de probabilités p_i définie sur l'ensemble des stratégies pures du joueur i ($\sum_{s_i} p_{i,s_i} = 1$) avec p_{i,s_i} est la probabilité que i joue la stratégie pure s_i .

6.2.2 Description du jeu dans GSS-VC

Dans notre approche, nous considérons deux joueurs : le Véhicule Fournisseur (VF) et le Véhicule Consommateur (VC) ; Joueurs = {VF, VC}. Chacun des deux joueurs a deux stratégies possibles. D'une part, le VC choisit entre Consommer (C) ou Ne pas Consommer (NC), cela dépend aux qualités de service et le prix proposé par le VF. D'autre part, le VF peut choisir entre Offrir (O) ou Ne pas Offrir (NO) son service au VC. Il est bien clair que le VC cherche à sélectionner un VF en offrant le meilleur rapport qualité/prix tandis que le VF détermine le prix en fonction de ses qualités de services. Ainsi, VF et VC vont sélectionner la stratégie la plus adéquate afin de maximiser leurs gains. L'équilibre de jeu correspond à l'état où aucun joueur ne peut augmenter son gain s'il change de stratégie. Pour cela, nous nous sommes basés

Tableau 6.6: Matrice des gains du VF et VC.

VF / VC	C	NC	
O	(X_{11}, Y_{11})	(X_{12}, Y_{12})	p
NO	(X_{21}, Y_{21})	(X_{22}, Y_{22})	$1 - p$
	q	$1 - q$	

sur l'équilibre de Nash pour déterminer cet état.

Nous représentons chaque service par un nombre de paquets de données qui vont être échangées entre VF et VC. De plus, pour déterminer le gain de chacun des joueurs, nous nous sommes appuyés sur les critères de qualité suivants : débit de données (q_{DTH}), le rapport d'offre de service avec succès (q_{SER}), la durée d'exécution (q_{ED}) et le prix d'exécution de service (q_{EP}) (voir sous-section 6.1.2).

La matrice des gains

Le Tableau 6.6 illustre la matrice des gains du VF et VC, où p et $1 - p$ sont les probabilités que le VF sélectionne les stratégies O et NO, respectivement, q et $1 - q$ sont les probabilités que le VC sélectionne les stratégies C et NC, respectivement. Nous distinguons quatre combinaisons de stratégies qui peuvent être sélectionnées par les deux joueurs :

1. Combinaison de stratégies (Offrir & Consommer)

$$\begin{aligned} - X_{11} &= N * q_{EP} - (q_{ED} + q_{SER}) \\ - Y_{11} &= N - (q_{ED} + N * q_{EP} + q_{SER}) \end{aligned}$$

Dans ce cas, le VF offre son service et le VC décide de le consommer. X_{11} représente le gain du VF qui est égale au nombre de paquets de données du service (N) multiplié par le prix de chaque paquet. Ce gain est par contre affecté par la durée d'exécution du service et le rapport d'offre de service avec succès. De plus, le gain du VC (Y_{11}) correspond au service consommé en termes de paquets de données. Ce gain est affecté aussi par le prix du service, la durée d'exécution du service et le rapport d'offre de service avec succès.

2. Combinaison de stratégies (Offrir & Ne pas consommer)

$$\begin{aligned} - X_{12} &= -N * q_{EP} \\ - Y_{12} &= -M * N \end{aligned}$$

Le VF offre son service alors que le VC refuse de le consommer. En effet, le VC refuse

de consommer un service à cause des qualités proposées ou bien il a déjà sélectionné un autre VF. X_{12} représente l'échec du VF à offrir son service qui est le prix du service. Dans ce cas, le VC est pénalisé en multipliant son gain par M qui reflète le nombre de fois qu'il a refusé de consommer un service alors que le VF accepte de l'offrir (Y_{12}).

3. Combinaison de stratégies (Ne pas offrir & Consommer)

$$\begin{aligned} - X_{21} &= \begin{cases} 0 & (Busy) \\ -L * N * q_{EP} & (Else) \end{cases} \\ - Y_{21} &= -N \end{aligned}$$

Dans ce cas, le VC décide de consommer le service mais le VF refuse de l'offrir. Le VF peut ne pas offrir son service car il est occupé (il a offert ou veut offrir son service à un autre VC). Si ce n'est pas le cas, il sera pénalisé en multipliant le prix de service par L qui représente le nombre de fois qu'il a refusé d'offrir un service tandis que le VC accepte de le consommer (X_{21}). Y_{21} reflète l'échec du VC à consommer le service.

4. Combinaison de stratégies (Ne pas offrir & Ne pas consommer)

$$\begin{aligned} - X_{22} &= 0 \\ - Y_{22} &= 0 \end{aligned}$$

Dans ce cas, les gains du VF et VC (X_{22} et Y_{22}) sont nuls puisque tous les deux décident de ni d'offrir ni de consommer le service, respectivement.

Nous notons que pour normaliser les gains des deux joueurs, chaque critère de qualité est standardisé et normalisé entre 0 et 1, en divisant sa valeur par la valeur standard (max). Par exemple, le prix de service est normalisé en fonction du prix standard $\max(q_{EP})$, c.à.d., $q_{EP} = q_{EP}/\max(q_{EP})$.

L'équilibre de Nash en stratégie mixte

L'état stable du jeu correspond à l'état où chacun des joueurs n'a pas intérêt à changer de stratégie. Pour cela, nous utilisons l'équilibre de Nash en stratégie mixte afin de déterminer l'état où le VF et le VC ne changent pas leurs stratégies Offrir et Consommer, respectivement.

Theorem 3 *Il existe un équilibre de Nash en stratégie mixte {VF (Offrir, p^*), VC (Consommer, q^*)} dans lequel le VF choisit l'action Offrir avec une probabilité $p > p^*$ et le VC choisit l'action Consommer avec une probabilité $q > q^*$.*

Preuve :

La stratégie mixte du VF est définie comme suit : $P = (p, 1 - p)$ et les gains du VC sont :

$$- u_{VC}(\text{Consommer}) = Y_{11} * p + Y_{21} * (1 - p) = p * (Y_{11} - Y_{21}) + Y_{21}$$

$$- u_{VC}(\text{Ne pas Consommer}) = Y_{12} * p + Y_{22} * (1 - p) = p * Y_{12}$$

Le VC sélectionne l'action Consommer lorsque $u_{VC}(\text{Consommer}) > u_{VC}(\text{Ne pas Consommer})$.

Par conséquent, nous obtenons :

$$- p > p^* \text{ where } p^* = \frac{-Y_{21}}{Y_{11} - Y_{21} - Y_{12}} \text{ with } 0 < p^* \leq 1$$

La stratégie mixte du VC est définie comme suit : $Q = (q, 1 - q)$ et les gains du VF sont :

$$- u_{VF}(\text{Offrir}) = X_{11} * q + X_{21} * (1 - q) = q * (X_{11} - X_{12}) + X_{12}$$

$$- u_{VF}(\text{Ne pas Offrir}) = X_{21} * q + X_{22} * (1 - q) = q * X_{21}$$

Le VF sélectionne l'action Offrir lorsque $u_{VF}(\text{Offrir}) > u_{VF}(\text{Ne pas Offrir})$. Par conséquent, nous obtenons :

$$- q > q^* \text{ où } q^* = \frac{-X_{12}}{X_{11} - X_{12} - X_{21}} \text{ with } 0 < q^* \leq 1$$

Nous concluons que lorsque la probabilité que le VC choisisse l'action consommer est supérieure ou égale à q^* et la probabilité que le VF choisisse l'action offrir est supérieure ou égale à p^* , les deux joueurs ne changent pas leurs stratégies. Ce qui correspond à l'état d'équilibre du jeu.

Simulation et résultats

Dans cette section, nous évaluons les performances de GSS-VC dans un environnement urbain (Manhattan) à travers la simulation. Les paramètres de simulation Pour valider notre modèle de sélection, nous avons utilisé les mêmes simulateurs ainsi que les mêmes paramètres que nous avons utilisés pour valider le protocole DCCS-VC (voir sous-section 5.4.1 dans chapitre 5). Le Tableau 6.7 montre les paramètres de simulation les plus importants.

Tableau 6.7: Les paramètres de simulation.

Paramètres	Valeurs
Temps de simulation	1000s
Zone de simulation	9x9 Km ²
Portée de communication	500 m
Débit	18 Mb/s
Densité de véhicules (CD)	[300, 1500]
Densité des véhicules fournisseurs (PV)	1/4, 1/3 , 1/2 de la densité de véhicules
Prix des services	[10, 100]\$
Le nombre maximum de services offerts par véhicule fournisseur	3 services
Le nombre maximum de demande de services par véhicule consommateur	2 services

Durant la simulation, chaque VC joue avec tous les VFs dans son voisinage. Par la suite, il sélectionne le VF avec qui il atteint la meilleure probabilité de consommer le service (probabilité q). Nous choisissons les métriques d'évaluation suivantes afin de valider notre modèle de sélection :

- **La convergence de la probabilité (PC, Probability Convergence)** : pour être plus réaliste, nous avons étudié la convergence de la probabilité entre les VFs et les VCs dans trois périodes de temps différentes : la période off-peak où le taux de la demande de service est réduit (par exemple durant la nuit), la période mid-peak où le taux de la demande est moyen et la période on-peak où le taux de demande est grand (par exemple durant une congestion).
- **Le délai moyen du service (ASD, Average Service Delay)** : mesure le délai de temps avant de recevoir tous les paquets de données du service par le VC. Cette métrique est utilisée afin d'évaluer le délai généré par le modèle de la théorie des jeux.
- **Le gain du service (SG, Service Gain)** : représente le gain des VCs après avoir consommés leurs services. Nous déterminons ce gain à l'aide de Y_{11} .
- **Le rapport d'offre de service avec succès (SER, Successful Execution Ratio)** : mesure la fiabilité d'offre de services, c.à.d. le taux des VCs qui ont consommé leurs services correctement et dans les délais prévus.

De plus, nous avons comparé GSS-VC avec deux autres modèles de sélection : (i) le modèle

Neutral [82] dans lequel un VC sélectionne toujours le premier VF disponible sans prendre en considération les qualités de service offertes. (ii) le modèle de négociation [81] qui propose un système de négociation entre les fournisseurs et les consommateurs à travers une entité de confiance intermédiaire.

Résultats et discussions

Dans cette section, nous présentons les résultats obtenus pour les métriques : la convergence de la probabilité, le gain du service et le rapport d'offre de service avec succès. L'objectif de notre évaluation est de répondre à la question : *Quand et comment les VCs sélectionnent les meilleurs VFs ?* Le Tableau 6.8 montre les cinq meilleurs véhicules fournisseurs dans chaque période de temps et les probabilités d'offre et de consommation qui correspondent à chaque sélection de service. D'un côté, nous observons que la probabilité de consommation peut atteindre jusqu'à 0.86 durant la période off-peak. Cette valeur diminue à 0.77 et 0.65 dans les périodes mid-peak et on-peak, respectivement. D'un autre côté, la probabilité d'offre augmente de 0.53 durant la période off-peak à 0.58 et 0.75 durant les périodes mid-peak et on-peak, respectivement. Ces résultats peuvent être justifiés par le fait que durant la période on-peak, le taux de demande de service augmente ce qui pousse les VFs à augmenter leurs prix de services afin de maximiser leurs gains. Par conséquent, la probabilité de consommation des VCs diminue de 0.86 à 0.65.

Tableau 6.8: Les cinq meilleurs véhicules fournisseurs dans chaque période.

	période off-peak				période Mid-peak				période On-peak			
	VC ID	VF ID	Probabilité Consommer (q)	Probabilité Offrir (p)	VC ID	VF ID	Probabilité Consommer (q)	Probabilité Offrir (p)	VC ID	VF ID	Probabilité Consommer (q)	Probabilité Offrir (p)
1	287	34	0.86	0.53	698	467	0.77	0.58	309	100	0.65	0.75
2	502	122	0.86	0.53	23	529	0.77	0.58	119	1391	0.65	0.75
3	129	98	0.86	0.53	176	190	0.76	0.59	1487	609	0.64	0.77
4	398	213	0.85	0.55	239	96	0.76	0.59	765	419	0.63	0.79
5	1100	387	0.85	0.55	004	702	0.76	0.59	901	200	0.63	0.79

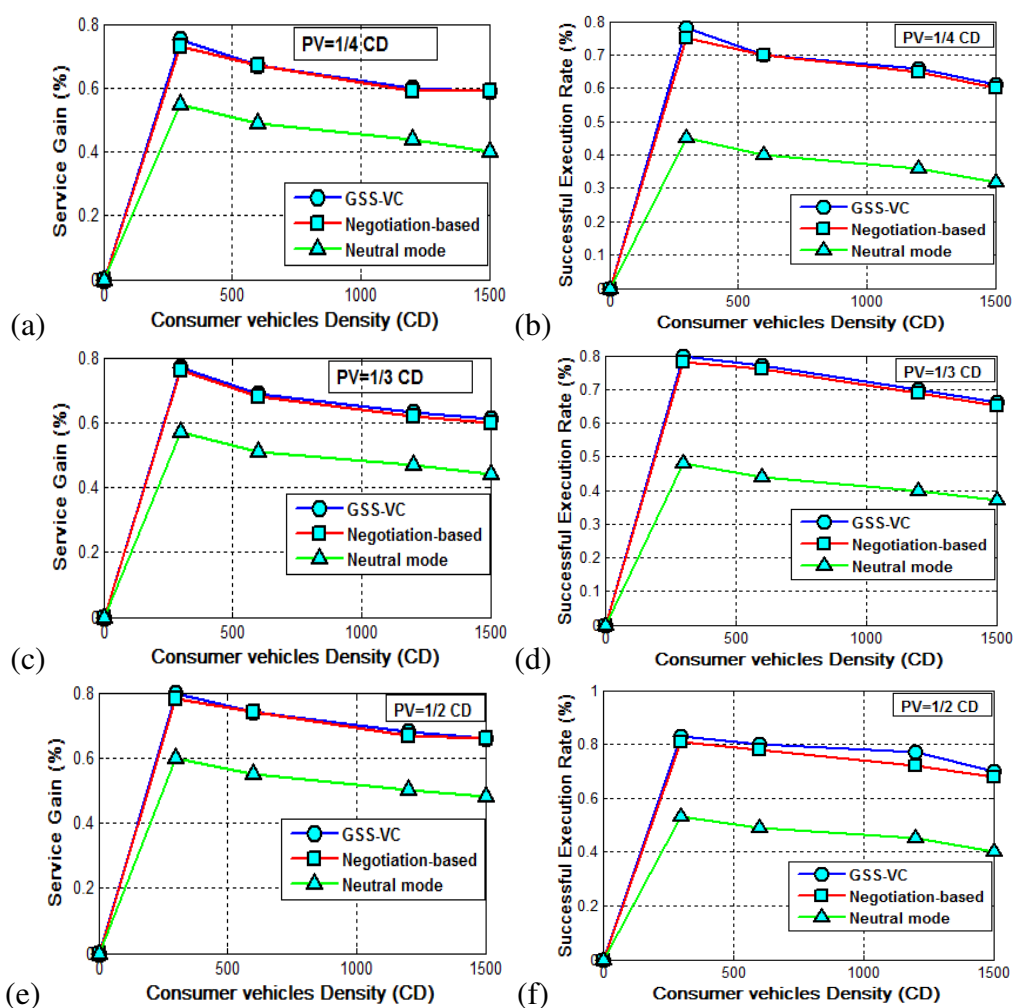


FIGURE 6.7: Évaluation des performances du GSS-VC en termes du gain de service et le rapport d'offre de service avec succès durant la période on-peak.

La FIGURE 6.7 illustre une comparaison de performances du GSS-VC, le modèle de négociation et le modèle Neutral. Nous remarquons bien que GSS-VC et le modèle de négociation génèrent les mêmes performances qui sont meilleures que celles du mode Neutral. Dans le mode Neutral, un VC sélectionne le premier VF disponible sans prendre en considération les qualités de service qu'il offre (VF). Ainsi, les VCs peuvent sélectionner des VFs en offrant des mauvaises qualités de services ce qui réduit le gain et la fiabilité du service du mode Neutral par rapport aux autres modèles. Nous remarquons aussi que le gain et la fiabilité de service augmentent proportionnellement à la densité des VFs. En effet, la présence de plus de VFs offre aux VCs plus de chance pour sélectionner les meilleurs fournisseurs en termes de qualités de service, ce qui améliore le gain et la fiabilité de consommation de service.

6.3 Conclusion

Dans ce chapitre, nous avons exposé une vue générale sur la logique floue et la théorie des jeux qui prouvaient leurs efficacités à résoudre plusieurs problématiques dans différents domaines. Par la suite, nous avons présentés nos deux modèles de sélection de service fondés sur ces deux concepts. Nos modèles de sélection considèrent les préférences des consommateurs et les exigences des fournisseurs pour sélectionner le meilleur véhicule fournisseur, selon des critères de qualité. Nous avons aussi étudié l'interaction entre les consommateurs et fournisseurs dans des périodes de temps différentes dans le modèle GSS-VC.

Les résultats de simulation montrent que nos modèles améliorent l'efficacité et la fiabilité de la sélection de service en maximisant le gain des consommateurs et en sélectionnant les meilleurs fournisseurs de service.

Cette contribution constitue nos premiers pas vers la modélisation des problèmes plus compliqués notamment dans le contexte du Cloud Ad hoc véhiculaire à savoir : la sélection de service en cas où les consommateurs demandent plus qu'un seul service, formation des coalitions de fournisseurs afin de répondre aux besoins des consommateurs, la gestion d'allocation de ressources des fournisseurs et la mise en œuvre des mécanismes de reprise, en cas où les fournisseurs quittent le réseau avant que les consommateurs consomment leurs services, etc.

Chapitre 7

Conclusion générale et perspectives

CETTE thèse a eu pour objectif de proposer des solutions adaptées aux réseaux Ad hoc véhiculaires afin de le rendre plus sûr, plus efficace et plus confortable. En effet, les réseaux ad hoc véhiculaires (VANet) constituent une nouvelle application des réseaux Ad hoc mobiles (MANet). Les particularités de ces réseaux véhiculaires proviennent des communications sans-fil qui peuvent s'établir entre véhicules ou avec les stations de base (RSUs) ainsi que la grande variété de ressources dont les véhicules sont dotés. Dans cette thèse, nous avons adressé deux applications prometteuses à savoir la collecte de données et l'offre de ressources en tant que services.

La collecte de données repose sur l'échange de données entre les véhicules qui permet de couvrir l'imperfection et l'insuffisance de perception d'un seul véhicule sur son environnement et rend ces données plus précises et plus pertinentes. Pour cela, l'objectif principale des protocoles de collecte de données est de définir comment les données collectées seront échangées entre les véhicules, ou bien avec l'infrastructure, afin de fournir des informations, en temps réel ou non, plus fiables et plus pertinentes.

Dans ce contexte, nous avons conçu deux protocoles de collecte de données efficaces et fiables pour réaliser différentes applications, nommés ECDGP et DDGP. Grâce à nos techniques d'accès au canal VL-CSMA et D-SDMA, DDGP et ECDGP garantissent trois propriétés très avantageuses dans un réseau qui se caractérise par une forte mobilité des nœuds. La première de ces propriétés réside dans le fait qu'ils offrent aux véhicules un accès équitable et avec moins de collisions. La deuxième est l'efficacité de la collecte par la collecte d'un maximum de données en un minimum de temps et en agrégeant les données collectées. Enfin, la troisième consiste en la mise en œuvre d'un mécanisme de retransmission de données en cas d'erreur et donc en

l'amélioration de la fiabilité de la collecte.

De plus, l'étude de performance de nos protocoles a été réalisée de manière analytique, à travers la preuve de correction et l'étude de la complexité, ainsi que par des multiples scénarios de simulations. Les résultats de simulation ont montré que nos protocoles sont plus performants en termes de fiabilité et d'efficacité de la collecte de données, par rapport aux autres protocoles de collecte existants.

D'autre part, les capacités de véhicules en termes de ressources et le non exploitation de ces ressources ont conduit à la naissance du concept Cloud véhiculaire. Dans ce type d'application, les véhicules peuvent agir comme étant des serveurs du Cloud mobile et offrir leurs ressources en tant que services à d'autres véhicules. Mais, vu la forte mobilité des véhicules, les fournisseurs de services peuvent rejoindre ou quitter le Cloud à tout moment. Ainsi, un protocole d'offre et de consommation de services doit fournir aux véhicules consommateurs la possibilité de découvrir les véhicules fournisseurs et leurs services, sélectionner le service (fournisseur) le plus adéquat avant de pouvoir le consommer. Dans ce contexte, notre contribution était double.

Premièrement, nous avons proposé deux protocoles d'offre et de consommation de services, nommés DCCS-VC et RCS-VC. DCCS-VC et RCS-VC exploitent, respectivement, les bus publics et l'infrastructure du réseau cellulaire LTE-A comme étant des répertoires de Cloud, dans lesquels les véhicules fournisseurs enregistrent leurs offres et à partir desquels les véhicules consommateurs découvrent tous les services offerts. Les résultats de simulation ont montré l'efficacité de nos protocoles à minimiser les délais de découverte et de consommation de services en les comparant avec des protocoles déjà proposés.

Deuxièmement, nous avons proposé deux modèles de sélection de services dans le Cloud véhiculaire. Le premier modèle se base sur la logique floue pour permettre à chaque bus public de sélectionner le véhicule fournisseur le plus adéquat dans le protocole DCCS-VC, au lieu d'envoyer aux consommateurs la liste de tous les véhicules fournisseurs. Tandis que le deuxième modèle se base sur la théorie de jeux, nommé GSS-VC pour Game theory-based approach for Service Selection in Vehicular Cloud. GSS-VC est un modèle générique qui permet aux véhicules consommateurs de sélectionner les meilleurs fournisseurs sur la route. Nos modèles prennent en considération les préférences des consommateurs et les exigences des fournisseurs afin de déterminer le meilleur fournisseur en termes de qualité de service.

De nombreuses perspectives de recherche peuvent être envisagées. Dans ce qui suit, nous présentons celles qui nous paraissent être les plus intéressantes :

- Sécuriser nos protocoles de collecte de données afin de protéger la vie privée des conducteurs et leurs données collectées.
- Dans le contexte du Cloud véhiculaire, plusieurs défis restent à relever, tels que :
 - La formation d’une architecture du Cloud véhiculaire flexible et robuste afin de faire face à la mobilité des nœuds. Cette architecture se compose des véhicules consommateurs, fournisseurs et d’un véhicule intermédiaire (broker).
 - La sélection de services dans le cas où les consommateurs demandent plus qu’un seul service.
 - La formation des coalitions de fournisseurs afin de répondre aux besoins des consommateurs,
 - La gestion d’allocation de ressources des fournisseurs et la mise en œuvre des mécanismes de reprise, dans le cas où les fournisseurs quittent le Cloud avant que les consommateurs consomment leurs services.
 - Sécuriser les communications V2V afin de garantir l’intégrité des données échangées.

Liste des publications

Reuves internationnales

1. Brik, B., Lagraa, N., Lakas, A., Cherroun, H., and Cheddad, A. (2016) ECDGP : extended cluster-based data gathering protocol for vehicular networks. *Wirel. Commun. Mob. Comput.*, 16 : 12381255. doi : 10.1002/wcm.2591.
2. Bouziane Brik, Nasreddine Lagraa, Abderrahmane Lakas, Abbas Cheddad : DDGP : Distributed Data Gathering Protocol for vehicular networks. *Vehicular Communications* 4 : 15-29 (2016).

Conférence internationnales

1. Brik, B. ; Lagraa, N. ; Cherroun, H. ; Lakas, A., Token-based Clustered Data Gathering Protocol (TCDGP) in vehicular networks, *Wireless Communications and Mobile Computing Conference (IWCMC)*, 2013 9th International , vol., no., pp.1070,1074, 1-5 July 2013 Doi : 10.1109/IWCMC.2013.6583705
2. B. Brik, N. Lagraa, A. Lakas, and Y. Ghamri-Doudane, Finding a public bus to rent out services in vehicular clouds, in *Vehicular Technology Conference (VTC Fall)*, 2015 IEEE 82nd, pp. 17, IEEE, 2015.
3. B. Brik, N. Lagraa, A. Lakas, and Y. Ghamri-Doudane, RCS-VC : Renting out and consuming services in vehicular clouds based on LTE-A, in *Global Information Infrastructure and Networking Symposium (GIIS)*, 2015, pp. 16, IEEE, 2015
4. B. Brik, N. Lagraa, Y. Ghamri-Doudane and A. Lakas, Finding the most adequate public bus in Vehicular Clouds, 2016 International Conference on Wireless Networks and Mobile Communications (WINCOM), Fez, Morocco, 2016, pp. 67-74.

À soumettre

- B. Brik, J.A Khan, Y. Ghamri-Doudane, N. Lagraa and A. Lakas GSS-VC : Game theory-based model for Service Selection in Vehicular Cloud.
- N.Tamani, B.Brik, N.Lagraa, Y.Ghamri-Doudane. Ranking Method for Vehicular Cloud Service Provider Selection : a Flexible Approach.
- F.Abdelhafidi, S.Djeradi, B.Brik, N.Lagraa, A.Lakas. Finding the Most Suitable Service Provider in Vehicular Clouds.

Bibliographie

- [1] Saif Al-Sultan, Moath M. Al-Doori, Ali H. Al-Bayatti, and Hussien Zedan. A comprehensive survey on vehicular ad hoc network. *J. Netw. Comput. Appl.*, 37 :380–392, January 2014.
- [2] R. Hussain, F. Abbas, Junggab Son, Sangjin Kim, and Heekuck Oh. Using public buses as mobile gateways in vehicular clouds. In *Consumer Electronics (ICCE), 2014 IEEE International Conference on*, pages 175–176, Jan 2014.
- [3] E.C.M. Transport. *Gérer la congestion urbaine*. OECD Publishing, 2010.
- [4] Oms, accidents de la route. <http://www.who.int/mediacentre/factsheets/fs358/fr/>.
- [5] M. McDonald. *Intelligent Transport Systems in Europe : Opportunities for Future Research*. World Scientific, 2006.
- [6] Hannes Hartenstein and Kenneth P Laberteaux. *VANET : Vehicular Applications and Inter-Networking Technologies*. Wiley Online Library, 2010.
- [7] White Paper. DSRC technology and the DSRC Industry consortium (DIC) prototype team. January 2005.
- [8] IEEE standard 802.11p. IEEE standard for information technology — telecommunications and information exchange between systems — local and metropolitan area networks — specific requirements — part 11 : Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 6 : Wireless access in vehicular environments, July 2010.
- [9] IEEE 1609 family of standards for wireless access in vehicular environments. *available from IEEE Standards*, 2009.
- [10] Continuous air-interface, long and medium range-CALM,. <http://www.calm.hu/>. [Online].
- [11] ETSI. <http://www.etsi.org/intelligenttransportsystems.aspx>. Page consultée le 11/12/2016.

- [12] Y. Qian and N. Moayeri. Survey of media access control protocols for vehicular ad hoc networks. *IET Communications*, 5 :1619–1631, 2011.
- [13] Jianqi Liu, Jiafu Wan, Qinruo Wang, Pan Deng, Keliang Zhou, and Yupeng Qiao. A survey on position-based routing for vehicular ad hoc networks. *Telecommun. Syst.*, 62(1) :15–30, May 2016.
- [14] Dharmendra Sutariya and Dr. S. N. Pradhan. Data dissemination techniques in vehicular ad hoc network. *International Journal of Computer Applications*, 8(10) :35–39, October 2010.
- [15] I. Salhi, M. O. Cherif, and S. M. Senouci. A new architecture for data collection in vehicular networks. In *2009 IEEE International Conference on Communications*, pages 1–6, June 2009.
- [16] W. R. Chang, H. T. Lin, and B. X. Chen. Trafficgather : An efficient and scalable data collection protocol for vehicular ad hoc networks. In *2008 5th IEEE Consumer Communications and Networking Conference*, pages 365–369, Jan 2008.
- [17] M. Gerla. Vehicular cloud computing. In *Ad Hoc Networking Workshop (Med-Hoc-Net), 2012 The 11th Annual Mediterranean*, pages 152–155, June 2012.
- [18] Mohamed Nidhal Mejri, Jalel Ben-Othman, and Mohamed Hamdi. Survey on {VANET} security challenges and possible cryptographic solutions. *Vehicular Communications*, 1(2) :53 – 66, 2014.
- [19] Bouziane Brik, Nasreddine Lagraa, Abderrahmane Lakas, Hadda Cherroun, and Abbas Cheddad. Ecdgp : extended cluster-based data gathering protocol for vehicular networks. *Wireless Communications and Mobile Computing*, 16(10) :1238–1255, 2016. wcm.2591.
- [20] Bouziane Brik, Nasreddine Lagraa, Mohamed Bachir Yagoubi, and Abderrahmane Lakas. An efficient and robust clustered data gathering protocol (cdgp) for vehicular networks. In *Proceedings of the Second ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications, DIVANet '12*, pages 69–74, New York, NY, USA, 2012. ACM.
- [21] S. V. Bana and P. Varaiya. Space division multiple access (sdma) for robust ad hoc vehicle communication networks. In *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No.01TH8585)*, pages 962–967, 2001.
- [22] Bouziane Brik, Nasreddine Lagraa, Abderrahmane Lakas, and Abbas Cheddad. Ddgp : Distributed data gathering protocol for vehicular networks. *Vehicular Communications*, 4 :15 – 29, 2016.

- [23] S. Bitam, A. Mellouk, and S. Zeadally. Vanet-cloud : a generic cloud computing model for vehicular ad hoc networks. *IEEE Wireless Communications*, 22(1) :96–102, February 2015.
- [24] Hassan Abid, Luong Thi Thu Phuong, Jin Wang, Sungyoung Lee, and Saad Qaisar. V-cloud : Vehicular cyber-physical systems and cloud computing. In *Proceedings of the 4th International Symposium on Applied Sciences in Biomedical and Communication Technologies*, ISABEL '11, pages 165 :1–165 :5. ACM, 2011.
- [25] Dimal Baby, R. D. Sabareesh, R. A. K. Saravanaguru, and Arunkumar Thangavelu. *VCR : Vehicular Cloud for Road Side Scenarios*, pages 541–552. 2013.
- [26] Swarun Kumar, Shyamnath Gollakota, and Dina Katabi. A cloud-assisted design for autonomous driving. In *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, pages 41–46, New York, NY, USA, 2012. ACM.
- [27] K. Mershad and H. Artail. Finding a star in a vehicular cloud. *Intelligent Transportation Systems Magazine, IEEE*, 5(2) :55–68, Summer 2013.
- [28] H. R. Arkian, R. E. Atani, and S. Kamali. Fcvca : A fuzzy clustering-based vehicular cloud architecture. In *2014 7th International Workshop on Communication Technologies for Vehicles (Nets4Cars-Fall)*, pages 24–28, Oct 2014.
- [29] R. Hussain, F. Abbas, J. Son, D. Kim, S. Kim, and H. Oh. Vehicle witnesses as a service : Leveraging vehicles as witnesses on the road in vanet clouds. In *2013 IEEE 5th International Conference on Cloud Computing Technology and Science*, volume 1, pages 439–444, Dec 2013.
- [30] Puya Ghazizadeh, Ravi Mukkamala, and Samy El-Tawab. Scheduling in vehicular cloud using mixed integer linear programming. In *Proceedings of the First International Workshop on Mobile Sensing, Computing and Communication*, MSCC '14, pages 7–12, New York, NY, USA, 2014. ACM.
- [31] Hamid Reza Arkian, Reza Ebrahimi Atani, Abolfazl Diyanat, and Atefe Pourkhalili. A cluster-based vehicular cloud architecture with learning-based resource management. *The Journal of Supercomputing*, 71(4) :1401–1426, 2015.
- [32] Bouziane Brik, Nasreddine Lagraa, Abderrahmane Lakas, and Yacine Ghamri-Doudane. Finding a public bus to rent out services in vehicular clouds. In *Vehicular Technology Conference (VTC Fall), 2015 IEEE 82nd*, pages 1–7. IEEE, 2015.
- [33] Bouziane Brik, Nasreddine Lagraa, Abderrahmane Lakas, and Yacine Ghamri-Doudane. Rcs-vc : Renting out and consuming services in vehicular clouds based on lte-a. In

- Global Information Infrastructure and Networking Symposium (GIIS), 2015*, pages 1–6. IEEE, 2015.
- [34] D. Astely, E. Dahlman, A. Furuskar, Y. Jading, M. Lindstrom, and S. Parkvall. Lte : the evolution of mobile broadband. *Communications Magazine, IEEE*, 47(4) :44–51, April 2009.
- [35] R. Zhang, J. Pan, D. Xie, and F. Wang. Ndcmc : A hybrid data collection approach for large-scale wsns using mobile element and hierarchical clustering. *IEEE Internet of Things Journal*, 3(4) :533–543, Aug 2016.
- [36] F. Zhou, Z. Chen, S. Guo, and J. Li. Maximizing lifetime of data-gathering trees with different aggregation modes in wsns. *IEEE Sensors Journal*, 16(22) :8167–8177, Nov 2016.
- [37] Z. Xu, L. Chen, C. Chen, and X. Guan. Joint clustering and routing design for reliable and efficient data collection in large-scale wireless sensor networks. *IEEE Internet of Things Journal*, 3(4) :520–532, Aug 2016.
- [38] I. Khan, F. Belqasmi, R. Glitho, N. Crespi, M. Morrow, and P. Polakos. Wireless sensor network virtualization : A survey. *IEEE Communications Surveys Tutorials*, 18(1) :553–576, Firstquarter 2016.
- [39] L. M. Borges, F. J. Velez, and A. S. Lebres. Survey on the characterization and classification of wireless sensor network applications. *IEEE Communications Surveys Tutorials*, 16(4) :1860–1890, Fourthquarter 2014.
- [40] A. A. Kumar S., K. Ovsthus, and L. M. Kristensen. An industrial perspective on wireless sensor networks x2014 ; a survey of requirements, protocols, and challenges. *IEEE Communications Surveys Tutorials*, 16(3) :1391–1412, Third 2014.
- [41] L. Yang, J. Xu, G. Wu, and J. Guo. Road probing : Rsu assisted data collection in vehicular networks. In *2009 5th International Conference on Wireless Communications, Networking and Mobile Computing*, pages 1–4, Sept 2009.
- [42] Y. Dieudonné, B. Ducourthial, and S. M. Senouci. Col : A data collection protocol for vanet. In *2012 IEEE Intelligent Vehicles Symposium*, pages 711–716, June 2012.
- [43] B. Brik, N. Lagraa, H. Cherroun, and A. Lakas. Token-based clustered data gathering protocol(tcdgp) in vehicular networks. In *2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 1070–1074, July 2013.
- [44] Claudio E. Palazzi, Fabio Pezzoni, and Pedro M. Ruiz. Delay-bounded data gathering in urban vehicular sensor networks. *Pervasive and Mobile Computing*, 8(2) :180 – 193, 2012.

- [45] G. Remy, S. M. Senouci, F. Jan, and Y. Gourhant. Lte4v2x : Lte for a centralized vanet organization. In *2011 IEEE Global Telecommunications Conference - GLOBECOM 2011*, pages 1–6, Dec 2011.
- [46] G. Rémy, S. M. Senouci, F. Jan, and Y. Gourhant. Lte4v2x - impact of high mobility in highway scenarios. In *Global Information Infrastructure Symposium - GIIS 2011*, pages 1–7, Aug 2011.
- [47] G. Rémy, S. M. Senouci, F. Jan, and Y. Gourhant. Lte4v2x x2014 ; collection, dissemination and multi-hop forwarding. In *2012 IEEE International Conference on Communications (ICC)*, pages 120–125, June 2012.
- [48] Katrin Bilstrup, Elisabeth Uhlemann, Erik G. Ström, and Urban Bilstrup. On the ability of the 802.11p mac method and stdma to support real-time vehicle-to-vehicle communication. *EURASIP J. Wirel. Commun. Netw.*, 2009 :5 :1–5 :13, January 2009.
- [49] A. Soua and H. Afifi. Adaptive data collection protocol using reinforcement learning for vanets. In *2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 1040–1045, July 2013.
- [50] Philippe Flajolet and G. Nigel Martin. Probabilistic counting algorithms for data base applications. *J. Comput. Syst. Sci.*, 31(2) :182–209, September 1985.
- [51] Navneet Kaur and Amarpreet Singh. A survey on data dissemination protocols used in vanets. *International Journal of Computer Applications*, 120(23) :43–50, June 2015.
- [52] Farah El Ali and Bertrand Ducourthial. A light architecture for opportunistic vehicle-to-infrastructure communications. In *Proceedings of the 8th ACM International Workshop on Mobility Management and Wireless Access, MobiWac '10*, pages 60–67, New York, NY, USA, 2010. ACM.
- [53] C. Ide, F. Kurtz, and C. Wietfeld. Cluster-based vehicular data collection for efficient lte machine-type communication. In *2013 IEEE 78th Vehicular Technology Conference (VTC Fall)*, pages 1–5, Sept 2013.
- [54] Z. He, J. Cao, and X. Liu. To carry or to forward ? a traffic-aware data collection protocol in vanets. In *2014 IEEE 11th International Conference on Mobile Ad Hoc and Sensor Systems*, pages 272–276, Oct 2014.
- [55] Zongjian He and Daqiang Zhang. Cost-efficient traffic-aware data collection protocol in {VANET}. *Ad Hoc Networks*, 55 :28 – 39, 2017.
- [56] D. Pacheco-Paramo, L. Iannone, D. Kofman, A. C. Viana, G. Mermoud, and J. P. Vasseur. Adaptation of topology-based routing protocols for data gathering applications in vanets. In *2015 IEEE Vehicular Networking Conference (VNC)*, pages 204–211, Dec 2015.

- [57] V. Naumov and T. R. Gross. Connectivity-aware routing (car) in vehicular ad-hoc networks. In *IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications*, pages 1919–1927, May 2007.
- [58] Anders Brandt, JP Vasseur, Jonathan Hui, Kris Pister, Pascal Thubert, P Levis, Rene Struik, Richard Kelsey, Thomas H. Clausen, and Tim Winter. RPL : IPv6 Routing Protocol for Low-Power and Lossy Networks. RFC 6550, March 2012.
- [59] OMNET++ simulateur. <http://docplayer.fr/13487930-Simulateur-omnet.html>. [Online].
- [60] Fan Bai, Hariharan Krishnan, Varsha Sadekar, Gavin Holland, and Tamer ElBatt. Towards characterizing and classifying communication-based automotive applications from a wireless networking perspective. In *IN PROCEEDINGS OF IEEE WORKSHOP ON AUTOMOTIVE NETWORKING AND APPLICATIONS (AUTONET)*, 2006.
- [61] NS2 simulator. www.isi.edu/nsnam/ns. [Online].
- [62] Sumo, simulation for urban mobility. <http://sourceforge.net/apps/mediawiki/sumo>. [Online].
- [63] D.L. Gerlough and M.J. Huber. *Traffic Flow Theory : A Monograph*. Number n° 165 in Special Report ; No. 165. Transportation Research Board, National Research Council, 1975.
- [64] S. Eichler. Performance evaluation of the ieee 802.11p wave communication standard. In *2007 IEEE 66th Vehicular Technology Conference*, pages 2199–2203, Sept 2007.
- [65] Yi Wang, A. Ahmed, B. Krishnamachari, and K. Psounis. Ieee 802.11p performance evaluation and protocol enhancement. In *2008 IEEE International Conference on Vehicular Electronics and Safety*, pages 317–322, Sept 2008.
- [66] Qi Zhang, Lu Cheng, and Raouf Boutaba. Cloud computing : state-of-the-art and research challenges. *Journal of Internet Services and Applications*, 1(1) :7–18, 2010.
- [67] Google appengine. <http://www.appengine.google.com/>. [Online].
- [68] Roger Jennings. *Cloud Computing with the Windows Azure Platform*. Wrox Press Ltd., Birmingham, UK, UK, 2009.
- [69] Amazon elastic compute cloud (amazon ec2). <http://www.whatis.com/>. [Online].
- [70] Hai Jin, Shadi Ibrahim, Tim Bell, Wei Gao, Dachuan Huang, and Song Wu. *Cloud Types and Services*, pages 335–355. Springer US, Boston, MA, 2010.

- [71] Niroshinie Fernando, Seng W. Loke, and Wenny Rahayu. Mobile cloud computing. *Future Gener. Comput. Syst.*, 29(1) :84–106, January 2013.
- [72] Md Whaiduzzaman, Mehdi Sookhak, Abdullah Gani, and Rajkumar Buyya. A survey on vehicular cloud computing. *Journal of Network and Computer Applications*, 40(0) :325 – 344, 2014.
- [73] Mahmoud Abuelela and Stephan Olariu. Taking vanet to the clouds. In *Proceedings of the 8th International Conference on Advances in Mobile Computing and Multimedia*, MoMM '10, pages 6–13, New York, NY, USA, 2010. ACM.
- [74] Hajar Mousannif, Ismail Khalil, and Stephan Olariu. Cooperation as a service in vanet : Implementation and simulation results. *Mob. Inf. Syst.*, 8(2) :153–172, 2012.
- [75] R. Hussain, S. Junggab, E. Hasoo, K. Sangjin, and O. Heekuck. Rethinking vehicular communications : Merging vanet with cloud computing. In *Proceedings of the 4th International Conference on IEEE Cloud Computing Technology and Science (CloudCom)*, pages 606–609. IEEE, 2012.
- [76] K. Mereshad, H. Artail, , and M. Gerla. We can deliver messages to far vehicles. *IEEE Trans Intell. Transp. Syst.*, 13(3) :1099–1115, Sept 2012.
- [77] George J. Klir and Bo Yuan. *Fuzzy Sets and Fuzzy Logic : Theory and Applications*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1995.
- [78] Christopher J.C.H. Watkins and Peter Dayan. Technical note : Q-learning. *Machine Learning*, 8(3) :279–292, 1992.
- [79] P. Ghazizadeh, S. Olariu, A. G. Zadeh, and S. El-Tawab. Towards fault-tolerant job assignment in vehicular cloud. In *2015 IEEE International Conference on Services Computing*, pages 17–24, June 2015.
- [80] P. Ghazizadeh, R. Florin, A. G. Zadeh, and S. Olariu. Reasoning about mean time to failure in vehicular clouds. *IEEE Transactions on Intelligent Transportation Systems*, 17(3) :751–761, March 2016.
- [81] M. Aloqaily, B. Kantarci, and H. T. Mouftah. Multiagent/multiobjective interaction game system for service provisioning in vehicular cloud. *IEEE Access*, 4 :3153–3168, 2016.
- [82] M. Aloqaily, B. Kantarci, and H. T. Mouftah. An auction-driven multi-objective provisioning framework in a vehicular cloud. In *2015 IEEE Globecom Workshops (GC Wkshps)*, pages 1–6, Dec 2015.

- [83] W. Tonghong Li ; Hazra, S.K. ; Seah. A position-based routing protocol for metropolitan bus networks. In *Proceedings of the 61st International Conference on Vehicular Technology Conference*, pages 2315 – 2319. IEEE, 2005.
- [84] O. K. Tonguz, N. Wisitpongphan, J. S. Parikh, F. Bai, P. Mudalige, and V. K. Sadekar. On the broadcast storm problem in ad hoc wireless networks. In *2006 3rd International Conference on Broadband Communications, Networks and Systems*, pages 1–11, Oct 2006.
- [85] B. Brik, N. Lagraa, Y. Ghamri-Doudane, and A. Lakas. Finding the most adequate public bus in vehicular clouds. In *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)*, pages 67–74, Oct 2016.
- [86] K. Yoon C.L. Hwang. Multiple criteria decision making. lecture notes in economics and mathematical systems., 1981.
- [87] 3gpp specification :evolved universal terrestrial radio access (e-utra) and evolved universal terrestrial radio access network (e-utran). <http://www.3gpp.org/dynareport/36300.htm>.
- [88] Ren-Huang Liou, Yi-Bing Lin, and Shang-Chih Tsai. An investigation on lte mobility management. *Mobile Computing, IEEE Transactions on*, 12(1) :166–176, Jan 2013.
- [89] Simulte. <https://github.com/inet-framework/simulte>.
- [90] B. Brik, J.A Khan, Y. Ghamri-Doudane, N. Lagraa, and A. Lakas. Gss-vc : Game theory-based model for service selection in vehicular cloud. In *Under submission*.
- [91] F. Abdelhafidi, S. Djeradi, B. Brik, N. Lagraa, and A. Lakas. Finding the most suitable service provider in vehicular clouds. In *Under submission*.
- [92] L.A. Zadeh. Fuzzy sets. *Information and Control*, 8(3) :338 – 353, 1965.
- [93] Timothy J. Ross. *Properties of Membership Functions, Fuzzification, and Defuzzification*, pages 89–116. John Wiley Sons, Ltd, 2010.
- [94] Joseph Aguilar-Martin. La logique floue et ses applications industrielles. *Quaderni*, 25(1) :75–86, 1995.
- [95] Nicolas Ragot. *Reconnaissance de formes par modélisation mixte intrinsèque/discriminante à base de systèmes d'inférence floue hiérarchisés*. PhD thesis, Rennes 1, 2003.
- [96] B Meunier. Logique floue, principes, aide à la décision. *Hermès*, 2003.
- [97] John Von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, First edition, 1944.

-
- [98] John Von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, Second edition, 1947.
- [99] John Von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, Third edition, 1953.
- [100] Avinash Dixit. Thomas Schelling's contributions to game theory. *Scandinavian Journal of Economics*, 108(2) :213–229, 2006.
- [101] Ken Binmore. *Game Theory : A Very Short Introduction*. Oxford University Press, 2007.
- [102] T. Alpcan and L. Pavel. Nash equilibrium design and optimization. In *2009 International Conference on Game Theory for Networks*, pages 164–170, May 2009.

Glossaire

Ack Acknowledge.

ADOPEL Adaptive Data cOllECTION Protocol using rEinforcement Learning.

AP Access Point.

ASD Average Service Delay.

BS Base Station.

CA Collection Area.

CaaS Cooperation as a Service.

CALM Continuous Air-interface, Long and Medium range.

CAV Cloud Ad hoc VÃ©hiculaire.

CC Cloud Computing.

CCH Control CHannel.

CCM Cloud Computing Mobile.

CCT Centre du ContrÃ´le de Trafic.

CD Consuming Delay.

CDD Cloud Directory Discovering.

CDGP Clustered Data Gathering Protocol.

CDMA Code Division Multiple Access.

CDS Cloud Directory Service.

CGP Clustered Gathering Protocol.

CH Cluster Head.

CMP Collection Message Propagation.

CROWN disCoveRing and cOnsuming services WithiN vehicular clouds.

CSMA-CA Carrier Sense Multiple Access with Collision Avoidance.

CS Collection Segment.

CTS Clear To Send.

CV Consumer Vehicle.

CVD Consumer Vehicle Density.

DB-VDG Delay-Bounded Vehicular Data Gathering.

DC Direction of the Collection.

DCCS-VC Discovering and Consuming Cloud Services in Vehicular Clouds.

DCF Distributed Coordination Function.

DD Discovering Delay.

-
- DDGP** Distributed Data Gathering Protocol.
- DSRC** Dedicated Short Range Communication.
- DT** Data Type.
- DTNs** Delay Tolerant Networks.
- D-SDMA** Dynamic Space Division Multiple Access.
- ECDGP** Extended Clustered Data Gathering Protocol.
- ETSI** European Telecommunications Standards Institute.
- E2ED** End to End Delay.
- FCC** Federal Communications Commission.
- FcVcA** Fuzzy clustering-based Vehicular cloud Architecture.
- FDCCS-VC** Fast Discovering and Consuming Cloud Services in Vehicular Cloud.
- GI** Guard Interval.
- GPS** Global Positioning System.
- GSS-VC** Game theory-based approach for Service Selection in Vehicular Cloud.
- GTC** Grid-based Tracking Cell.
- IaaS** Infrastructure as a Service.
- IEEE** Institute of Electrical and Electronics Engineers.
- INaaS** INformation as a Service.

-
- IP** Internet Protocol.
- ISO** International Organization for Standardization.
- ITS** Intelligent Transportation System.
- I2V** Infrastructure to Vehicle.
- LLC** Logical Link Control.
- LTE-A** Long Term Evolution-Advanced.
- MAC** Medium Access Control.
- MANet** Mobile Ad hoc Network.
- MD** More Data.
- MME** Mobility Management Entity.
- MV** Member Vehicle.
- NaaS** Network as a Service.
- NAV** Network Allocation Vector.
- NS2** Network Simulator 2.
- OSI** Open System Interconnection.
- OMS** Organisation Mondiale de la Santé©.
- OMNet++** Object Modular Network testbed in C++.
- PaaS** Platform as a Service.

PC Probability Convergence.

PHY PHYSical layer.

PM Passerelle Mobile.

PV Provider Vehicle.

P-GW Packet Gateway.

QD Queing Delay.

QoS Quality of Service.

RCS-VC Renting out and Consuming Services in Vehicular Clouds.

RIDE Real-time traffic Information aware Data collEction.

RR Request for Ressources.

RRP Request for Ressources Packet.

RS Request for Service.

RSB Road Side Backbone.

RSC Road Side Collectors.

RSP Request for Service Packet.

RSU Road Side Units.

RTS Request To Send.

RV Road Vehicles.

RVC Road Vehicle Collector.

RVS Road Vehicle Sink.

SaaS Software as a Service.

SAW Simple Additive Weighting.

SCH Service CHannel.

SDMA Space Division Multiple Access.

SER Successful Execution Ratio.

SG Service Gain.

SR Service Registration.

SRP Service Registration Packet.

SS Silence Segment.

STaaS SStorage as a Service.

S-GW Serving Gateway.

TAL Tracking Area List.

TBP Tracking Bus Path.

TBPI TBP Identity.

TCDGP Token-based Clustered Data Gathering Protocol.

TCP Transmission Control Protocol.

TDMA Time Division Multiple Access.

TD-SDMA Token-based Dynamic Space Division Multiple Access.

TIC Technologies de l'Information et de la Communication.

UDP User Datagram Protocol.

UE User Equipment.

VANet Vehicular Ad hoc Network.

VC Vehicule Consommateur.

VCPS Vehicular Cyber-Physical System.

VCR Vehicular Cloud for Road-side scenarios.

VD Vehicle Density.

VF Vehicule Fournisseur.

VL-CSMA Vehicular Location-based Carrier Sense Multiple Access.

VSD Vehicular Services Directory.

VSN Vehicular Sensor Networks.

VT Validating Time.

VuC Véhicules utilisant le Cloud computing.

V2I Vehicle to Infrastructure.

V2V Vehicle to Vehicle.

WAVE Wireless Ability in Vehicular Environnements.

WG Work Group.

Wi-Fi Wireless Fidelity.

WMN Wireless Mesh Networks.

WSMP Wave Short Message Protocol.

WSN Wireless Sensor Networks.