

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Abderrahmane Mira de Béjaïa



Faculté des Sciences Exactes

Département d'Informatique

Mémoire de Master Professionnel (ASR)

Thème

**Évaluation des performances des protocoles
de routage DSR et AODV pour les réseaux
véhiculaires en milieu urbain et autoroutier**

Présenté par :

HOUARI Brahim ZIREM Nassim

Soutenu le 30 juin 2016 devant le jury composé de :

Présidente : Mme OUYAHIA Samira Maître de Conférences A (UAMB)
Examinatrice : Mme TASSOULT Nadia Maître de Conférences A (UAMB)
Encadreuse : Mme ZIDANI Ferroudja Maître de Conférences A (UAMB)

Promotion 2015 – 2016

Remerciements

Grâce à Dieux vers lequel vont toutes les louanges, ce travail c'est accompli.

Au terme de ce travail, nous tenons à exprimer nos gratitude et nos remerciements pour toutes les personnes qui ont contribué à sa réalisation.

Nous tenons à exprimer tout d'abord nos vifs remerciements pour notre encadreuse Mme **ZIDANI Ferroudja**, de nous avoir encadrés pour réaliser ce travail par la disponibilité de son temps précieux, ses encouragements, ses aides, ses conseils avisés, sa gentillesse...

Nous tenons, également, à remercier vivement les membres du jury pour l'honneur qu'ils nous ont fait en acceptant de juger ce travail. Nos remerciements vont aussi à tous nos enseignants et toutes les personnes qui nous ont soutenus jusqu'au bout, et qui n'ont pas cessé de nous donner des conseils très importants en signe de reconnaissance.

Dédicaces

A nos parents, pour leurs sacrifices déployés à notre égard, pour leurs patiences, leurs amours et leurs confiances. Qu'ils trouvent dans ce modeste travail, le témoignage de notre profonde affection et de notre attachement indéfectible ; nulle dédicace ne puisse exprimer ce qu'on leur doit.

A nos frères et sœurs et tous nos amis pour chaque mot reçu, chaque geste d'amitié, à chaque main tendue et pour toute attention témoignée.

Table des Matières

Table des Matières	i
Table des Figures	vi
Liste des Tableaux	viii
Liste des Abréviations	ix
Introduction générale	1
1 LES RESEAUX VANETs	3
Introduction	4
1.1 Les réseaux Ad Hoc	4
1.1.1 Définition des réseaux Ad Hoc	4
1.1.2 Modes de communication dans les réseaux mobile Ad Hoc	5
1.1.2.1 Le mode Unicast	5
1.1.2.2 Le mode Multicast (multipoint)	5
1.1.2.3 Le mode Broadcast (la diffusion)	6
1.1.3 Caractéristiques des réseaux Ad Hoc	6
1.1.3.1 Absence d'infrastructure	6
1.1.3.2 Routage par relais	7
1.1.3.3 Topologie dynamique	7
1.1.3.4 La taille du réseau	8
1.1.3.5 L'hétérogénéité des nœuds	8

1.1.3.6	La taille des réseaux Ad Hoc	8
1.1.3.7	Multi-sauts	9
1.1.4	Les contraintes liées aux réseaux Ad Hoc[3]	9
1.2	Les Réseaux Véhiculaires Ad Hoc	13
1.2.1	Définition d'un réseau VANET	13
1.2.2	Nœuds d'un réseau VANET	14
1.2.3	Technologies utilisées dans la communication véhiculaire[3]	14
1.2.3.1	Communication de véhicule à véhicule	14
1.2.3.2	Communication de véhicule avec utilisation d'infrastructures	15
1.2.3.3	Communication Hybride	16
1.2.4	Caractéristiques des réseaux VANETs	16
1.2.4.1	La Collecte d'informations et la perception de l'environnement proche	17
1.2.4.2	Capacité de traitement, d'énergie et de communication	17
1.2.4.3	Environnement de déplacement et modèle de mobilité	17
1.2.4.4	Forte mobilité, topologie du réseau et connectivité	17
1.2.4.5	Type d'information transportée et diffusée	18
1.2.5	Application des réseaux VANETs	18
1.2.5.1	Application dans la sécurité routière	18
1.2.5.2	Application pour l'optimisation du trafic et aide dans la conduite	18
1.2.5.3	Application au confort du conducteur et des passagers	19
1.2.6	Standardisation et normalisation dans un VANET	19
1.2.6.1	DSRC (Dedicated Short Range Communications)	19
1.2.6.2	WAVE (Wireless Access in Vehicular Environments)	19
1.2.6.3	La norme IEEE 802.11p	19
1.2.7	Les défis	20
1.2.7.1	Sécurité	20
1.2.7.2	Qualité de service	20
1.2.7.3	Canal radio fiable	20
1.2.7.4	Routage	20
1.2.7.5	L'accès au canal	21

1.2.7.6	Localisation des véhicules	21
1.2.7.7	Problèmes de congestion	21
1.2.7.8	Mobilité dans la simulation des réseaux	21
Conclusion		21
2	LE ROUTAGE DANS LES RÉSEAUX VÉHICULAIRES	23
Introduction		24
2.1	Routage dans les réseaux mobiles Ad Hoc	24
2.2	Problématique de routage dans les réseaux mobiles Ad Hoc	24
2.3	Les objectifs des protocoles de routage dans les réseaux mobiles Ad Hoc	25
2.4	Classification des protocoles de routage dans les réseaux mobiles Ad Hoc	26
2.4.1	Classification selon l'architecture	26
2.4.1.1	Les protocoles uniformes	26
2.4.1.2	Les protocoles non uniformes	26
2.4.2	Classification selon l'approche de routage	26
2.4.2.1	les protocoles proactifs	26
2.4.2.2	Les protocoles réactifs	27
2.4.2.3	Les protocoles hybrides	27
2.4.3	Classification selon le type d'algorithme utilisé	29
2.4.3.1	Les protocoles de routage à vecteur de distance (Distance Vector Protocols)	29
2.4.3.2	Les protocoles de routage à état des liens (Link State Protocols)	29
2.4.3.3	Le protocole de routage source	30
2.4.3.4	Le protocole de routage par apprentissage en arrière	30
2.5	Protocoles de routage dans les réseaux VANET	31
2.5.1	Classification des protocoles de routage dans les réseaux VANET	31
2.5.1.1	Les protocoles de routage basés sur la topologie	31
2.5.1.2	Les protocoles de routage basés sur la géographique	32
2.5.2	Protocoles de routage utilisés dans notre étude	33
2.5.2.1	Le protocole DSR	33
2.5.2.2	Le protocole AODV	33

Conclusion	34
3 SIMULATION ET DISCUSSION DES RESULTATS	35
Introduction	36
3.1 Simulation dans les VANETs	36
3.1.1 Simulation de la mobilité dans les VANETs	36
3.2 Outils de simulation	37
3.2.1 Objectif de la simulation	37
3.2.2 Les simulateurs Réseaux	37
3.2.2.1 OMNeT++	38
3.2.2.2 Modèle de simulation	41
3.2.2.3 Le simulateur de mobilité SUMO	44
3.3 Environnement de travail	44
3.3.0.4 Environnement matériel	44
3.3.0.5 Environnement logiciel	44
3.4 Etapes de la simulation	45
3.4.1 Préparation des fichiers avant la simulation	50
3.4.2 Métriques de simulation	55
3.4.2.1 Le débit	55
3.4.2.2 Le taux de paquet perdu	55
3.4.3 paramètres de simulation	56
3.5 Résultats de la simulation et discussion	56
3.5.1 Premier cas de simulation (milieu urbain)	56
3.5.1.1 Le débit avec le protocole DSR	58
3.5.2 Deuxième cas de simulation (milieu autoroutier)	59
3.5.2.1 Le débit avec le protocole AODV	60
3.5.2.2 Le débit avec le protocole DSR	60
Conclusion	61
Conclusion Générale et Perspectives	62

A Annexe	64
A.1 Installation de OMNeT++4.6 sous Windows 7	65
A.2 Importation d'un nouveau projet sous OMNeT++	66
A.3 Configuration de la variable d'environnement pour SUMO	67

Table des figures

1.1	Exemple de réseau Ad hoc	5
1.2	Les différents modes de communication	6
1.3	Portée d'un nœud[7]	7
1.4	Topologie dynamique des réseaux Ad Hoc	8
1.5	Influence des puissances d'émission sur les liens de communications	10
1.6	Nœud caché	11
1.7	Nœud exposé[7]	12
1.8	Un exemple des réseaux VANETs[7]	13
1.9	Les éléments constituant le véhicule intelligent[9]	14
1.10	Communication véhicule à véhicule	15
1.11	Communication véhicule à station de base	16
1.12	communication hybride	16
2.1	Protocoles de routage dans les réseaux VANET[34]	32
2.2	Mécanisme de routage AODV	34
3.1	Architecture modulaire du simulateur OMNeT++	39
3.2	Interface d'OMNeT++	41
3.3	Architecture de VEINS[41]	43
3.4	Zone d'étude de la ville de Bejaia dans OpenStreetMap	45
3.5	Génération de bejaia.net.xml	46
3.6	Copie du contenu de fichier web	47
3.7	Enregistrement du fichier typemap.xml	47

3.8	Génération de bejaia.poly.xml	48
3.9	Génération de bejaia.rou.xml	49
3.10	Génération des routes aléatoires dans SUMO	50
3.11	Modification du fichier erlangen.sumo.cfg	51
3.12	Modification du fichier erlangen.lauchd.xml	51
3.13	Ouverture et écoute sur le port TCP 9999	52
3.14	Modification du fichier omnetpp.ini	53
3.15	Fenêtre de la simulation de OMNet++	53
3.16	Fenêtre de la simulation de SUMO	54
3.17	Démarrage de la simulation sur le terminal	54
3.18	Résultats de la simulation	55
3.19	Taux de paquets perdus en milieu urbain avec le protocole AODV	57
3.20	Taux de paquets perdus en milieu urbain avec le protocole DSR	57
3.21	Débit en milieu urbain avec le protocole AODV	58
3.22	Débit en milieu urbain avec le protocole DSR	58
3.23	Taux de paquets perdus en milieu autoroutier avec le protocole AODV	59
3.24	Taux de paquets perdus en milieu autoroutier avec le protocole DSR	59
3.25	Débit en milieu autoroutier avec le protocole AODV	60
3.26	Débit en milieu autoroutier avec le protocole DSR	60
A.1	Exécution de la commande . setenv	65
A.2	Exécution de la commande ./configure de OMNeT++	65
A.3	Exécution de la commande make de OMNeT++	66
A.4	Première étape d'importation d'un nouveau projet sous OMNeT++	66
A.5	Deuxième étape d'importation d'un nouveau projet sous OMNeT++	67
A.6	Ajout de la variable PATH de SUMO	67

Liste des tableaux

2.1	Comparaison entre protocoles proactifs et protocoles réactifs	28
3.1	Les composants d'OMNeT++	40
3.2	Tableau des protocoles supportés par INET	42
3.3	Scénarios de mouvement utilisés.	56

Liste des Abréviations

AODV Ad hoc On-Demand Vector routing

API Application Programming Interface

CPU Central Processus Unit

CSMA/CA Carrier Sense Multiple Access with Collision Avoidance

DREAM Distance Routing Effect Algorithm for Mobility

DSDV Destination Sequenced Distance Vector

DSR Dynamic Source Routing

ELB ACTD Extending the Littoral Battle-space Advanced Concept Technology Demonstration

FCC Federal Communications Commission

GPS Global Positioning System

GPSR Greedy Perimeter Stateless Routing

GSR Geographic Source Routing

IEEE Institute of Electrical and Electronics Engineers

INET Institut National des Études Territoriales

IT Intelligent Transportation

ITSA	Intelligent Transportation Society of America
IVC	Inter-Vehicle Communications
LAR	Location-Aided Routing
LRSQ	Link Quality Source Routing
MAC	Media Access Control
MANET	Mobile Ad hoc Networks
NS	Network Simulator
OLSR	Optimized Link State Routing
PHY	Physical layer
PRNET	Packet Radio Network
QOS	Quality Of Service
RERR	Route Error
RREP	Routing Reply
RREQ	Route Request
RSU	Road Side Units
SN	Sequence Number
STI	Society of Technology Industrials
SUMO	Simulation of Urban MObility
TCP/IP	Transmission Control Protocol/Internet Protocol
TGP	Task Group
TORA	Temporally Ordered Routing Algorithm
UMB	Urban Multi-Hop Broadcast

V2I Vehicle to Infrastructure

V2V Vehicle to Vehicle

VANET Vehicular Ad hoc Networks

WAVE Wireless Access in Vehicular Environment

WIFI Wireless Fidelity

ZRP Zone Routing Protocol

Introduction Générale

Les gouvernements investissent massivement dans la prévention routière pour les véhicules et les routes de demain, leurs objectifs sont de réduire le nombre d'accidents sur la route et sauver des vies. Les réseaux VANETs sont une solution à ce problème.

Actuellement, nos véhicules sont électronisés avec des ordinateurs de bord. Ils traitent des informations telles que : la température interne et externe, le GPS (Global Positioning System), le système de frein à main, etc. Malheureusement, ces informations sont confinées dans chaque véhicule et aucun système de communication d'information n'était présent pour diffuser ces informations vitales aux autres véhicules. Les communications sans fil peuvent résoudre ce problème en fournissant des applications de confort aux usagers et des applications de sécurité tels l'optimisation du trafic routier ou l'avertissement d'un risque de verglas.

Les communications sans fil 3G, Wifi et Bluetooth, sont largement utilisées de nos jours. Leurs coûts sont abordables pour des débits et des portées de transmissions de plus en plus élevés. De l'émergence de ces technologies sans fil sont nés les réseaux MANET (Mobile Ad hoc NETWORK) connectant les ordinateurs entre eux et par dérivations, les réseaux VANETs (Vehicular Ad hoc NETWORK) connectant les véhicules autoroutiers.

Contrairement aux MANET, les VANETs sont soumis à : une topologie hautement dynamique, une forte mobilité des nœuds, une connectivité changeante, etc. Les autres types de réseaux sans fil s'accordent sur certains critères de stabilité au niveau topologique pour avoir des performances maximales. Néanmoins, la communication dans les VANETs se fait en temps réel, avec des véhicules pouvant aller à plus de 100 km/h (sur autoroute et en fonction des

pays). La topologie est hautement dynamique. Des pertes de connectivités sont à prévoir et les connexions non fiables doivent être sécurisées par des protocoles.

Notre projet consiste à Évaluer les performances des protocoles de routage DSR et AODV pour les réseaux véhiculaires dans deux milieux différents urbain et autoroutier, en terme de taux de paquet perdu et de débit, on utilisons les deux simulateurs OMNeT++ et SUMO.

Notre étude s'étale sur trois chapitres, dans le premier chapitre nous commençons dans un premier temps par définir les réseaux Ad Hoc, ces différents modes de communication ainsi que ces caractéristiques, les contraintes qui lui sont liées, puis nous abordons les réseaux Ad Hoc véhiculaires (VANET) en décrivant les entités communicantes, les modes de communication et les caractéristiques de ces réseaux, on cite aussi quelques applications conçu pour les VANETs avant de présenter les standards de communication véhiculaire. Au final nous parlons des différents défis qui ont un impact sur le futur déploiement des réseaux véhiculaires.

Dans le deuxième chapitre, nous traitons le routage dans les réseaux mobiles Ad Hoc de manière générale, la classification des protocoles selon des différents critères. Ensuite, nous passons au routage et la classification des protocoles de routage dans les réseaux véhiculaires.

Dans le troisième et dernier chapitre on commence par l'explication du principe de la simulation dans les VANETs. Nous décrivons ainsi, l'environnement de travail, les outils, les étapes suivis, quelques métriques et paramètres principaux utilisés dans notre simulation. Nous finissons par une discussion sur l'évaluation des performances des protocoles de routage DSR et AODV dans les réseaux véhiculaires en milieu urbain et autoroutier.

1

LES RESEAUX VANETS

Introduction

Avec l'adoption des technologies de communication sans fil, les réseaux ont connus ces dernières années un essor spectaculaire et s'imposent aujourd'hui de façon indéniable. Parmi les technologies récentes de communication sans fil on trouve les réseaux véhiculaires (VANETs), très inspiré des MANET (réseau mobile Ad Hoc), VANET permet aux véhicules de communiquer via des messages d'alertes de sécurité envoyés entre eux.

Ces réseaux véhiculaires sont passés du stade de simple curiosité pour revêtir aujourd'hui un intérêt certain aussi bien du point de vue de l'industrie automobile que des opérateurs de réseaux et services et de la communauté de recherche. Ces réseaux véhiculaires sont en effet une classe émergente de réseaux sans fil permettant des échanges de données entre véhicules ou encore entre véhicules et infrastructure. Ils ont été amplement étudiés dans le but de fournir de nouvelles technologies capables d'améliorer la sécurité et l'efficacité des transports routiers.

Dans ce chapitre nous commençons dans un premier temps par définir les réseaux Ad Hoc, ces différents modes de communication ainsi que ces caractéristiques, les contraintes qui lui sont liées, puis nous abordons les réseaux Ad Hoc véhiculaires (VANETs) en décrivant les entités communicantes, les modes de communication et les caractéristiques de ces réseaux, on cite aussi quelques applications pour VANET avant de présenter les standards de communication véhiculaire. Puis, nous parlons sur les différents défis qui ont un impact sur le futur déploiement des réseaux véhiculaires et au final, on expose les modèles de propagation utilisé dans les VANETs.

1.1 Les réseaux Ad Hoc

1.1.1 Définition des réseaux Ad Hoc

Un réseau mobile Ad Hoc, appelé généralement MANET (Mobile Ad Hoc NETWORK), est un réseau sans fil qui consiste en une grande population, relativement dense, d'unités mobiles qui se déplacent dans un territoire quelconque et dont le seul moyen de communication est l'utilisation des interfaces sans fil généralement le médium radio, sans l'aide d'une infrastructure préexistante ou administration centralisée, ces unités mobiles jouent à la fois le rôle de

terminaux et de routeurs pour permettre le passage de l'information entre elles[1].

Il permet donc à deux nœuds qui sont chacun à portée des ondes l'un de l'autre (condition appropriée de propagation d'ondes radio) de rentrer en communication directement.

Un réseau Ad Hoc doit être facilement déployé, les nœuds peuvent joindre ou quitter le réseau de manière totalement dynamique sans informer le réseau, et si possible sans effet de bord sur les communications des autres membres.

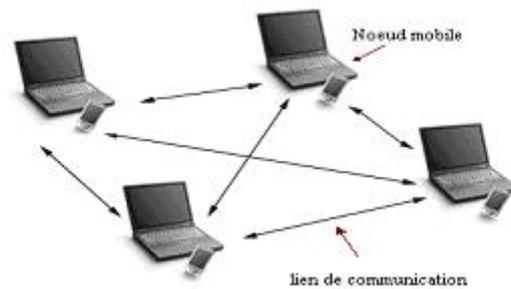


FIGURE 1.1 – Exemple de réseau Ad hoc

1.1.2 Modes de communication dans les réseaux mobile Ad Hoc

Les échanges de données dans les réseaux mobiles utilisent les modes de communication suivants[2] :

1.1.2.1 Le mode Unicast

Le terme unicast définit une connexion réseau point à point. On entend par unicast, le fait de communiquer entre deux ordinateurs identifiés chacun par une adresse réseau unique. Les paquets de données sont routés sur le réseau suivant l'adresse du destinataire, seul le destinataire intercepte et décode le paquet qui lui est adressé.

1.1.2.2 Le mode Multicast (multipoint)

On entend par multicast, le fait de communiquer simultanément avec un groupe d'ordinateurs identifié par une adresse spécifique (adresse de groupe). Son avantage par rapport au

mode classique unicast devient évident quand on veut diffuser de la vidéo. Les paquets de données sont routés sur le réseau selon l'adresse des destinataires en-capsulée dans la trame transmise. Seuls les destinataires interceptent et décodent les paquets qui leurs sont adressés.

1.1.2.3 Le mode Broadcast (la diffusion)

Le broadcast est un terme anglais définissant une diffusion de données depuis une source unique à un ensemble de récepteurs. Contrairement à une communication Point à Point, il est possible d'adresser des paquets de données à un ensemble de machines d'un même réseau uniquement par des adresses spécifiques qui seront interceptées par toutes les machines du réseau ou sous réseau. La figure 1.2 illustre les différents modes de communication.

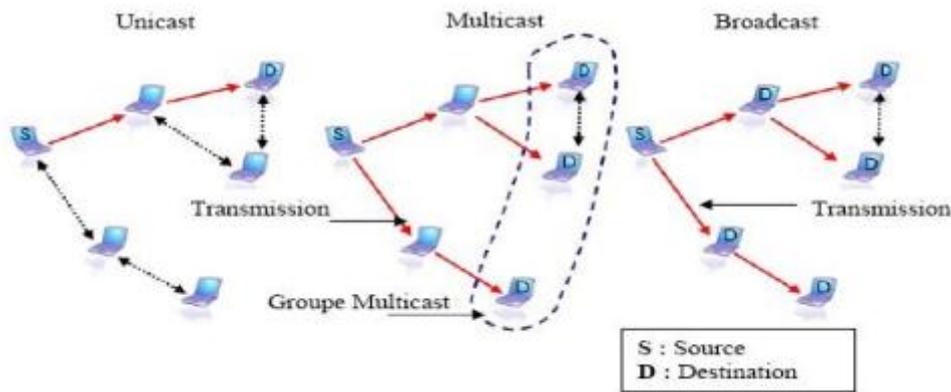


FIGURE 1.2 – Les différents modes de communication

1.1.3 Caractéristiques des réseaux Ad Hoc

Les réseaux mobiles Ad Hoc présentent plusieurs caractéristiques, à savoir[3][4][5] :

1.1.3.1 Absence d'infrastructure

Les nœuds d'un réseau Ad Hoc travaillent dans un environnement pair à pair totalement distribué, ce qui leur permet de se déplacer librement. Ces nœuds agissent en tant que routeurs pour relayer des communications ou générer leurs propres données.

1.1.3.2 Routage par relais

Dans un réseau Ad Hoc, un terminal peut communiquer directement avec les terminaux à sa portée (ses voisins).

Lorsqu'une machine veut communiquer avec une autre se trouvant hors de sa portée, chaque nœud actif du réseau sert de routeur pour ses voisins.

Dans l'exemple ci-dessus, la machine A dont la portée est schématisée par le cercle orange, veut communiquer avec la machine C se trouvant hors de son champ de réception. Pour aboutir, la connexion réseau va donc utiliser la machine B se trouvant à portée de réception des machines A et C.

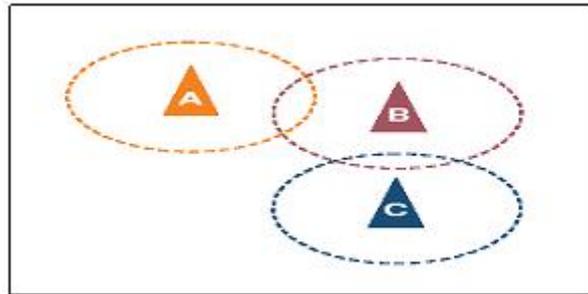


FIGURE 1.3 – Portée d'un nœud[7]

1.1.3.3 Topologie dynamique

Une particularité très importante qui distingue les réseaux mobiles Ad Hoc des réseaux filaires est la mobilité de ses nœuds. Les nœuds sont libres de se déplacer arbitrairement, des routes peuvent se créer et disparaître très souvent, ce qui provoque des changements fréquents dans la topologie du réseau. Ces modifications doivent être prises en compte par le protocole de routage. Cette caractéristique rend la topologie de ce type du réseau sans fil très dynamique. La figure 1.4 illustre la topologie dynamique des réseaux Ad Hoc.

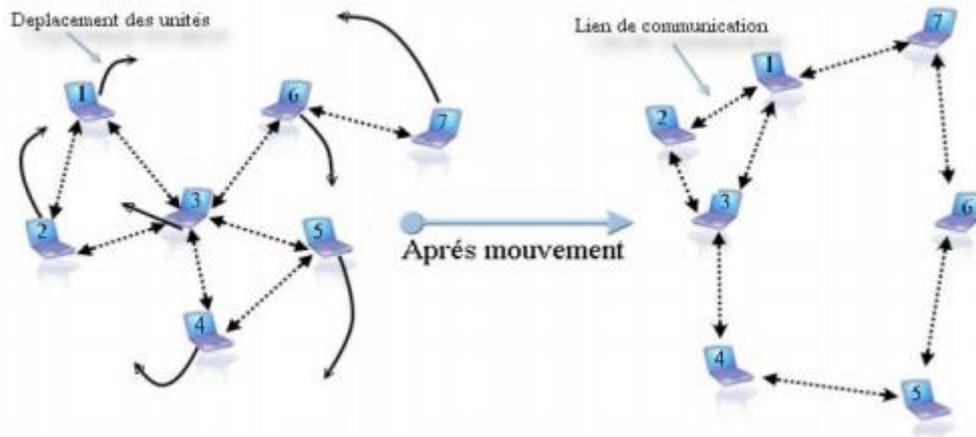


FIGURE 1.4 – Topologie dynamique des réseaux Ad Hoc

1.1.3.4 La taille du réseau

Étant donné les avancées importantes réalisées dans le domaine des communications sans fil et les bas coûts des équipements associés, les véhicules qui intègrent déjà massivement des systèmes GPS et des équipements Bluetooth, seront très probablement équipés et ce, tout aussi massivement, de plateformes de communication leur permettant de constituer de véritables réseaux. Ce faisant, et compte tenu de l'importance sans cesse grandissante de la densité et du parc des véhicules, on peut s'attendre à ce que la taille des réseaux véhiculaires dont les déploiements restent encore très confidentiels, soit d'une tout autre ampleur.

1.1.3.5 L'hétérogénéité des nœuds

Un nœud mobile peut être équipé d'une ou plusieurs interfaces radio ayant des capacités de transmission variées et opérant dans des plages de fréquences différentes. Cette hétérogénéité de capacité peut engendrer des liens asymétriques dans le réseau. De plus, les nœuds peuvent avoir des différences en terme de capacité de traitement (CPU, mémoire), de logiciel, de taille (petit, grand) et de mobilité (lent, rapide). Dans ce cas, une adaptation dynamique des protocoles s'avère nécessaire pour supporter de telles situations.

1.1.3.6 La taille des réseaux Ad Hoc

Elle est souvent de petite ou moyenne taille (une centaine de nœuds) ; le réseau est utilisé pour étendre temporairement un réseau filaire, comme pour une conférence ou des situations

où le déploiement du réseau fixe n'est pas approprié. Cependant, certaines applications des réseaux Ad Hoc peuvent nécessiter une utilisation allant jusqu'à des dizaines de milliers de nœuds.

1.1.3.7 Multi-sauts

Les réseaux Ad Hoc utilisent souvent des sauts multiples pour éviter les obstacles, minimiser la consommation d'énergie ou pour joindre un nœud qui n'est pas dans la portée de communication de l'émetteur.

1.1.4 Les contraintes liées aux réseaux Ad Hoc[3]

- **Bande passante limitée**

La communication dans les réseaux Ad Hoc se base sur le partage d'un médium de transmission (les ondes radio), ce qui induit que la bande passante réservée pour un hôte soit modeste.

- **Contrainte d'énergie**

Les nœuds mobiles sont destinés à être portables et mobiles et donc à être alimentés par des sources d'énergie autonomes comme les batteries ou les autres sources consommables (exemple d'énergie solaire). La durée de vie des batteries est limitée, ce qui se répercute par conséquent sur les services et les applications supportées par chaque nœud. Le paramètre d'énergie doit être pris en considération dans tout le contrôle fait par le système.

- **Liens unidirectionnels**

Un lien unidirectionnel représente la possibilité qu'un nœud soit entendu (à portée d'ondes) d'un autre mais pas l'inverse. Cela arrive notamment lorsque les puissances d'émission sont différentes suivant les émetteurs (mode de propagation des ondes). Ce qui pose un problème des acquittements que l'on ne peut pas envoyer.

La figure 1.5 représente l'influence des puissances d'émission sur les liens de communication. Les nœuds B et C sont totalement à portée du nœud A, alors que le nœud D ne l'est pas. De plus, nous pouvons remarquer que les portées de communication des nœuds A, B et C se chevauchent, alors que le nœud D est à la portée de B seulement, et B n'est pas à la portée de D.

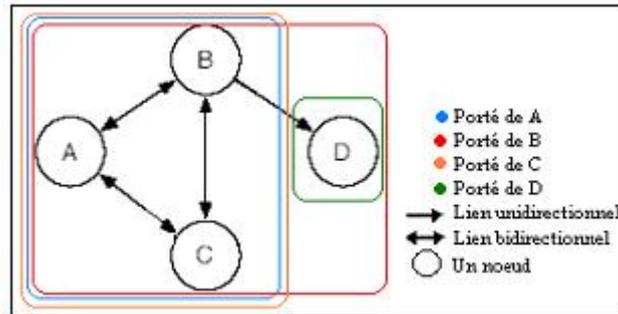


FIGURE 1.5 – Influence des puissances d'émission sur les liens de communications

- **Sécurité physique limitée**

Les réseaux mobiles Ad Hoc sont très sensibles aux attaques extérieures par rapport aux réseaux filaires classiques. Les données envoyées transitent par des équipements d'utilisateurs inconnus ce qui pose un problème de confidentialité et nécessite l'utilisation d'outils de cryptage et de sécurisation des données.

- **Interface radio multiple**

Lorsqu'un nœud possède plusieurs interfaces radio, les protocoles deviennent tout de suite beaucoup plus complexes, car chaque interface possède sa propre zone de couverture.

- **Changement de topologie**

La mobilité des nœuds provoque des changements fréquents dans la topologie du réseau. La réactivité du dispositif mis en place face à ces modifications a un impact direct sur la qualité du service proposé à l'utilisateur.

- **Interférences**

Les liens radio ne sont pas isolés, ce qui fait que le taux d'erreur de transmission dans les réseaux radio est plus élevé que dans les réseaux filaires. Cela est dû généralement aux problèmes d'interférences qui peuvent être de natures diverses, à savoir :

- Le nombre limité de canaux disponibles.
- Les fréquences d'émissions sont proches, ainsi, les émetteurs travaillant à des fréquences proches peuvent interférer entre eux.
- Les bruits produits par l'environnement (certains équipements électriques, certains moteurs...).
- Les phénomènes d'atténuation, réflexion et de chemins multiples qui rendent le signal incompréhensible en le déformant.

- **Nœud caché**

Ce phénomène est très particulier à l'environnement sans fil. Un exemple est illustré dans la figure 1.6 Dans cet exemple, les nœuds B et C ne s'entendent pas, à cause d'un obstacle qui empêche la propagation des ondes. Les mécanismes d'accès au canal vont permettre alors à ces nœuds de commencer leurs émissions simultanément. Ce qui provoque des collisions au niveau du nœud A.

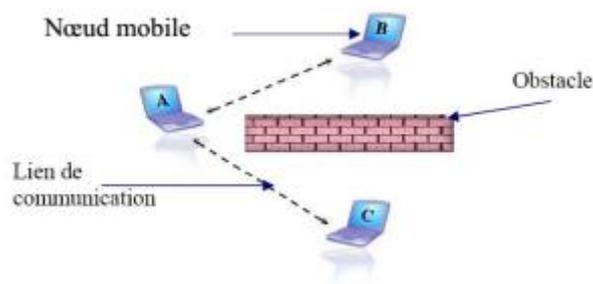


FIGURE 1.6 – Nœud caché

- **Le nœud exposé[7]**

Ce problème survient quand un nœud veut établir une transmission avec un deuxième, mais doit la retarder car il y a une transmission en cours entre deux autres nœuds se trouvant dans son voisinage[6]. La figure 1.7 décrit un scénario typique. Supposons que les stations A et C peuvent entendre les transmissions de B, mais que A n'entend pas C (et vice-versa).

Supposons aussi que B est entrain d'envoyer des données vers A et que, au même moment, C veut communiquer avec D. En suivant la logique CSMA/CA, le nœud C va commencer par déterminer si le support est libre. Á cause de la communication entre B et A, C trouve le support occupé et il retarde son envoi bien que celui-ci n'aurait pas causé de collisions.



FIGURE 1.7 – Nœud exposé[7]

- **Configuration et adressage**

Le schéma d'adressage à adopter et le mécanisme de configuration qui permet à ces adresses d'être assignées aux diverses machines dans le réseau sont deux autres problèmes liés aux réseaux Ad Hoc.

- **La mobilité des nœuds et maintenance des routes**

La mobilité continue des nœuds crée un changement dynamique de topologie. Par exemple, un nœud peut rejoindre le réseau, changer de position ou quitter le réseau. Ce déplacement a naturellement un impact sur la morphologie du réseau et peut modifier le comportement du canal de communication. Ajoutons à cela, la nature des communications (longues et synchrones,

courtes et asynchrones,...). Les algorithmes de routage doivent ainsi résoudre ces problèmes et supporter la maintenance et prendre en charge en un temps limité la reconstruction des routes tout en minimisant l'overhead généré par les messages de contrôle.

1.2 Les Réseaux Véhiculaires Ad Hoc

1.2.1 Définition d'un réseau VANET

Un réseau VANET est un réseau de communication entre véhicules intelligents équipés de calculateurs, de périphériques réseau et de différents types de capteurs[8]. Les VANETs font parti de la famille des réseaux mobiles MANET qui fonctionnent dans des réseaux à liaison point à point sans infrastructure, c'est-à-dire que tout nœud constituant le réseau est un point d'accès. Dans un réseau VANET les nœuds sont les véhicules intelligents appartenant au réseau. Ils peuvent communiquer directement entre eux (V2V : Vehicle to Vehicle) afin d'échanger des informations sur le trafic par exemple ou avec des stations de bases placées tout au long des routes (accès à internet...) (V2I : Vehicle to Infrastructure). L'objectif principale des réseaux VANETs est d'améliorer la sécurité routière tout en élaborant des routes plus sûres et efficaces en fournissant des informations opportunes aux conducteurs. Un exemple de réseau VANET urbain est illustré dans la Figure 1.8.



FIGURE 1.8 – Un exemple des réseaux VANETs[7]

1.2.2 Nœuds d'un réseau VANET

Un nœud d'un réseau VANET est un véhicule équipé de terminaux tels que les calculateurs, les interfaces réseaux ainsi que des capteurs capables de collecter les informations et de les traiter. On parle de la notion de « véhicule intelligent ». La Figure 1.9 modélise un véhicule intelligent.

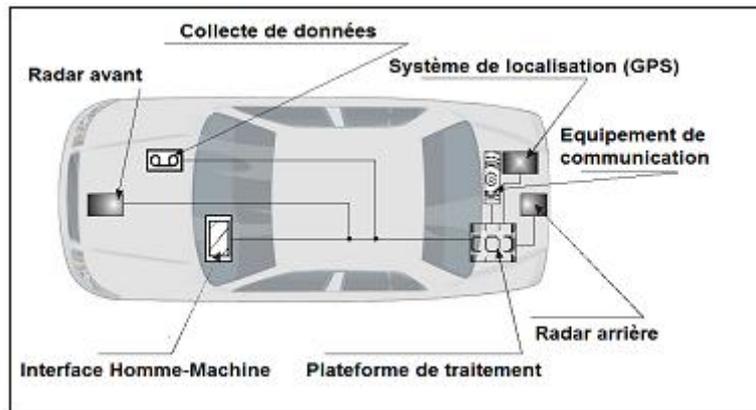


FIGURE 1.9 – Les éléments constituant le véhicule intelligent[9]

1.2.3 Technologies utilisées dans la communication véhiculaire[3]

Les réseaux véhiculaires par analogie à ce qui existe dans les réseaux sans fil peuvent être déployés suivant trois catégories :

1.2.3.1 Communication de véhicule à véhicule

Dans cette catégorie, un réseau de véhicule est vu comme un cas particulier du réseau MANET (Mobile Ad Hoc Network) où les contraintes d'énergie, de mémoire et de capacité sont relaxées et où le modèle de mobilité n'est pas aléatoire mais prévisible avec une très grande mobilité. Cette architecture peut être utilisée dans le scénario de diffusion d'alertes (freinage d'urgence, collision, ralentissement...) ou pour la conduite coopérative.

Aucune infrastructure n'est utilisée, aucune installation n'est nécessaire sur les routes et tous les véhicules sont équipés pour communiquer directement entre eux n'importe où, que ce soit sur les autoroutes, des routes de montagnes ou des routes urbaines, ce qui donne une communication moins coûteuse et plus flexible.

Cette approche souffre de certains inconvénients dont nous citons :

- Les délais de communication qui sont élevés, étant donné que la communication se fait en utilisant le multiple sauts.
- Les déconnexions fréquentes dues au fait que les véhicules sont mobiles.
- La sécurité réseau est très limitée.

La figure 1.10 montre un exemple de communication de véhicule à véhicule.

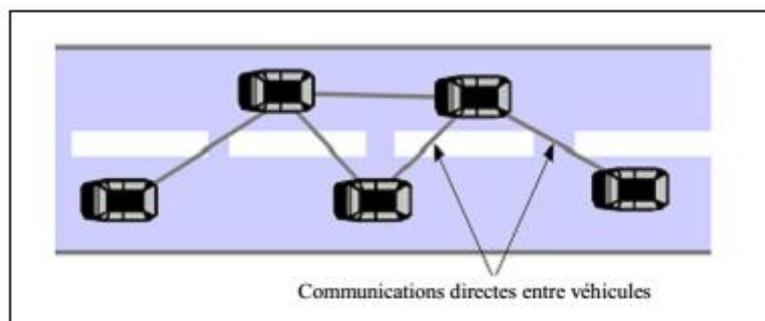


FIGURE 1.10 – Communication véhicule à véhicule

1.2.3.2 Communication de véhicule avec utilisation d'infrastructures

Dans cette catégorie, on ne se concentre pas seulement sur des simples systèmes de communications inter véhicules mais aussi sur ceux qui utilisent des stations de bases ou points d'infrastructure RSU (Road Side Units, dénomination proposée par le consortium C2C-CC). Cette approche repose sur le modèle client/serveur où les véhicules sont les clients et les stations installées le long de la route sont les serveurs. Ces serveurs sont connectés entre eux via une interface filaire ou sans fil. Toute communication doit passer par eux. Ils peuvent aussi offrir aux utilisateurs plusieurs services concernant le trafic, accès à internet, échange de données de voiture-à-domicile et même la communication de voiture-à-garage pour le diagnostic distant. L'inconvénient majeur de cette approche est que l'installation des stations le long des routes est une tâche coûteuse et prend beaucoup de temps, sans oublier les coûts relatifs à la maintenance des stations.

La figure 1.11 montre un exemple de communication de véhicule à station de base.

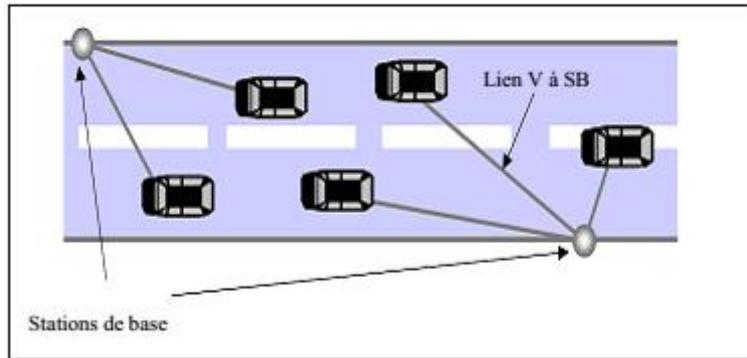


FIGURE 1.11 – Communication véhicule à station de base

1.2.3.3 Communication Hybride

La combinaison des communications véhicules à véhicules avec les communications de véhicules avec utilisation d'infrastructures, permet d'obtenir une communication hybride très intéressante. En effet, les portées des infrastructures (stations de bases) étant limitées, l'utilisation des véhicules comme relais permet d'étendre cette distance. Dans un but économique et afin d'éviter la multiplication des stations de bases à chaque coin de rue, l'utilisation des sauts par véhicules intermédiaires prend tout son importance.

La figure 1.12 montre un exemple de communication hybride.

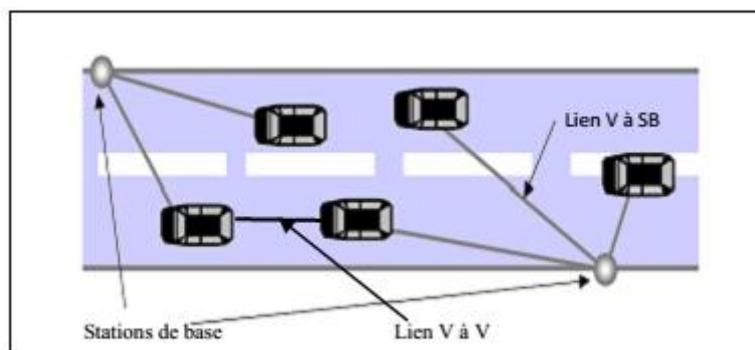


FIGURE 1.12 – communication hybride

1.2.4 Caractéristiques des réseaux VANETS

Les réseaux véhiculaires ont des caractéristiques spécifiques qui les distinguent des réseaux Ad Hoc, à savoir :

1.2.4.1 La Collecte d'informations et la perception de l'environnement proche

La collecte d'informations se fait en utilisant différents capteurs de toutes catégories (caméras, capteurs de pollution, capteurs de pluies, capteurs de l'état de la route et de voiture, etc. . .) qui permettent au conducteur à bord de son véhicule de disposer d'un certain nombre d'informations et d'une meilleure visibilité pour pouvoir réagir d'une manière adéquate aux changements de son environnement proche.

1.2.4.2 Capacité de traitement, d'énergie et de communication

Contrairement au contexte des réseaux Ad Hoc où la contrainte d'énergie à titre d'exemple représente une des problématiques traitées, les éléments du réseau VANET n'ont pas de limite en terme d'énergie et disposent d'une grande capacité de traitement et peuvent avoir plusieurs interfaces de communication (WIFI, Bluetooth et autres). Grâce aux Nouvelles Technologies de l'information et de la Communication (NTIC), le conducteur peut prendre une décision à l'aide des traitements et des interprétations des informations collectées[8][3].

1.2.4.3 Environnement de déplacement et modèle de mobilité

Les environnements pris en compte par les réseaux Ad Hoc sont souvent limités à des espaces ouverts ou indoor (comme le cas d'une conférence ou à l'intérieur d'un bâtiment). Les déplacements des véhicules quant à eux sont liés aux structures des routes (intersections, panneaux de signalisation, etc. . .) et aux stations de base routières (infrastructures) que ce soit dans les autoroutes ou au sein d'une zone métropolitaine. Les contraintes imposées par ce type d'environnement, à savoir les obstacles radio et les effets de la propagation à trajets multiples (multipath) ou d'évanouissement (fading), affectent considérablement le modèle de mobilité et la qualité des transmissions radio à prendre en compte dans les protocoles de routage. En outre la mobilité est un facteur lié directement au conducteur du véhicule.

1.2.4.4 Forte mobilité, topologie du réseau et connectivité

A la différence des réseaux Ad Hoc, les réseaux VANETs sont caractérisés par la forte mobilité des nœuds (véhicules), liée à la vitesse des voitures qui est très importante dans les autoroutes. Par conséquent, un nœud peut rejoindre ou quitter le réseau en un temps très

court, ce qui rend les changements de topologie très fréquent. De plus, des problèmes peuvent apparaître quand le système IVC (Inter Vehicle Communication) n'est pas équipé dans la majorité des véhicules[10].

1.2.4.5 Type d'information transportée et diffusée

Un des objectifs des réseaux VANETs étant la sécurité routière. Les types de communications s'axeront sur les diffusions de messages d'une source vers plusieurs destinataires. Néanmoins, les véhicules sont concernés par la diffusion d'informations en fonction de leurs positions géographiques et leurs degrés d'implication dans l'évènement déclenché. Dans de telles situations, les communications sont principalement unidirectionnelles.

1.2.5 Application des réseaux VANETs

Les principales applications des réseaux VANETs peuvent être classées en trois catégories :

1.2.5.1 Application dans la sécurité routière

Afin d'anticiper à un danger quelconque, les véhicules peuvent s'échanger entre eux des messages d'alertes ; les VANETs permettent de prévenir les collisions, les travaux sur les routes et de détecter les obstacles, ainsi que les accidents deviennent plus rapidement détectables et l'intervention devient plus rapide, cela peut minimiser le risque de décès après un accident.

1.2.5.2 Application pour l'optimisation du trafic et aide dans la conduite

Les applications de gestion de trafic sont axées sur l'amélioration des conditions de circulation dans le but de réduire les embouteillages et les risques d'accidents. Elles fournies aux conducteurs un support technique leur permettant d'adapter leur parcours à la situation du trafic routier. Ces applications visent à équilibrer la circulation des véhicules sur les routes pour une utilisation efficace de la capacité des routes et des carrefours et à réduire par conséquent les pertes humaines, la durée des voyages et la consommation d'énergie...etc.

1.2.5.3 Application au confort du conducteur et des passagers

En plus des services liés à la sécurité des véhicules et leurs occupants, les réseaux véhiculaires peuvent aussi améliorer le confort des conducteurs et des passagers. Dans le but de rendre les voyages plus agréables, les passagers peuvent communiquer soit avec d'autres véhicules par exemple jouer en réseaux, ou avec des stations fixes « infrastructure » comme l'accès à internet, la messagerie, le chat inter – véhicule, télécharger des fichiers MP3, envoyer des cartes à des amis...etc.

1.2.6 Standardisation et normalisation dans un VANET

1.2.6.1 DSRC (Dedicated Short Range Communications)

Les premiers standards définis pour les communications sans fil dans les STI utilisent la bande de fréquence de $915MHz$ essentiellement pour assurer des services tels que, le péage électronique, l'accréditation et la surveillance des opérations des véhicules commerciaux[10]. Pour offrir une norme qui forme la base pour la mise en œuvre d'un large éventail des applications dans les réseaux véhiculaires, l'ITSA a sollicité la FCC (Federal Communications Commission) d'allouer une bande passante de $75MHz$ dans la gamme de fréquences $5,850-5,925GHz$ pour les communications à courte portée dédiées aux STI aux USA qui a donné naissance à la technologie DSRC.

1.2.6.2 WAVE (Wireless Access in Vehicular Environments)

L'IEEE a développé une architecture connue sous le nom de WAVE (Wireless Access in Vehicular Environments), pour fournir l'accès sans fil dans les environnements véhiculaires[10][11]. L'architecture VAGUE « WAVE » définit deux modes de communication véhicule-à-véhicule et véhicule-à-infrastructure.

1.2.6.3 La norme IEEE 802.11p

La norme IEEE 802.11p[10][12] est un amendement du standard IEEE802.11 que le groupe de travail IEEE (TGP : Task Group p) a commencé à développer en 2004 pour l'accès sans fil dans les systèmes de transport intelligents. Il définit les spécifications des couches MAC et PHY dans le cadre des réseaux véhiculaires.

1.2.7 Les défis

1.2.7.1 Sécurité

La sécurité dans les réseaux véhiculaires Ad Hoc est cruciale, car elle affecte la vie des gens. Il est essentiel, par exemple, que l'information vitale ne puisse pas être modifiée ou supprimée par un attaquant[13]. Les communications passant par un véhicule du réseau ainsi que des informations sur les véhicules et leurs conducteurs doivent être garanties et protégées de façon à assurer le bon fonctionnement des systèmes de transport intelligents[14].

1.2.7.2 Qualité de service

La Qualité de service se mesure en fonction de l'application supportée. On peut distinguer plusieurs contraintes dans les applications utilisées dans les VANETs, parmi : la latence, les messages doivent parvenir à destination dans des délais courts ; une connectivité non intermittente, par exemple les applications de confort tel le transfert de fichiers ou le téléchargement le besoin nécessitent une connectivité permanente, . . . etc.

1.2.7.3 Canal radio fiable

Le rôle des mécanismes de gestion du canal radio est d'offrir des transmissions fiables et robustes et un partage équitable du médium de communication. Pour atteindre cet objectif dans le cas des réseaux véhiculaires, il est nécessaire de définir des méthodes qui permettent de faire face aux deux problèmes majeurs des transmissions qui sont, les interférences inter-symboles dues à la propagation des ondes par trajets multiples et l'effet Doppler causé par le mouvement des véhicules.

1.2.7.4 Routage

Pour que les véhicules puissent communiquer entre eux, un protocole de routage doit être défini. En effet, quand les terminaux ne sont pas à une portée de transmission radio directe, le routage est exigé pour établir la communication entre les véhicules. Les problèmes auxquels doivent faire face ces protocoles sont la connectivité intermittente qui rend les routes déjà établies obsolètes et le partitionnement du réseau qui empêche la propagation des paquets.

1.2.7.5 L'accès au canal

Les réseaux véhiculaires utilisent des communications radio. Par conséquent, il est important de concevoir des solutions spécifiques aux réseaux VANETs qui permettent d'apporter de la qualité de service et de gérer les priorités en résolvant les problèmes d'interférences radio, des problèmes de propagation à multiple-trajets des ondes ainsi que les irrégularités électromagnétiques.

1.2.7.6 Localisation des véhicules

Si l'un des véhicules du réseau doit être localisé (dans le cas d'un accident par exemple), les autres doivent être informés de sa position. Le problème est que tous les véhicules ne sont pas équipés d'un système de repérage par satellite (GPS). Pour cette raison, un mécanisme de localisation sans utilisation de GPS est nécessaire.

1.2.7.7 Problèmes de congestion

L'un des problèmes des VANETs est que chaque véhicule communique avec tous ceux qui sont dans sa zone de couverture. Ceci entraîne une dégradation de la qualité de service (QoS) avec l'augmentation du nombre de véhicules. Ce problème a fait l'objet de plusieurs études.

1.2.7.8 Mobilité dans la simulation des réseaux

Dans la simulation des VANETs, le facteur mobilité a longtemps été négligé. On ne considérait pas la différence de mouvements entre les nœuds des VANETs et des MANET, ce qui pouvait biaiser les résultats de la simulation. Pour cette raison, de plus en plus d'équipes de recherche s'intéressent à l'étude de la mobilité dans les VANETs. Avec un bon simulateur, plus le modèle de mobilité est réaliste, plus les résultats de la simulation sont proches de la réalité. D'où l'impact direct des modèles de mobilité sur la réussite d'une simulation.

Conclusion

Dans ce chapitre, nous avons présenté les réseaux mobiles Ad Hoc ainsi que les réseaux véhiculaires VANETs qui ne sont qu'une particularité des réseaux MANET. Nous avons signalé également leurs caractéristiques, leurs applications et leurs contraintes.

Dans le chapitre suivant nous présenterons le routage dans les réseaux mobile Ad Hoc et les réseaux VANETs, la classification des protocoles de routage selon différents critères, puis nous finissons par citer les protocoles de routage a utilisés dans notre simulation.

2

LE ROUTAGE DANS LES RÉSEAUX VÉHICULAIRES

Introduction

Le routage consiste à acheminer les paquets d'une source vers une destination suivant un itinéraire optimal. Le problème de routage a été apparu avec l'apparition des réseaux filaires étendus, où plusieurs protocoles sont définis. Le contexte sans fil des réseaux mobiles Ad Hoc a exigé la rénovation de ces protocoles qui doivent prendre en compte les contraintes de cette génération des réseaux.

Dans ce chapitre, nous allons présenter d'abord le routage dans les réseaux mobiles Ad Hoc de manière générale, la classification des protocoles selon des différents critères. Ensuite, nous passons au routage et la classification des protocoles de routage dans les réseaux véhiculaires.

2.1 Routage dans les réseaux mobiles Ad Hoc

Les réseaux mobiles Ad Hoc se caractérisent par l'absence de l'infrastructure fixe, ce qui conduit ces réseaux d'assurer leur propre organisation d'acheminer les données entre les entités mobiles.

Cet acheminement requière l'utilisation de protocoles de communication ou de routages spécifiques. Ces protocoles visent à sélectionner la meilleure route pour acheminer les paquets depuis la source vers la destination.

2.2 Problématique de routage dans les réseaux mobiles Ad Hoc

Contrairement aux réseaux filaires classiques où les nœuds sont fixes ce qui donne une stabilité des routes, la mobilité fréquente des nœuds dans les réseaux mobiles Ad Hoc et la nature des liens sans fil impliquent la modification, la disparition des routes, et l'apparition de nouvelles routes.

Ainsi que l'absence d'une infrastructure chargée de routage augmente sa complexité. Ce qui change complètement l'approche de routage par rapport aux approches classiques, tel que la

participation des nœuds dans le routage et les algorithmes utilisés. D'où la nécessité de trouver des protocoles de routage plus performants pour s'adapter à cet environnement.

Ces protocoles exigent des ressources importantes : mémoires, processeurs, bande passante et énergie, mais ces ressources sont rares dans les réseaux mobiles Ad Hoc vu ses caractéristiques, ce qui complique la conception de ces protocoles, ainsi que la participation des nœuds dans le calcul des routes nécessite l'utilisation des algorithmes de routage distribués.

2.3 Les objectifs des protocoles de routage dans les réseaux mobiles Ad Hoc

L'objectif principal d'un protocole de routage est de trouver et maintenir le chemin optimal multiple-sauts pour une communication quelconque, le sens du mot optimal dépend de la nature du réseau, la nature de l'application, la nature des informations échangées et la nature des utilisateurs, on distingue plusieurs points de vue :

- Chemin optimal qui consomme le minimum de ressources (bande passante, mémoire, processeur, énergie).
- Chemin optimal qui utilise le minimum de sauts.
- Chemin optimal qui est le plus court chemin (distance).
- Chemin optimal qui est le plus sûr.
- Chemin optimal qui assure la meilleure qualité de service ...etc.

Cet objectif peut être atteint par les points suivants :

- Éviter les boucles de routage.
- Réduire le nombre et la taille des messages de contrôle.
- Maintenir la topologie d'une manière dynamique.
- Réduire et simplifier les traitements.

2.4 Classification des protocoles de routage dans les réseaux mobiles Ad Hoc

Les protocoles de routage destinés aux réseaux mobiles Ad Hoc peuvent être classés de différentes manières, selon plusieurs critères. Nous allons citer, selon leur architecture (uniforme ou non uniforme), leur approche de routage (protocole pro-actif, réactif ou hybride).

2.4.1 Classification selon l'architecture

Ce critère divise les protocoles de routage en deux classes :

2.4.1.1 Les protocoles uniformes

Dans les protocoles de routage uniformes, les nœuds du réseau jouent le même rôle pour la fonction de routage, tous les nœuds sont égaux, aucune hiérarchie n'est définie entre les nœuds du réseau, chaque nœud envoie et reçoit des messages de contrôle de routage.

2.4.1.2 Les protocoles non uniformes

Contrairement aux protocoles uniformes, les protocoles de routage non uniformes «hiérarchiques» visent de limiter la complexité du routage en minimisant le nombre de nœuds qui contribuent à la détermination des routes ; cette structure hiérarchique attribue aux nœuds des rôles différents selon leurs fonctions.

2.4.2 Classification selon l'approche de routage

Dans cette classification basée sur le mécanisme d'établissement de la route, nous distinguons trois classes :

2.4.2.1 les protocoles proactifs

Les protocoles proactifs dits à diffusion de table se caractérisent par l'établissement de route à l'avance, de façon que lorsqu'un nœud désire envoyer un paquet à un autre nœud, une route soit immédiatement connue, chaque nœud maintient une ou plusieurs tables qui contiennent des informations de routage.

Cette catégorie de protocole requiert un échange périodique de messages de contrôles afin de mettre à jour les tables de routages dans chaque nœud. Ainsi, de nouvelles routes seront construites à partir des informations transportées par les trames de contrôle. Ce processus est déclenché à chaque changement de topologie pour reconstruire à nouveau les routes vers toutes les destinations possibles.

Les protocoles basés sur ce principe sont : DSDV[15] et OLSR[16].

Un des avantages de ces protocoles est la disponibilité immédiate de la route lors du besoin « gain de temps ». Cependant, la bande passante diminue à cause du trafic généré par l'échange de paquets de contrôles.

2.4.2.2 Les protocoles réactifs

Le principe des protocoles réactifs également appelés protocoles de routage à la demande (On-Demand routing protocols) est de lancer le processus de recherche de routes uniquement en cas de besoin (à la demande).

Ces protocoles se basent sur la découverte et le maintien des routes. Suite à un besoin, une procédure de découverte globale de route est lancée[17], en cherchant un chemin jusqu'à la destination, une fois ce chemin trouvé, il est inscrit dans la table de routage et peut être utilisé tant que la destination est joignable ou jusqu'au moment où la route devient inutile.

Ce type de routage minimise l'échange de messages de contrôle ce qui libère la bande passante « bande passante plus large » mais le délai d'établissement de la route est plus important en comparaison avec les protocoles proactifs.

Actuellement le plus connu de ces protocoles est : DSR[18], AODV[19], TORA[20].

2.4.2.3 Les protocoles hybrides

Les protocoles hybrides combinent les deux approches précédentes. Pour bénéficier de leurs avantages, ils utilisent un protocole proactif, pour connaître les voisins les plus proches dans le but de réduire le délai et un protocole réactif dans le but de réduire la charge des paquets

de contrôles. Les protocoles hybrides cumulent aussi les inconvénients des protocoles Table-driven et réactifs à savoir, les paquets de contrôles périodiques et le délai de découvertes de routes[21]. Parmi les protocoles hybrides les plus connus on peut citer le protocole : ZRP[22]. Une comparaison entre les deux classes proactive et réactive est présentée dans le tableau 2.1[23].

Routage proactif		Routage réactif	
Avantages	inconvénients	Avantages	inconvénients
<p>La topologie du réseau est connue de tous les mobiles.</p> <p>Les routes sont disponibles immédiatement.</p>	<p>Il faut diffuser régulièrement des informations sur les changements de topologie du réseau.</p>	<p>Les mobiles ne conservent pratiquement aucune information sur la topologie globale du réseau : seules les informations sur les routes actives sont stockées.</p>	
<p>Les protocoles proactifs disposent en permanence d'une route pour chaque destination dans le réseau.</p>	<p>Un volume de signalisations important.</p>	<p>Les protocoles réactifs génèrent à priori un volume plus faible de signalisations.</p>	<p>Les protocoles réactifs engendrent un délai lors de la construction (ou de la reconstruction) des routes et produisent plus difficilement des routes optimales.</p>

TABLE 2.1 – Comparaison entre protocoles proactifs et protocoles réactifs

2.4.3 Classification selon le type d’algorithme utilisé

Une autre classification basée sur le type d’algorithme utilisé est possible pour les protocoles de routage Ad Hoc. Il existe deux grandes familles d’algorithmes de routage dynamique (vecteur de distance ou état de liens) et deux grandes familles d’algorithmes de routage à la demande (source ou apprentissage en arrière).

2.4.3.1 Les protocoles de routage à vecteur de distance (Distance Vector Protocols)

Ces protocoles sont basés sur l’algorithme de Bellman-Ford. Leur principe est basé sur l’échange entre les nœuds voisins d’informations de distance des destinations connues. Autrement dit, chaque nœud envoie à ses voisins la liste des destinations joignables et le coût (distance) associé au chemin le plus court menant vers cette destination.

A chaque réception d’un paquet contenant les informations topologiques, le nœud en question met à jour sa liste de destination par le coût minimum.

Les protocoles à vecteur de distance sont simples à programmer et facile à implémenter. En revanche, ils ont un mauvais comportement dans les réseaux dynamiques tel que les boucles de routage et le comptage à l’infini connu aussi sous le nom de problème de Bellman-Ford[24][25].

2.4.3.2 Les protocoles de routage à état des liens (Link State Protocols)

Ces protocoles utilisent un algorithme plus efficace en calcul du plus court chemin que celui des protocoles de routage à vecteur de distance, qui est l’algorithme de Dijkstra. Ils sont basés sur le l’état des liens (topologie) du réseau, l’ensemble de ces informations permet aux nœuds de dessiner une vue globale sur le réseau. Une table de routage est maintenue dans chaque nœud. Elle est construite à partir des informations échangées sur l’état des liens du réseau[26].

A partir de cette vue globale du réseau, il est facile de trouver des routes alternatives lorsqu’un lien est rompu. Ainsi une route est immédiatement disponible à la demande. Il est même

possible d'utiliser simultanément plusieurs routes vers une même destination, augmentant ainsi la répartition de la charge et la tolérance aux pannes dans le réseau.

En contre partie, si le réseau est étendu, la quantité d'informations sur l'état de tous les liens du réseau au niveau de chaque nœud nécessite un espace de stockage considérable.

2.4.3.3 Le protocole de routage source

Chaque nœud doit posséder la topologie et les caractéristiques du réseau en entier. Les données doivent être parfaitement à jour. Ce routage convient aux réseaux de tailles moyennes (pour éviter la surcharge de la mémoire) à hauts débits (le calcul de routes est effectué une seule fois). Pour cela, le temps de calcul des routes ne doit pas être trop grand.

Dans cet algorithme, afin d'émettre un paquet de données à un nœud, l'émetteur spécifie dans l'entête du paquet à envoyer l'adresse de chaque nœud à travers lequel le paquet va passer pour atteindre la destination (route source). Par la suite, l'émetteur transmet le paquet via son interface, au premier nœud spécifié dans la route source. Un nœud qui reçoit le paquet, et qui est différent de la destination, supprime son adresse de l'entête du paquet reçu et le transmet au nœud suivant identifié dans la route source.

Ce processus se répète jusqu'à ce que le paquet atteigne sa destination finale. Enfin, le paquet est délivré à la couche réseau du dernier hôte.

2.4.3.4 Le protocole de routage par apprentissage en arrière

Le chemin établi entre les nœuds est un chemin bidirectionnel simultané (full duplex). La source gardera trace du chemin tant qu'il restera en cours d'utilisation. Ce type de routage nécessite moins de mémoire que le routage source. Par conséquent, il est plus adapté pour des réseaux de plus grandes tailles.

Afin de transmettre un paquet à l'aide de cette méthode, le nœud émetteur inonde le réseau avec sa requête. Ainsi chaque nœud intermédiaire, dit le transit, indique le chemin au nœud source lors de la réception de la requête. On dit qu'il apprend le chemin au nœud source, tout en sauvegardant la route dans la table transmise.

Enfin, lorsque la requête arrive au nœud destinataire, et suivant le même chemin, ce dernier transmet sa réponse sous forme de requête.

2.5 Protocoles de routage dans les réseaux VANET

Le problème de routage dans les réseaux de véhicules réside essentiellement dans l'instabilité des chemins causée par la forte mobilité des nœuds et les fragmentations fréquentes du réseau[27]. Aussi bien, la vitesse de déplacement des nœuds est beaucoup plus élevée que dans les MANET.

Dans notre étude, on s'intéresse exclusivement au routage unicast car le besoin en ressources est plus critique pour faire aboutir les communications. On distingue deux grandes classes de protocoles de routage unicast selon le type d'informations utilisées pour acheminer les données. La première classe est celle des protocoles qui se basent sur des informations sur la topologie du réseau, ce sont les protocoles du groupe MANET. La seconde classe est celle des nouveaux protocoles dits géographiques ou de position qui se basent sur des informations supplémentaires sur la position géographique[10].

2.5.1 Classification des protocoles de routage dans les réseaux VANET

Nous définissons deux classes de protocoles de routage : les protocoles basés sur la topologie qui sont divisés en protocoles proactifs, réactifs et hybrides et les protocoles basés sur la localisation (géographique).

2.5.1.1 Les protocoles de routage basés sur la topologie

Les protocoles de routage basés sur la topologie utilisent les informations sur les liens qui existent entre les nœuds pour l'acheminement des paquets. Cette famille de protocoles peut être divisée en trois catégories : proactifs, réactifs et hybrides, qui sont déjà défini dans la partie 2.4.2 de ce chapitre.

Chaque nœud utilise comme données l'état de ses connexions avec ses nœuds voisins ; cette information est ensuite transmise aux autres nœuds pour leur offrir une connaissance plus précise sur la topologie du réseau.

2.5.1.2 Les protocoles de routage basés sur la géographique

Les protocoles de routage géographique (ou basés sur la position) utilisent des coordonnées géographiques (par exemple, fournies par un système de géolocalisation tel que le GPS) afin de trouver un chemin vers la destination[28]. Chaque nœud source inclut l'identifiant et la position de la destination dans l'entête de tout paquet à envoyer, les nœuds recevant ce paquet utilisent les informations géographiques incluses dans ce dernier et celles disponibles dans leurs tables de routage pour retransmettre le paquet et répètent le même mécanisme jusqu'à ce que celui-ci atteigne la destination.

L'avantage majeur de ces protocoles par rapport aux protocoles basés topologie, est qu'ils réduisent considérablement les paquets de contrôles, particulièrement dans les réseaux larges et dynamiques. Parmi les protocoles géographiques les plus largement étudiés : GPSR[29], LAR[30] et DREAM[31].

La figure 2.1 montre les différents protocoles de routage dans les réseaux VANET.

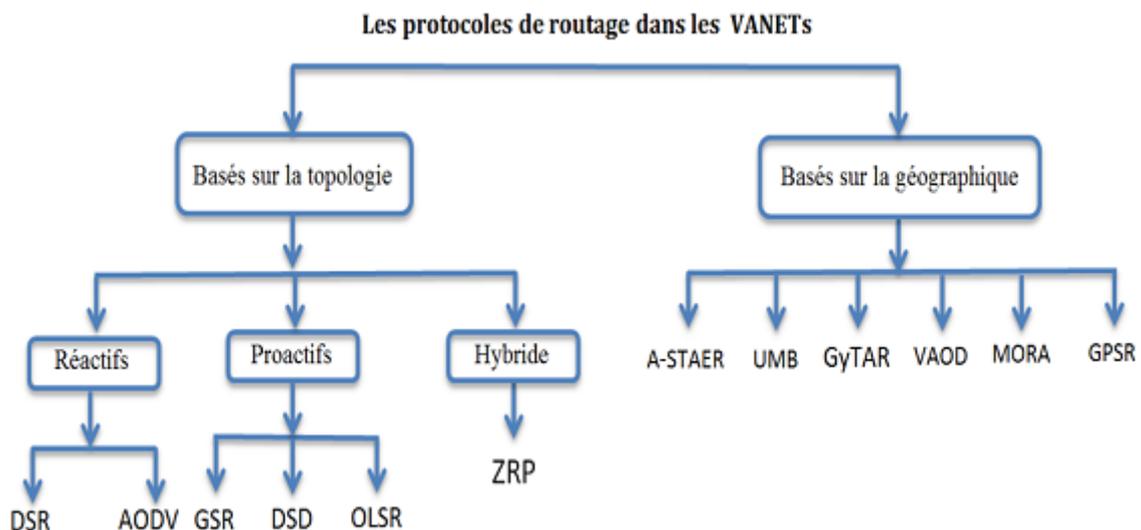


FIGURE 2.1 – Protocoles de routage dans les réseaux VANET[34]

2.5.2 Protocoles de routage utilisés dans notre étude

2.5.2.1 Le protocole DSR

DSR pour Dynamic Source Routing est un protocole de routage réactif uniforme, simple et efficace, développé dans le cadre du projet monarch à l'université Carnegie Mellon, il est basé sur la technique de routage par la source sans utilisation de la table de routage contrairement au protocole OLSR et AODV[32]. Dans cet algorithme, chaque paquet contient la séquence complète des nœuds à travers lesquels il doit passer pour arriver à la destination. L'avantage de l'utilisation de l'algorithme de routage source est que les nœuds de transit n'ont pas besoin de maintenir à jour les informations sur la route puisque le paquet possède toutes ces informations. Un autre avantage de routage par la source est la possibilité du contrôle du trafic en lui proposant de suivre le chemin le plus adapté à ses besoins de qualités de services, Microsoft a proposé une version DSR (LQSR : Link Quality Source Routing) qui inclut la qualité de services. L'inconvénient de cette méthode est l'augmentation de la taille des paquets qui grandit avec l'augmentation des nœuds à traverser. DSR est décomposé en deux processus, le premier est utilisé pour la découverte de la route à la demande et le second s'occupe des routes de communication en cours.

2.5.2.2 Le protocole AODV

AODV pour Ad-hoc On-Demand Distance Vector est un protocole réactif destiné aux réseaux mobiles Ad Hoc. Dans ce protocole, les nœuds se réfèrent à leurs tables de routage comme une base pour acheminer les paquets. Ces tables de routages contiennent des informations qui sont mis à jours à partir des données contenues dans les paquets de contrôle. Lorsqu'un nœud source désire envoyer des données vers un destinataire, il vérifie tout d'abord dans sa table de routage s'il existe une route valide vers ce destinataire. Si la route n'est pas trouvée, le nœud source lance la procédure de découverte de route en diffusant en broadcast un paquet RREQ à la recherche d'un chemin vers le destinataire. A la réception de ce paquet, le nœud répond par un paquet RREP s'il est lui-même le destinataire ou s'il possède dans sa table une route vers la destination. Dans le cas contraire, c'est-à-dire si la table de routage ne contient pas de route vers le destinataire, le nœud rediffuse le RREQ. Une fois la route trouvée, le nœud source transmet les paquets de données en transitant de proche en proche et

chaque nœud détermine le prochain relai à partir de sa table de routage. En cas de rupture de route, le nœud intermédiaire envoie un paquet RERR pour informer la source qui décide ou non de recommencer l'envoi du paquet[33].

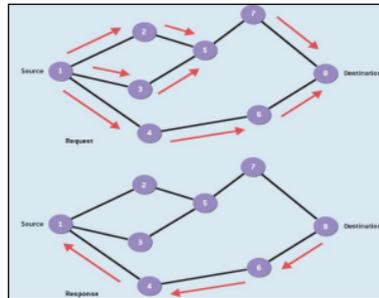


FIGURE 2.2 – Mécanisme de routage AODV

Conclusion

Dans ce chapitre nous avons présenté le concept de routage dans les réseaux Ad Hoc et la classification des protocoles selon différentes critères. Ensuite, on a défini le routage dans VANET, par conséquent, nous avons constaté que les différentes caractéristiques supportées dans VANET comme la forte mobilité qui entraîne une topologie très dynamique, et la grande vitesse rendent difficile, dans certains cas, la détermination de routes stables entre les sources et les destinations des données. En effet, cela nous a permis de conclure que le choix de l'algorithme de routage dépend de plusieurs contraintes. Enfin, nous avons expliqués les deux protocoles de routages qui sont de base défini dans les réseaux MANET ainsi déployés dans VANET, à savoir DSR et AODV.

Dans le prochain chapitre, Nous allons évaluer a l'aide d'une simulation les performances des protocoles de routage DSR et AODV dans les réseaux véhiculaires en milieu urbain et autoroutier pour en-tirer le meilleur.

3

SIMULATION ET DISCUSSION DES RESULTATS

Introduction

Le but de toute simulation est de tester les systèmes réalisés avant qu'ils soient mis en marche afin de déterminer les conditions exactes de leur fonctionnement, évaluer leurs performances et prévoir les différents états liés à ces systèmes et donc de bien réagir et prendre des décisions.

Dans notre cas d'étude, les tests à effectuer sont nombreux (selon le nombre de métriques à faire varier). Nous avons choisi les plus importantes afin de montrer les performances des protocoles de routage DSR et AODV dans un réseau VANET. Ces protocoles sont implémentés sous OMNeT++ et ils ont déjà fait leurs preuves dans des études précédentes, c'est pour cela que notre choix s'est porté dessus.

Dans ce dernier chapitre on commence par l'explication du principe de la simulation dans les VANETs. Nous décrivons ainsi, l'environnement de travail, les outils, les étapes suivies, quelques métriques et paramètres principaux utilisés dans notre simulation. Nous finissons par une discussion sur les résultats de performance des protocoles DSR et AODV dans les réseaux véhiculaires en milieu urbain et autoroutier.

3.1 Simulation dans les VANETs

La simulation dans les VANETs implique deux différents aspects. Le premier réside aux problèmes liés à la communication entre les véhicules. Un simulateur de réseau, comme OMNeT++, fait face à ces problèmes, il se focalise sur les caractéristiques du protocole de réseau, le deuxième aspect très important est lié à la mobilité des nœuds « véhicules », c'est le simulateur SUMO qui gère la mobilité et le mouvement des véhicules. Les simulateurs définis pour les MANET peuvent être utilisés pour les VANETs, néanmoins, ces simulateurs doivent être étendus pour inclure les solutions spécifiques aux réseaux véhiculaires tels que la norme 802.11p (un système de communication pour véhicule).

3.1.1 Simulation de la mobilité dans les VANETs

Les véhicules dans les réseaux véhiculaires ne peuvent se déplacer que sur les routes ou les chemins définis dans la topologie, et doivent obéir aux règles de circulation, et d'autre part,

par la prise en compte de l'interaction entre véhicules. Alors, ils ont moins de liberté dans leurs mouvements et ils sont limités par les signalisations routières telles que des panneaux d'arrêt, feux de circulation et interdépendants obstacles tels que les bâtiments...etc.

3.2 Outils de simulation

3.2.1 Objectif de la simulation

L'objectif principal de notre simulation est l'évaluation des performances des protocoles de routage DSR et AODV des réseaux véhiculaires en milieu urbain et autoroutier en terme de débit et de taux de paquets perdu. Mais le problème que nous avons rencontré c'est qu'il n'y a pas de protocole de routage conçu pour les VANETs c'est pour ça que nous sommes dans l'obligation d'utiliser des protocoles MANET.

3.2.2 Les simulateurs Réseaux

La simulation des réseaux est une technique par laquelle un logiciel ou simulateur modélise le comportement d'un réseau. Il existe plusieurs simulateurs de réseaux, nous décrivons ci-dessous ceux qui sont fréquemment utilisés.

Parmi les simulateurs les plus utilisés dans les laboratoires de recherche, nous citons NS-3[35] qui est la troisième version de NS (Network Simulator). C'est un simulateur gratuit à événements discrets qui fournit un support étendu pour la simulation de TCP/IP, le routage et les protocoles multicast aussi bien dans les réseaux câblés que dans les réseaux sans fil. NS-3 comprend des technologies et des applications réseaux les plus courantes en plus c'est un logiciel libre d'utilisation pour la recherche et l'éducation. Il est conçu de base pour des systèmes Unix mais des outils comme CygWin permettent de le faire fonctionner sur un système Windows.

Comme solution payante nous retrouvons OPNET[36]. C'est une plate-forme commerciale pour simuler les réseaux de communication. Conceptuellement, le modèle d'OPNET comprend des processus basés sur des machines à états finis. Le modèle sans fil est basé sur une architecture pipeline pour déterminer la connectivité et la propagation entre les nœuds. Les utilisateurs

peuvent spécifier la fréquence, la bande passante, et la puissance entre autres caractéristiques, y compris les diagrammes de gain d'antenne et des modèles de terrain.

3.2.2.1 OMNeT++

OMNeT++[37] pour Objective Modular Network Test-bed in C++ est un, espace de simulation modulaire à base de composants Open Source. Son domaine d'application principal est celui des réseaux de communication. OMNeT++ présente une architecture générique et flexible ce qui lui permet aussi d'être efficace dans d'autres domaines tels que les systèmes informatiques, les réseaux de files d'attente, des architectures matérielles, ou même des processus d'affaires.

OPNET est un simulateur payant, le code source n'est pas publié aux utilisateurs et tous les modules complémentaires ne sont pas gratuits aussi. Il est possible que les fabricants puissent fournir une licence gratuite pour un usage académique, mais ça ne sera qu'avec certaines limites et conditions fixés par les fabricants ; il est possible aussi de devoir renouveler la licence libre après une certaine période. NS-3, et OMNeT++ sont des simulateurs open source et gratuits, notre choix c'est porté sur OMNeT++ pour les raisons suivantes :

- **Gestion des modèles**

Le noyau de simulation OMNeT ++ est une bibliothèque de classes, les modèles dans OMNeT++ sont indépendants du noyau. Les chercheurs peuvent écrire leurs propres composants (modules simples) sans devoir toucher au noyau du simulateur grâce aux API qu'il fournisse. Le code source de OMNeT ++ n'est jamais affecté par les nouveaux modèles développés. En plus, les Modules simples sont réutilisables, et peuvent être combinés librement pour créer des simulations. En revanche, dans NS-3, la frontière entre le noyau de simulation et les modèles est étroite, sans des API qui séparent le simulateur et les modèles.

- **Système d'exploitation**

Tandis que OMNeT++ fonctionne sur Windows, Linux, Mac OS et d'autres systèmes de type Unix. NS-3 n'est disponible que sur Linux et Mac OS X et il faut passer par Cygwin pour le faire fonctionner sur Windows.

- **Modèle de simulation**

Notre projet se base principalement sur les deux normes IEEE 802.11p et WAVE 1609.X. Le modèle 802.11p est disponible à la fois sur OMNeT++ et sur NS-3. Cependant, le modèle (VEINS Framework) disponible sur OMNET++ est plus simple et plus complet pour simuler le trafic dans les réseaux VANETs.

Architecture d'OMNeT++

L'architecture du modèle OMNeT++ se compose de plusieurs modules hiérarchiquement (visualisé dans la Figure 3.1) imbriqués qui sont :

- Le module système
- Modules simples (les feuilles) : programmés en C++ encapsulant le comportement d'un réel d'un système. Pour chaque module simple correspond un fichier.cc et un fichier.h.
- Modules composés : constitués d'un ou de plusieurs modules simples ou des modules composés reliés entre eux. Les paramètres, les ports et les modules de chaque module sont spécifiés dans un fichier.ned. l'architecture d'OMNET++ est visualisée dans la figure 3.1 qui suit :

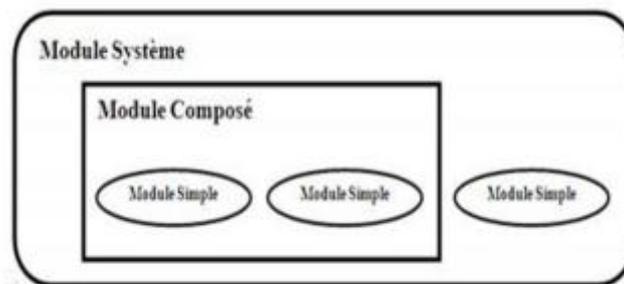


FIGURE 3.1 – Architecture modulaire du simulateur OMNeT++

Les modèles OMNeT++ sont écrits en langage C++. La communication entre deux modules communiquant ce fait par échange de messages qui sont soit des paquets, des trames d'un réseau informatique, des clients dans une file d'attente ou d'autres types d'entités. Les messages sont échangés à travers des interfaces d'entrées/sorties des modules.

Dans la construction d'un nouveau programme de simulation, il y'aura à chaque fois des informations chargées dynamiquement telles que la topologie du réseau à partir des fichiers.ned et les configurations sont disponibles dans les fichiers « .ini ».

Lors de la simulation, différents fichiers trace seront remplies. On a aussi le « Plove » qui est un outil pouvant visualiser les données enregistrées. Les deux fichiers omnet.vec et omnet.sca seront utiles lors du traçage de la courbe et du calcul des statistiques.

Composants d'OMNeT++

Les composants les plus pertinents dans le simulateur OMNeT++ ainsi que son interface sont respectivement représentés dans le Tableau 3.1 et la figure 3.2.

Application	FTP, Telnet, generateur de trafic (IP Trf Gen..), Ethernet, Ping App, UDP App, TCP App
Transport	TCP, UDP, RTP
Réseau	IPv4, IPv6, ARP, OSPF, LDP, MPLS, ICMP, TED
Liaison	Mgmt, MAC, Radio
Node	Ad Hoc, Wireless, MPLS

TABLE 3.1 – Les composants d'OMNeT++

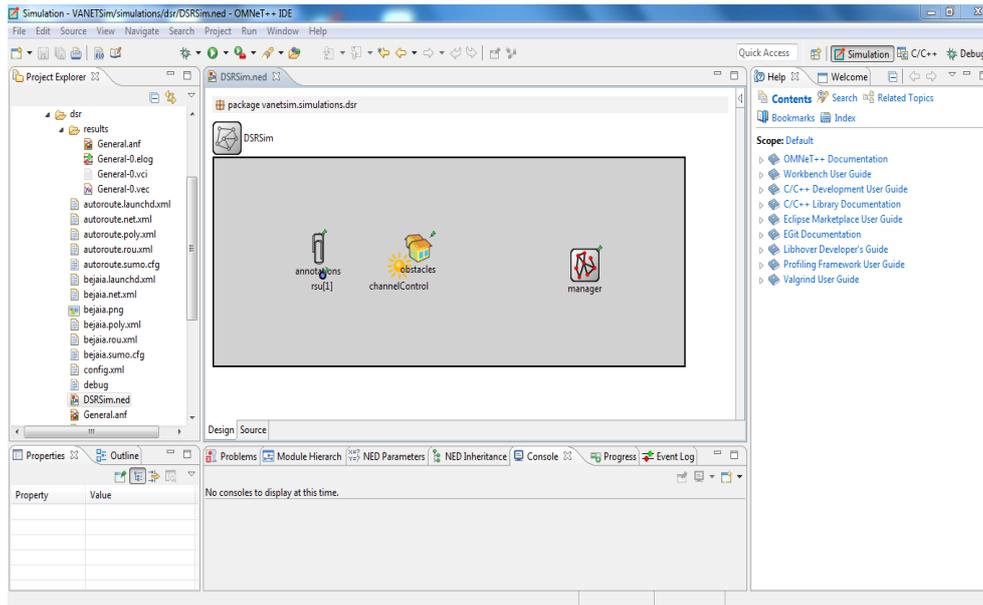


FIGURE 3.2 – Interface d’OMNeT++

3.2.2.2 Modèle de simulation

Pour notre projet, nous avons utilisé OMNeT++ 4.6, INET-3.2.4 et VEINS 4.4 comme modèle de simulation de 802.11p.

INET Framework

INET[38] est un package OMNeT++ open source qui fait office de simulateur de communication réseaux. Il permet d’avoir un module avec Hôtes, routeurs, commutateurs et autres périphériques réseau qui sont composé par des modules OMNeT++. Ces modules composés sont assemblés à partir de modules simples qui représentent des protocoles, des applications et d’autres unités fonctionnelles. Un réseau est aussi un module OMNeT++ composé qui contient : hôte, routeur et d’autres modules. Les interfaces externes de modules sont décrites dans les fichiers NED qui définit les paramètres et les passerelles (à savoir les ports ou connecteurs) de modules, ainsi que les sous-modules et les connexions de modules composés.

Les protocoles supportés par INET-3.2.4 sont listés dans le Tableau 3.2.

Application	CBR/VBR HTTP File Transfer DHCP Video Voice Peer-to-peer
Transport	TCP, UDP, SCTP
Network	IPv4 ICMv4 ARP IGMPv2 IPv6 ICMPv6 mipV6 HIP
routing	link-state routing OSPF, BGP, RIP, RSTP
MANET Routing	AODV, DYMO, DSR, OSLR, GPSR
MPLS	MPLS, LDP, RSVP-TE
Wired	PPP Ethernet
Wireless	802.11a, 802.11b, 802.11g, 802.11P, 802.11s, 802.11e, 802.16(WiMAX)

TABLE 3.2 – Tableau des protocoles supportés par INET

VEINS Framework

VEINS[39] est un Framework open source qui permet d'exécuter des simulations des réseaux ad-hoc véhiculaires VANETs. Il est basé sur deux simulateurs qui sont OMNeT++ et SUMO, un simulateur de trafic routier. L'architecture de Veins est illustrée dans la Figure 3.3.

Les deux simulateurs sont combinés pour offrir une plateforme complète de modèles pour la simulation des VANETs. La simulation du trafic routier est effectuée par SUMO, qui est désigné pour le domaine de l'ingénierie de trafic. La simulation des communications réseaux est effectuée par OMNeT ++ avec la modélisation de la couche physique grâce au modelés MIXIM[40], ce qui permet d'employer des modèles précis pour les interférences radio, ainsi que l'observation par des obstacles fixes et mobiles. Les deux simulateurs sont inter-connectés et des simulations sont effectuées en ligne. De cette façon, l'influence des réseaux de véhicules sur la circulation routière peut être modélisée et des interactions complexes entre les deux simulateurs sont possibles grâce à une connexion via un socket TCP.

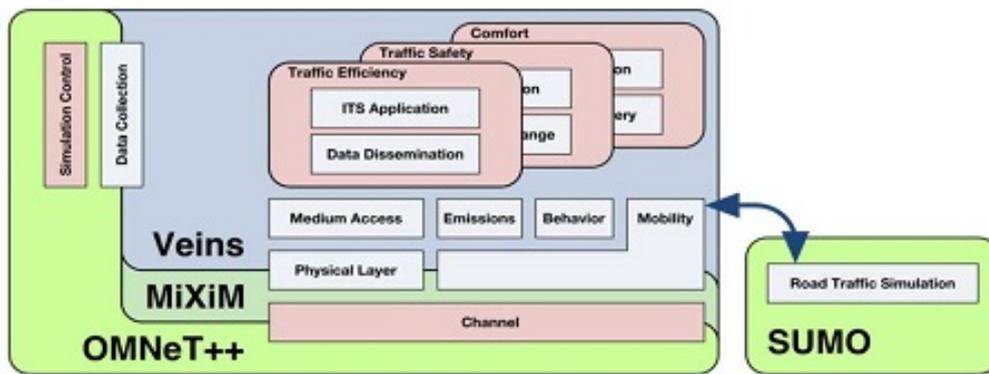


FIGURE 3.3 – Architecture de VEINS[41]

Les protocoles IEEE 802.11p et IEEE 1609,4 sont entièrement pris en charge par la version 4.4 de VEINS. Ses caractéristiques sont répertoriées comme suit :

- Basé sur cent pour cent de logiciels open source offrant une extensibilité illimitée.
- Permet la reconfiguration en re-routage en ligne des véhicules en réaction des paquets réseau.
- Repose sur des modèles entièrement détaillées de couches réseaux 802.11p IEEE et IEEE 1609,4 DSRC / WAVE, avec le support du multi-canal, QoS, les effets de bruit et d'interférence.
- Peut être déployé sur des clusters de calcul pour la simulation dans un MRIP distribués de façon parallèle.
- Peut importer des scénarios entiers du site web d'OpenStreetMap, y compris les bâtiments, les limitations de vitesse, nombre de voies, feux de circulation, l'accès et les interdictions de tourner.
- Permet d'utiliser un large éventail de mesures, comme le temps de Voyage et les délais de transmissions.
- Pris en charge solide par un nombre très important d'utilisateurs venant des cinq continents.

3.2.2.3 Le simulateur de mobilité SUMO

SUMO[42] « Simulation of Urban MObility » est un logiciel open source disponible pour Windows et Linux, qui permet d'effectuer des simulations de trafic routier. Il est très portable et nécessite uniquement l'installation des bibliothèques C++. Il permet la simulation de trafic microscopique, multi-modal, et de simuler la manière dont une demande de trafic donnée qui se compose de véhicules individuels se déplace à travers un réseau routier donné.

La simulation permet d'aborder un large éventail de sujets de gestion du trafic. Elle est purement microscopique : chaque véhicule est modélisé de manière explicite, présente une voie propre, et se déplace individuellement à travers le réseau. SUMO comprend toutes les applications nécessaires pour préparer et effectuer une simulation de trafic (réseau et itinéraires importation, DUA, simulation), une simulation Mouvement du véhicule spatial continu et à temps discret. Il offre aussi différents types de véhicules, rues à plusieurs voies avec changement de voie, signalisations différentes à droite de la voie, feux de circulation. Avec une interface utilisateur graphique rapide openGL, Il gère des réseaux avec plus de 10.000 bords (rues) avec une vitesse d'exécution rapide (jusqu'à 100.000 véhicules mises à *jour/s* sur une machine à 1GHz). L'interopérabilité avec d'autres applications lors de l'exécution nous a permis de l'inter-connecter avec OMNeT++ via une connexion TCP avec une seule ligne dans l'invité de commande(Mingwenv) :

```
/c/"veins path"/sumo-launchd.py -vv -c /c/"sumo path"/bin/sumo-gui.exe
```

3.3 Environnement de travail

3.3.0.4 Environnement matériel

La simulation a été réalisée sur un ordinateur COMPAQ dont la configuration est la suivante : COMPAQ cq56, Dual Core ,250GO Disque Dur et 4Go de Ram.

3.3.0.5 Environnement logiciel

Notre simulation a été réalisée dans l'environnement logiciel suivant :

- Système d'exploitation : Windows 7.

- Le simulateur OMNeT++ 4.6 : C'est un simulateur Open Source des réseaux de communication supportant des modèles de mobilités. Il est basé sur C++ et réalise des simulations discrètes.
- Le simulateur SUMO 0.19.0 : C'est un simulateur Open Source, portable, microscopique et continu de la circulation routière. Il est conçu pour gérer de grands réseaux routiers.
- Outils de conception.

3.4 Etapes de la simulation

Dans notre simulation nous avons deux cas d'études, le premier dans un milieu urbain en ville de bejaia plus exactement aux alentours du rondpoint de Dawadji et le deuxième dans un milieu autoroutier a l'autoroute Est-Ouest. Les étapes de simulation sont les mêmes pour les deux cas, donc on a pris en compte le premier cas. Ça carte a été obtenue à partir du premier site open source des cartes routières qui est <http://www.openstreetmap.org>, comme il est montré dans la figure 3.4.

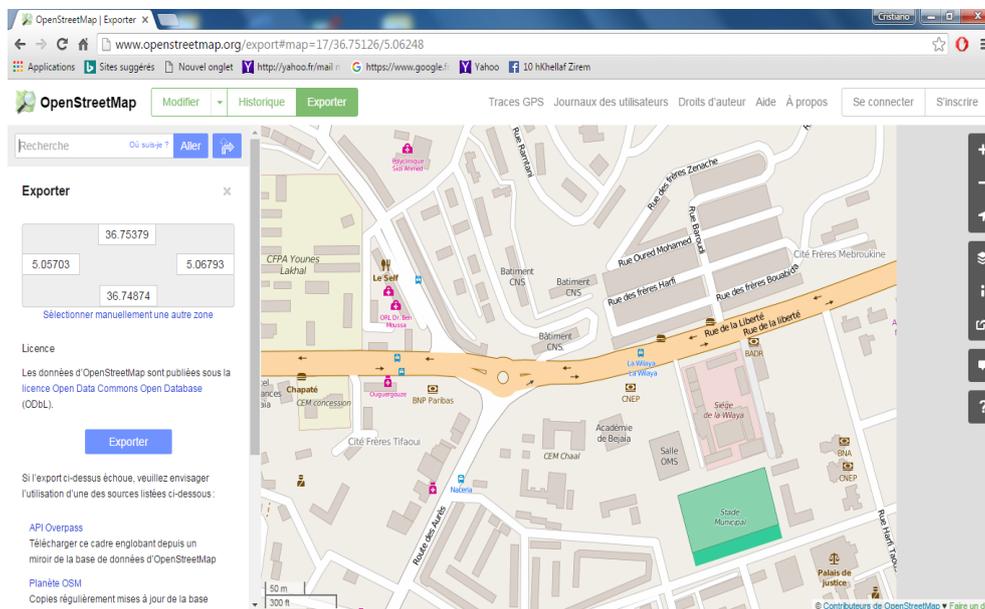


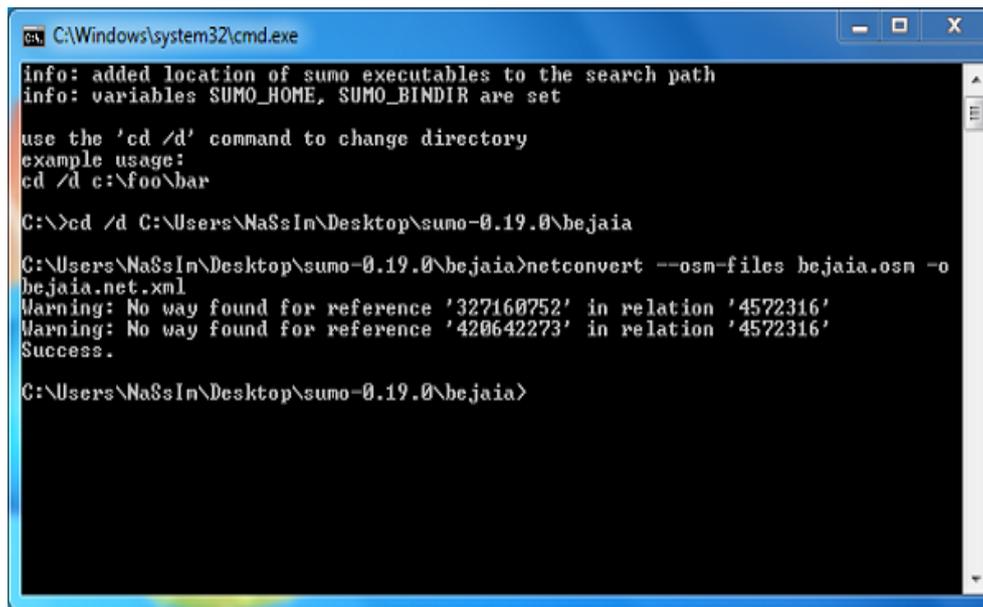
FIGURE 3.4 – Zone d'étude de la ville de Bejaia dans OpenStreetMap

Ensuite, on clique sur le bouton Exporter qui se trouve en haut , une fenêtre apparaît nous demande si nous souhaitons ouvrir ou enregistrer la carte, nous devons cliquer sur enregistrer.

Préparation de la carte pour l'utilisation dans SUMO

On copie le fichier map.osm que nous avons téléchargé précédemment dans le répertoire bejaia dans sumo-0.19.0, puis cet carte en format « *.osm » est convertie vers un autre format connu par le simulateur SUMO grâce à la commande :

netconvert --osm-files bejaia.osm -o bejaia.net.xml illustré dans la figure 3.5.



```

C:\Windows\system32\cmd.exe
info: added location of sumo executables to the search path
info: variables SUMO_HOME, SUMO_BINDIR are set

use the 'cd /d' command to change directory
example usage:
cd /d c:\foo\bar

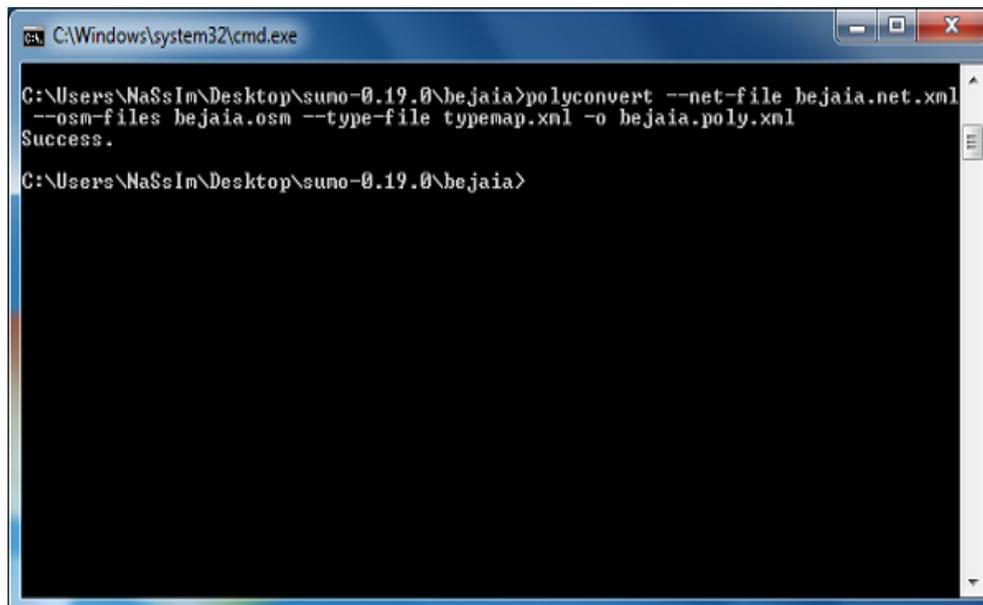
C:\>cd /d C:\Users\NaSsIn\Desktop\sumo-0.19.0\bejaia
C:\Users\NaSsIn\Desktop\sumo-0.19.0\bejaia>netconvert --osm-files bejaia.osm -o
bejaia.net.xml
Warning: No way found for reference '327160752' in relation '4572316'
Warning: No way found for reference '420642273' in relation '4572316'
Success.
C:\Users\NaSsIn\Desktop\sumo-0.19.0\bejaia>
    
```

FIGURE 3.5 – Génération de bejaia.net.xml

Maintenant, il est nécessaire de créer le fichier typemap.xml. Il existe de nombreuses façons de le faire, une option est d'utiliser l'interface graphique dans un environnement Linux. Alors, on entre dans le répertoire bejaia et on crée un nouveau fichier et on le nomme typemap.xml. Ensuite, on ouvre un navigateur et on accède au site Web suivant :

<http://sumo.dlr.de/wiki/Networks/Import/OpenStreetMap> et on copie son contenu dans le fichier typemap.xml que nous avons créé avant et l'enregistrer, comme il est représenté sur les figures 3.6, 3.7.

Cette étape est illustrée sur la figure 3.8.



```

C:\Windows\system32\cmd.exe
C:\Users\Na$Im\Desktop\suno-0.19.0\bejaia>polyconvert --net-file bejaia.net.xml
--osm-files bejaia.osm --type-file typemap.xml -o bejaia.poly.xml
Success.
C:\Users\Na$Im\Desktop\suno-0.19.0\bejaia>
    
```

FIGURE 3.8 – Génération de bejaia.poly.xml

Après avoir défini la topologie du réseau, il ne reste plus qu'à générer la demande de trafic, à savoir la description des itinéraires que les véhicules suivent. Il existe plusieurs méthodes pour générer la demande de trafic en SUMO :

- Utilisation de définitions Route.
- Utilisation de définitions de voyage.
- En utilisant les définitions de flors (semblables ci-dessus, mais unir les véhicules avec Voyage similaire dans les groupes).
- En utilisant les définitions des flux aux intersections et au taux de rotation (la cible du lien est pas spécifié, et au lieu de la probabilité de faire des virages aux intersections indiquées).
- Utilisation de routes aléatoires.

Dans ce cas, on utilisera les routes aléatoires. Il y a un script Python développé dans le but de produire des routes aléatoires, son nom est randomTrips.py. Actuellement c'est la méthode la plus recommandée pour obtenir cette fonctionnalité.

Alors, on réouvre l'invité de commande, et on tape la première commande suivante, comme on peut le voir dans la figure 3.9.

```
python C :/Users/NaSsIm/Desktop/sumo-0.19.0/tools/trip/randomTrips.py -n bejaia.net.xml -e 100 -l
```

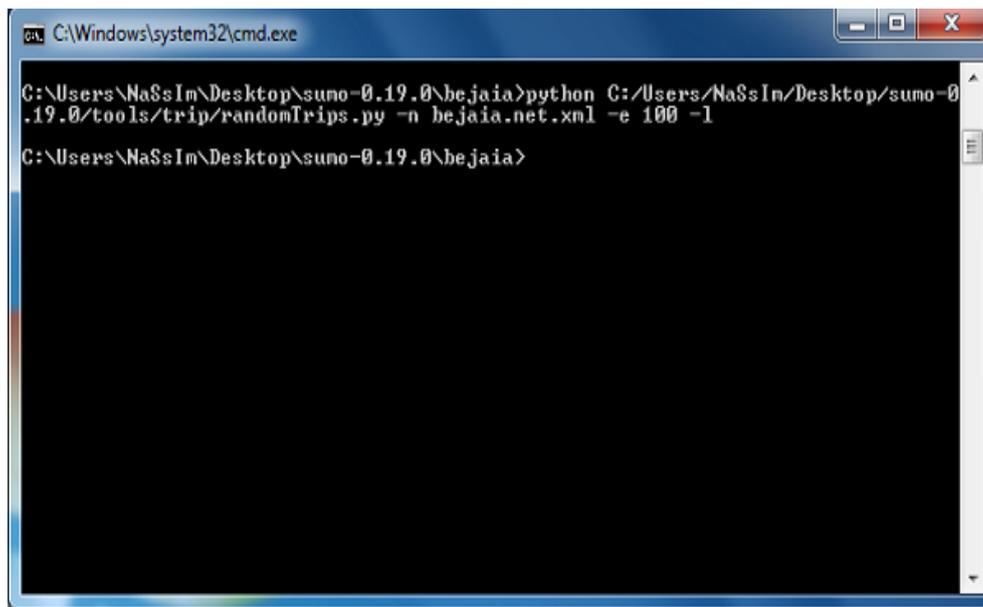
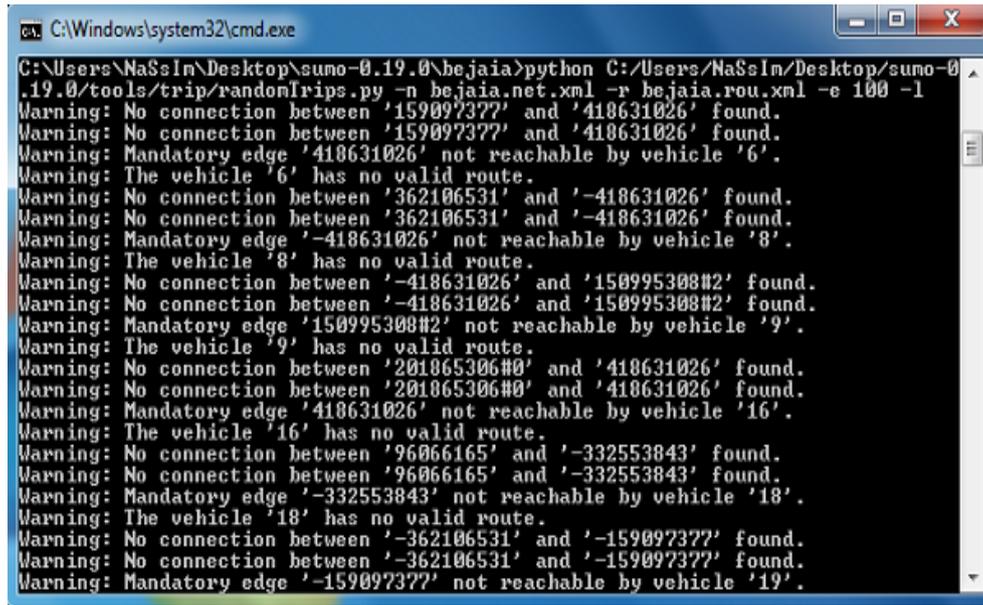


FIGURE 3.9 – Génération de bejaia.rou.xml

Puis on procède avec cet dernière commande comme on peut le voir sur la figure 3.10 :

```
python C :/Users/NaSsIm/Desktop/sumo-0.19.0/tools/trip/randomTrips.py -n bejaia.net.xml -r bejaia.rou.xml -e 100 -l
```



```
C:\Windows\system32\cmd.exe
C:\Users\NaSsIm\Desktop\sumo-0.19.0\bejaia>python C:/Users/NaSsIm/Desktop/sumo-0.19.0/tools/trip/randomtrips.py -n bejaia.net.xml -r bejaia.rou.xml -e 100 -l
Warning: No connection between '159097377' and '418631026' found.
Warning: No connection between '159097377' and '418631026' found.
Warning: Mandatory edge '418631026' not reachable by vehicle '6'.
Warning: The vehicle '6' has no valid route.
Warning: No connection between '362106531' and '-418631026' found.
Warning: No connection between '362106531' and '-418631026' found.
Warning: Mandatory edge '-418631026' not reachable by vehicle '8'.
Warning: The vehicle '8' has no valid route.
Warning: No connection between '-418631026' and '150995308#2' found.
Warning: No connection between '-418631026' and '150995308#2' found.
Warning: Mandatory edge '150995308#2' not reachable by vehicle '9'.
Warning: The vehicle '9' has no valid route.
Warning: No connection between '201065306#0' and '418631026' found.
Warning: No connection between '201065306#0' and '418631026' found.
Warning: Mandatory edge '418631026' not reachable by vehicle '16'.
Warning: The vehicle '16' has no valid route.
Warning: No connection between '96066165' and '-332553843' found.
Warning: No connection between '96066165' and '-332553843' found.
Warning: Mandatory edge '-332553843' not reachable by vehicle '18'.
Warning: The vehicle '18' has no valid route.
Warning: No connection between '-362106531' and '-159097377' found.
Warning: No connection between '-362106531' and '-159097377' found.
Warning: Mandatory edge '-159097377' not reachable by vehicle '19'.
```

FIGURE 3.10 – Génération des routes aléatoires dans SUMO

3.4.1 Préparation des fichiers avant la simulation

Dans cette étape, on copie les fichiers que nous avons générés dans les sections précédentes vers le dossier VANETSIm qui est dans OMNeT++. Après cela, Nous devons modifier les fichiers de configuration de VANETSIm, alors on ouvre avec gedit les fichiers erlangen.launchd.xml et erlangen.sumo.cfg et on écrit le nom des fichiers que nous avons copiés avant : bejaia.net.xml, bejaia.rou.xml et bejaia.poly.xml, comme il est illustré dans les figures 3.11, 3.12.

```

1 <?xml version="1.0" encoding="iso-8859-1"?>
2
3 <configuration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://sumo.sf.net/xsd/sumoConfiguration.xsd">
4
5   <input>
6     <net-file value="bejaia.net.xml"/>
7     <route-files value="bejaia.router.xml"/>
8     <additional-files value="bejaia.poly.xml"/>
9   </input>
10
11   <time>
12     <begin value="0"/>
13     <end value="1000"/>
14     <step-length value="0.1"/>
15   </time>
16
17   <gui_only>
18     <start value="true"/>
19   </gui_only>
20
21 </configuration>
22

```

FIGURE 3.11 – Modification du fichier erlangen.sumo.cfg

```

1 <?xml version="1.0"?>
2 <!-- debug config -->
3
4 <launch>
5   <copy file="bejaia.net.xml" />
6   <copy file="bejaia.router.xml" />
7   <copy file="bejaia.poly.xml" />
8   <copy file="bejaia.sumo.cfg" type="config" />
9 </launch>
10

```

FIGURE 3.12 – Modification du fichier erlangen.launchd.xml

Et enfin on renomme ces deux derniers fichiers par : bejaia.sumo.cfg et bejaia.launchd.xml.

Pour pouvoir procéder à la simulation on doit exécuter SUMO et OMNeT++ à la fois, c'est pour ça que VEINS est livré avec un petit script python qui fait une connexion TCP proxy

entre OMNET ++ et SUMO. Pour le faire, ce script commence une nouvelle copie de la simulation de SUMO pour la simulation de chaque OMNeT connexion. Ainsi, dans l'invite de commande on tape la commande suivante :

```
/D/ETUDES/Simulation/omnetpp-4.6/VANETSim/sumo-launchd.py -vv -c /C/-  
Users/NaSsIm/Desktop/sumo-0.19.0/bin/sumo-gui.exe
```

Pour pouvoir le démarrer on entre d'abord dans le chemin de Veins sur OMNeT++, le script va écouter sur le port 9999, comme il est illustré à la figure 3.13 , et on attend le début de la simulation pour commencer.

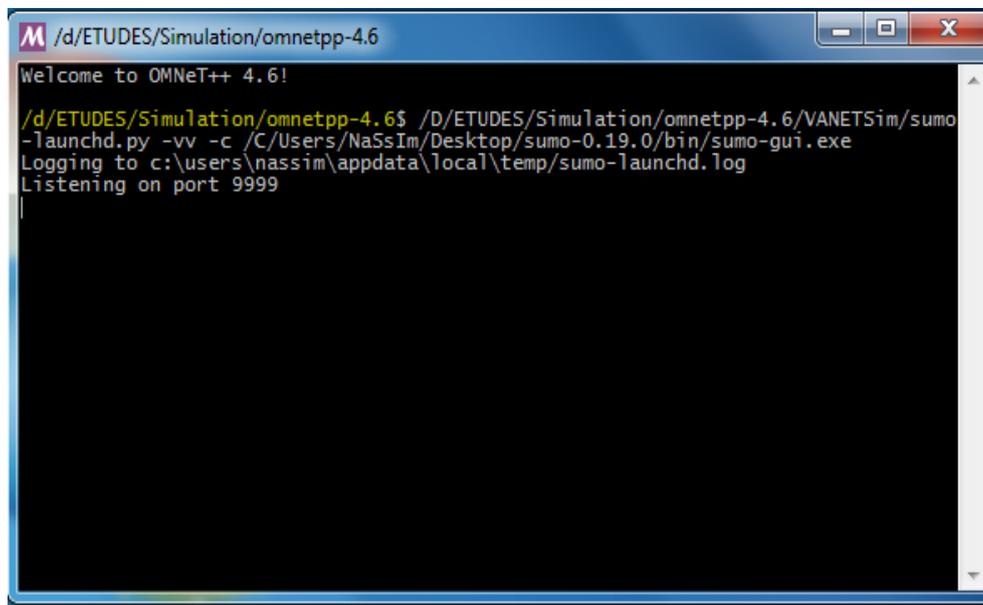


FIGURE 3.13 – Ouverture et écoute sur le port TCP 9999

On laisse cette fenêtre ouverte et on ouvre OMNeT++ après avoir créer notre projet VANETSim qui est composés de VEINS et INET.

Alors on sera en mesure d'exécuter les exemples VEINS-INET (VEINS/exemples/INET), pour pouvoir utilisé les protocoles de routage de INET sous VEINS.

Enfin on a qu'à modifier le fichier omnetpp.ini avec (playgroundSizeX = 25000mm et playgroundSizeY = 20000mm et aussi manager.launchConfig = xmldoc("bejaia.launchd.xml")) comme la figure 3.14 le montre.

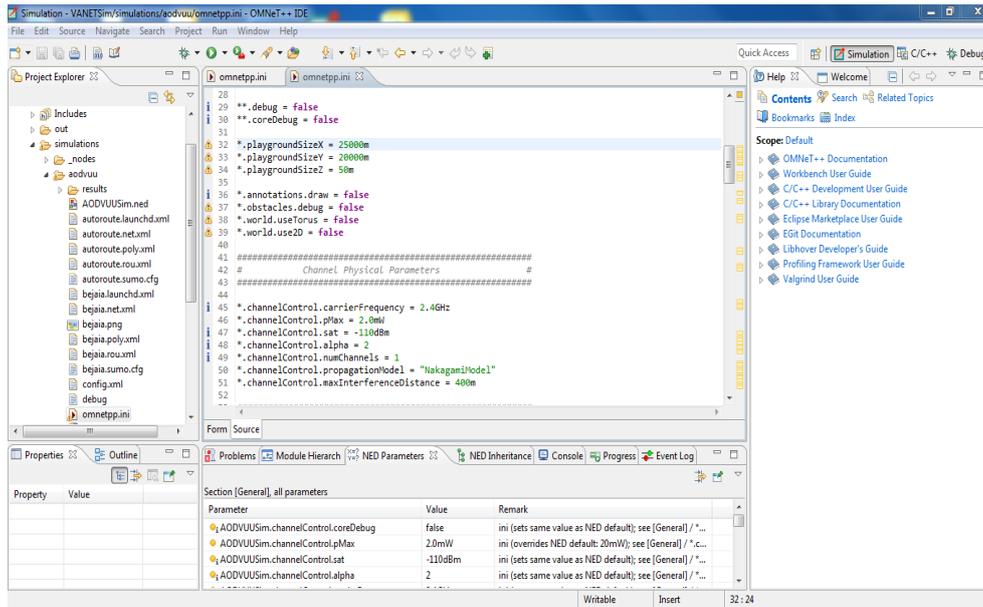


FIGURE 3.14 – Modification du fichier omnetpp.ini

On clique alors sur RUN pour pouvoir démarrer notre exemple de simulation et on obtiendra la figure 3.15.

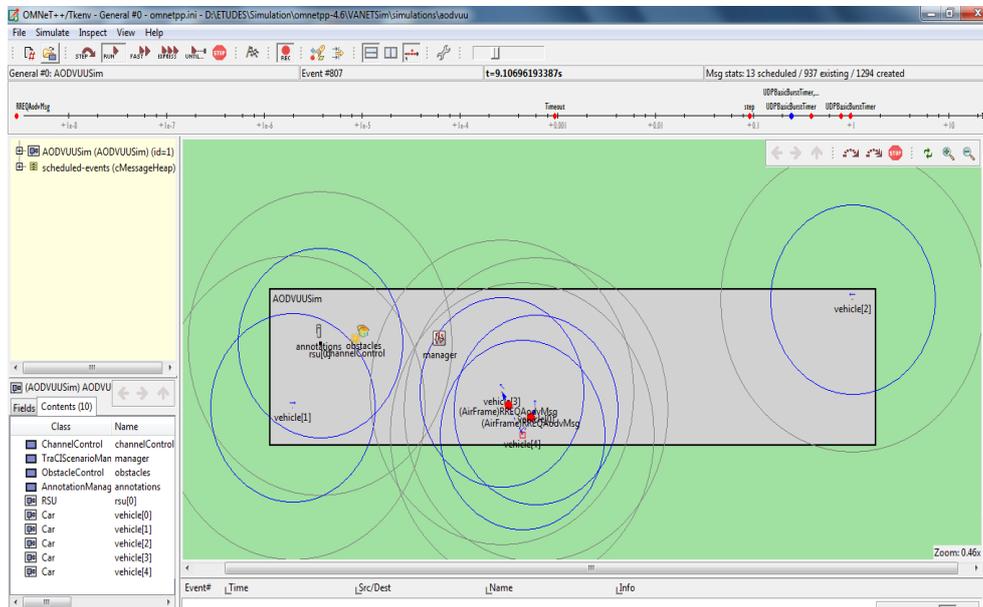


FIGURE 3.15 – Fenêtre de la simulation de OMNeT++

Une autre fenêtre de SUMO sera lancé automatiquement comme illustré sur la figure 3.16.

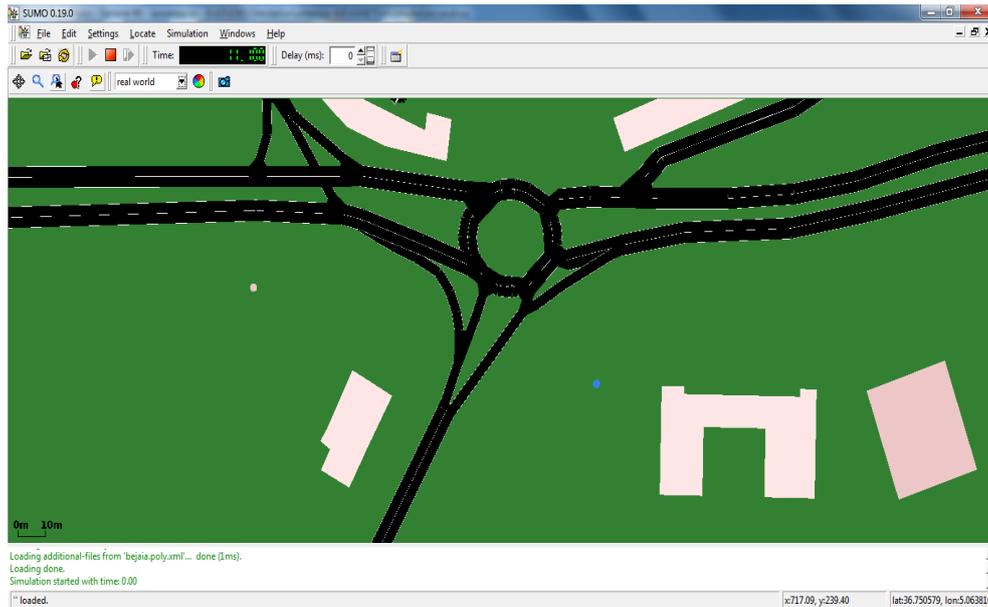


FIGURE 3.16 – Fenêtre de la simulation de SUMO

Pour pouvoir vérifier le bon fonctionnement de SUMO avec VEINS on regarde l'invité de commande qu'on a laissé ouvert pour voir le résultat sur la figure 3.17 :

```

M /d/ETUDES/Simulation/omnetpp-4.6
Welcome to OMNeT++ 4.6!
/d/ETUDES/Simulation/omnetpp-4.6$ /D/ETUDES/Simulation/omnetpp-4.6/VANETSIm/sumo
-launchd.py -vv -c /C/Users/NaSsIm/Desktop/sumo-0.19.0/bin/sumo-gui.exe
Logging to c:\users\nassim\appdata\local\temp\sumo-launchd.log
Listening on port 9999
Connection from 127.0.0.1 on port 55462
Handling connection from 127.0.0.1 on port 55462
Got TraCI message of length 2
Got TraCI command of length 1
Got TraCI command 0x0
Got CMD_GETVERSION
Got TraCI message of length 314
Got TraCI command of length 309
Got TraCI command 0x75
Got CMD_FILE_SEND for "sumo-launchd.launch.xml"
Got CMD_FILE_SEND with data "<launch>
<copy file="autoroute.net.xml"/>
<copy file="autoroute.rou.xml"/>
<copy file="autoroute.poly.xml"/>
<copy file="autoroute.sumo.cfg" type="config"/>
<basedir path="D:/ETUDES/Simulation/omnetpp-4.6/VANETSIm/simulations/aodvuu/" />
>
<seed value="0"/>
    
```

FIGURE 3.17 – Démarrage de la simulation sur le terminal

Un nouveau dossier qui porte le nom de « résultats » va apparaître dans le dossier du protocole de routage comme sur cette figure 3.18.

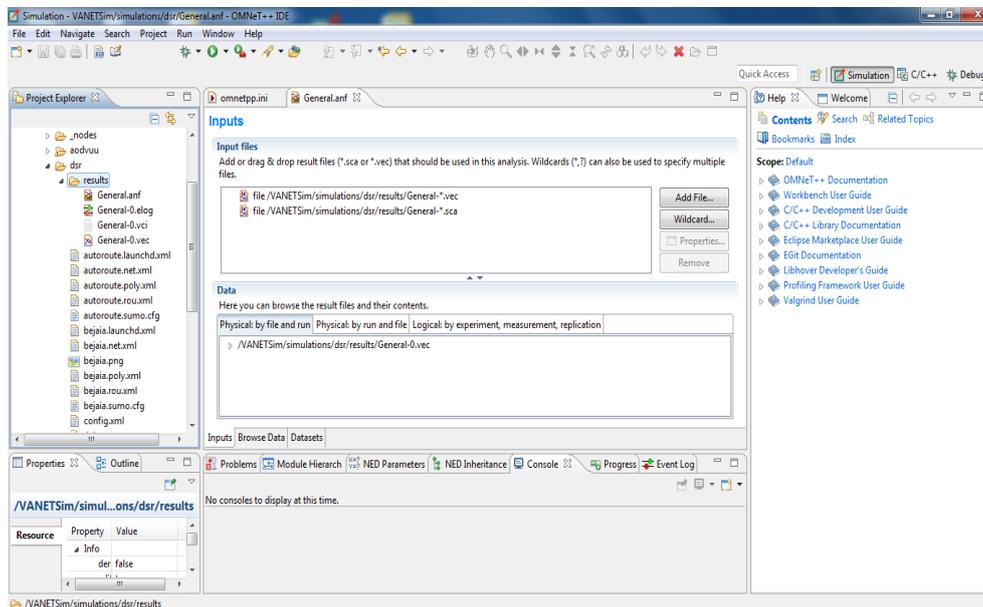


FIGURE 3.18 – Résultats de la simulation

Les mêmes étapes ont été suivies pour le deuxième cas de simulation.

3.4.2 Métriques de simulation

Dans notre étude, nous avons pris en compte les métriques suivantes :

3.4.2.1 Le débit

Le débit est une mesure de la quantité de données numériques transmises par unité de temps :

Débit = quantité de données transmises / le temps total.

3.4.2.2 Le taux de paquet perdu

Le taux de perte des paquets et le rapport entre le nombre de paquets perdus et le nombre total de paquets envoyés :

Taux de perte = le nombre de paquets perdus / le nombre de paquets émis.

3.4.3 paramètres de simulation

Avant de commencer les tests, nous allons présenter les scénarios de mouvement que nous avons utilisés :

Simulateur	OMNeT++4.6
Protocoles	AODV, DSR
Temps de simulation	100s
Nombre des nœuds	par défaut
Période entre nœuds	1s
Terrain de simulation	X=25000m, Y=20000m, Z=50m
MAC	1609-4
Physical Layer	802.11p
Type de Mobilité	TraCIDemo11P
Zone de Trafic	Urbain/Autoroute

TABLE 3.3 – Scénarios de mouvement utilisés.

3.5 Résultats de la simulation et discussion

3.5.1 Premier cas de simulation (milieu urbain)

Les figures 3.19, 3.20 montrent le taux de perte de paquet avec les protocoles AODV et DSR.

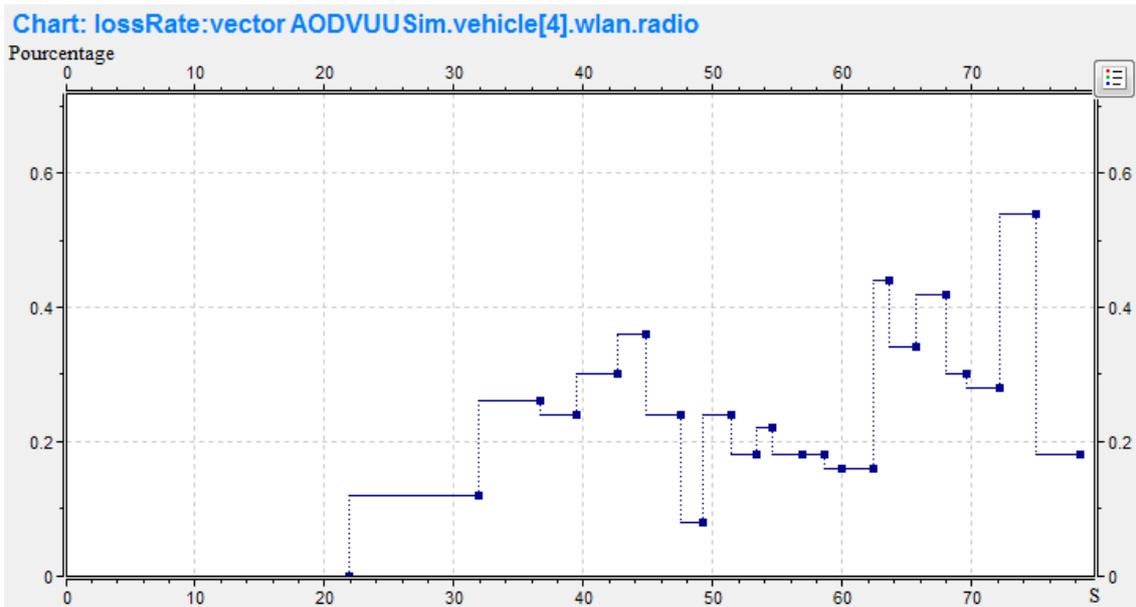


FIGURE 3.19 – Taux de paquets perdus en milieu urbain avec le protocole AODV

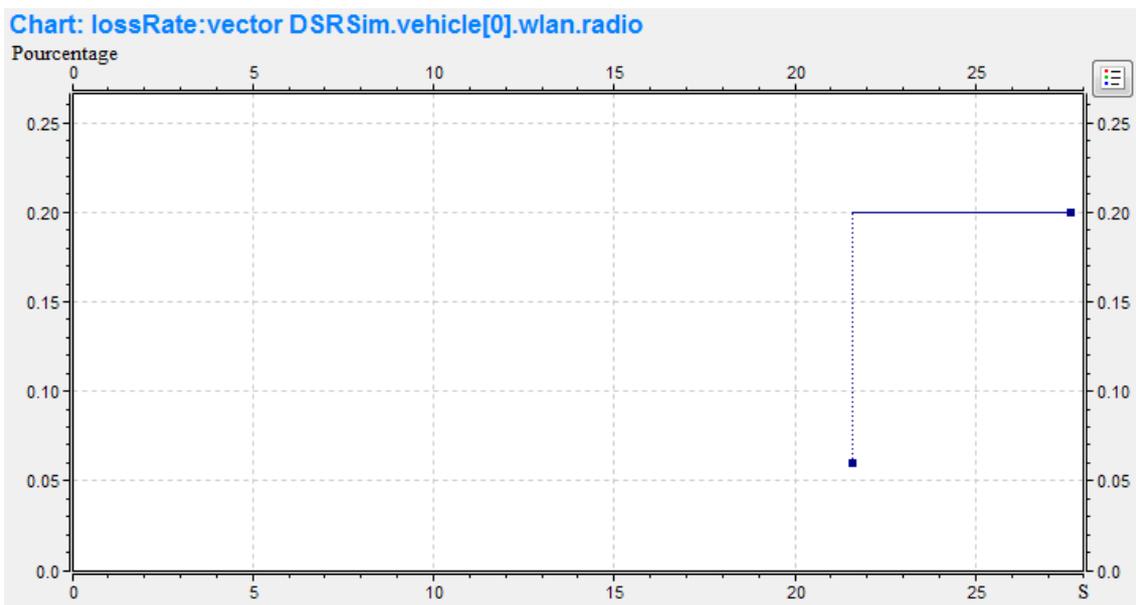


FIGURE 3.20 – Taux de paquets perdus en milieu urbain avec le protocole DSR

Les figures 3.21, 3.22 montrent le débit avec les protocoles AODV et DSR.

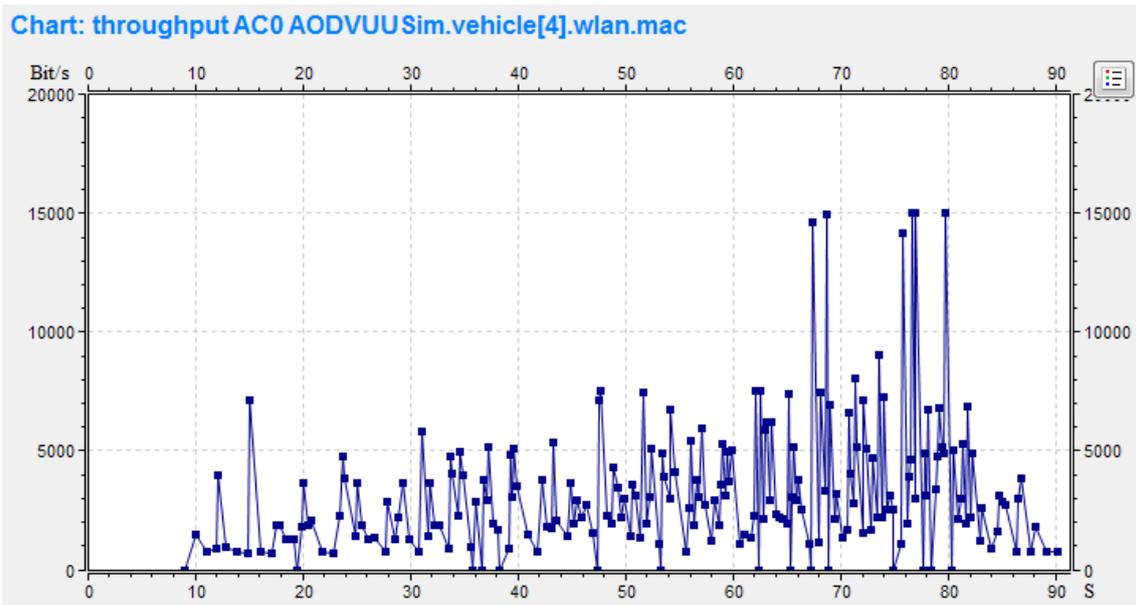


FIGURE 3.21 – Débit en milieu urbain avec le protocole AODV

3.5.1.1 Le débit avec le protocole DSR

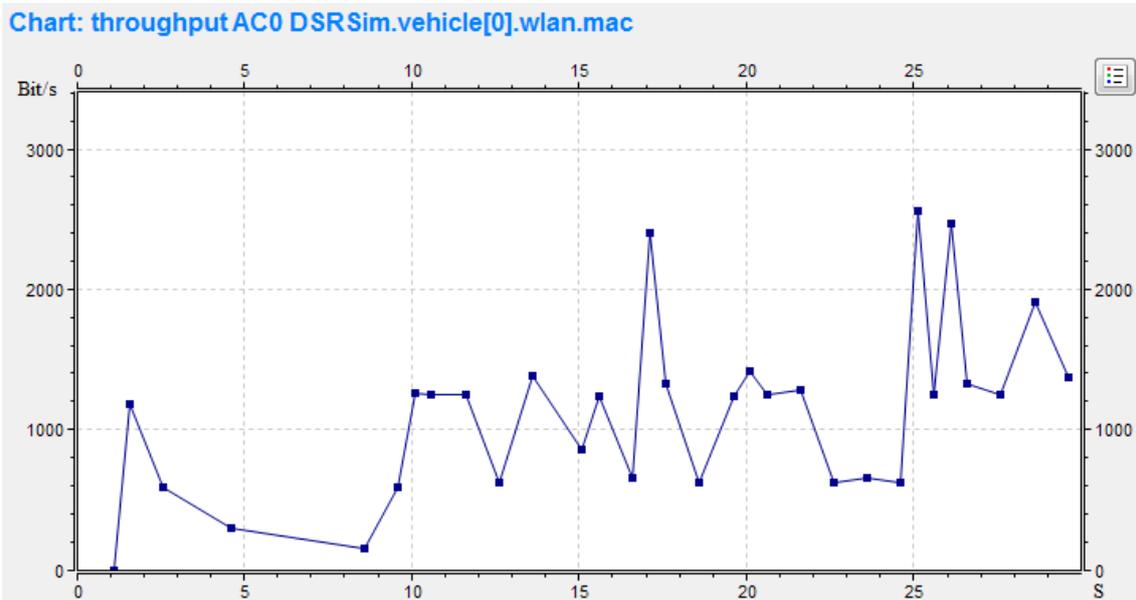


FIGURE 3.22 – Débit en milieu urbain avec le protocole DSR

3.5.2 Deuxième cas de simulation (milieu autoroutier)

Les figures 3.23, 3.24 montrent le taux de perte de paquet avec les protocoles AODV et DSR.

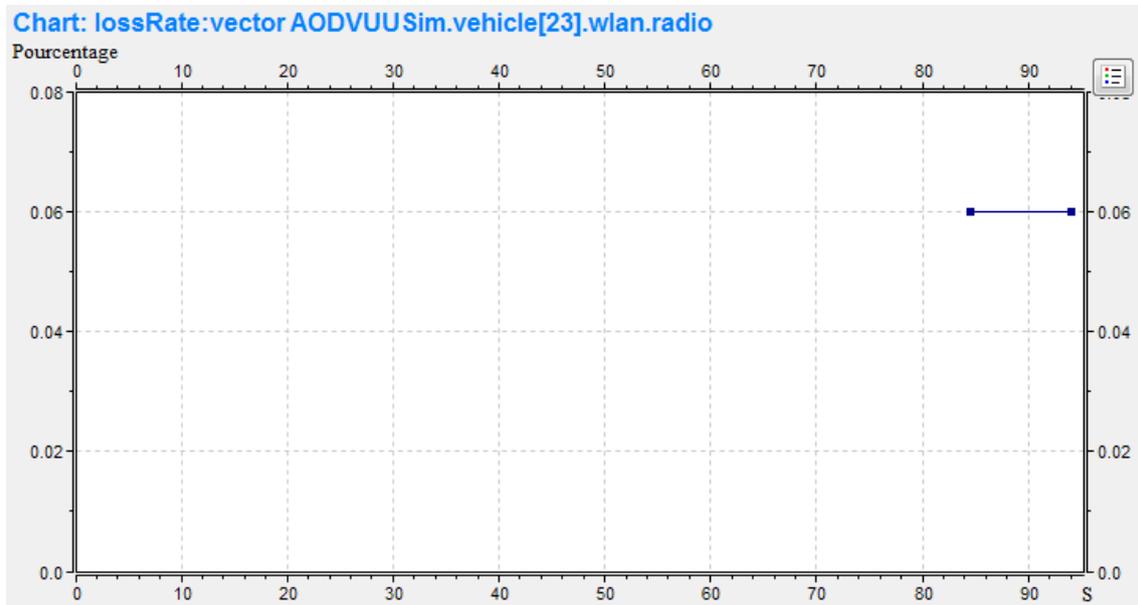


FIGURE 3.23 – Taux de paquets perdus en milieu autoroutier avec le protocole AODV

Les figures 3.25, 3.26 montrent le débit avec les protocoles AODV et DSR.

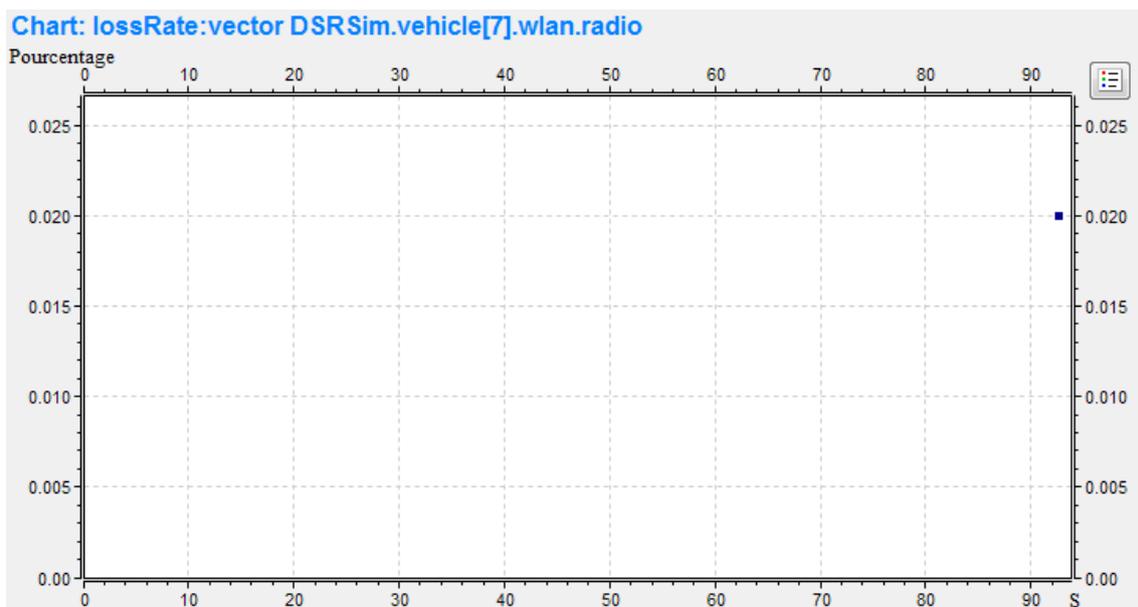


FIGURE 3.24 – Taux de paquets perdus en milieu autoroutier avec le protocole DSR

3.5.2.1 Le débit avec le protocole AODV

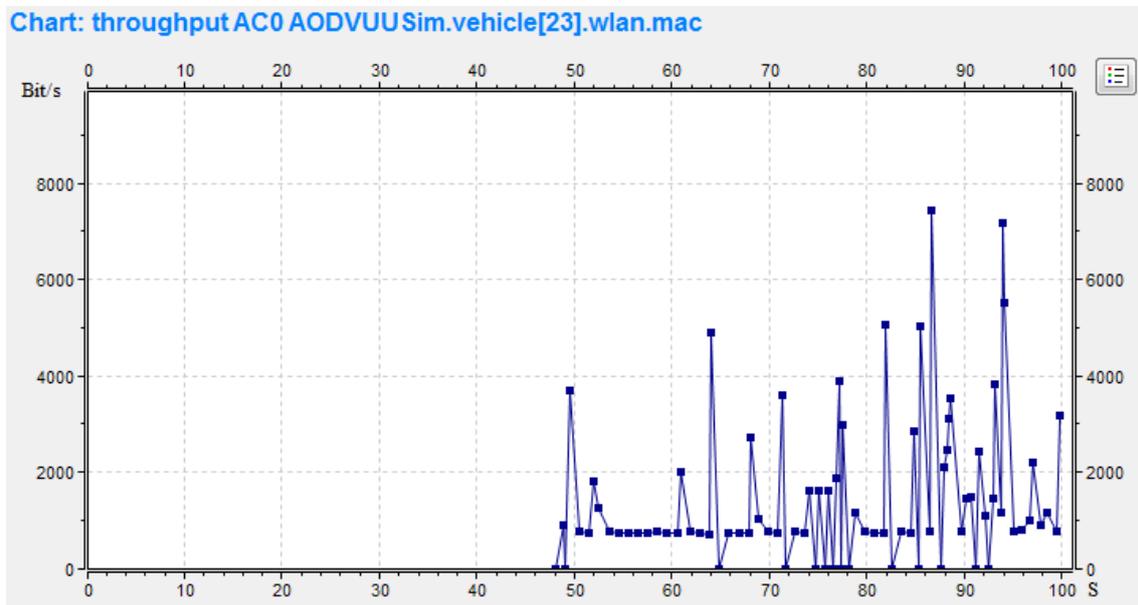


FIGURE 3.25 – Débit en milieu autoroutier avec le protocole AODV

3.5.2.2 Le débit avec le protocole DSR

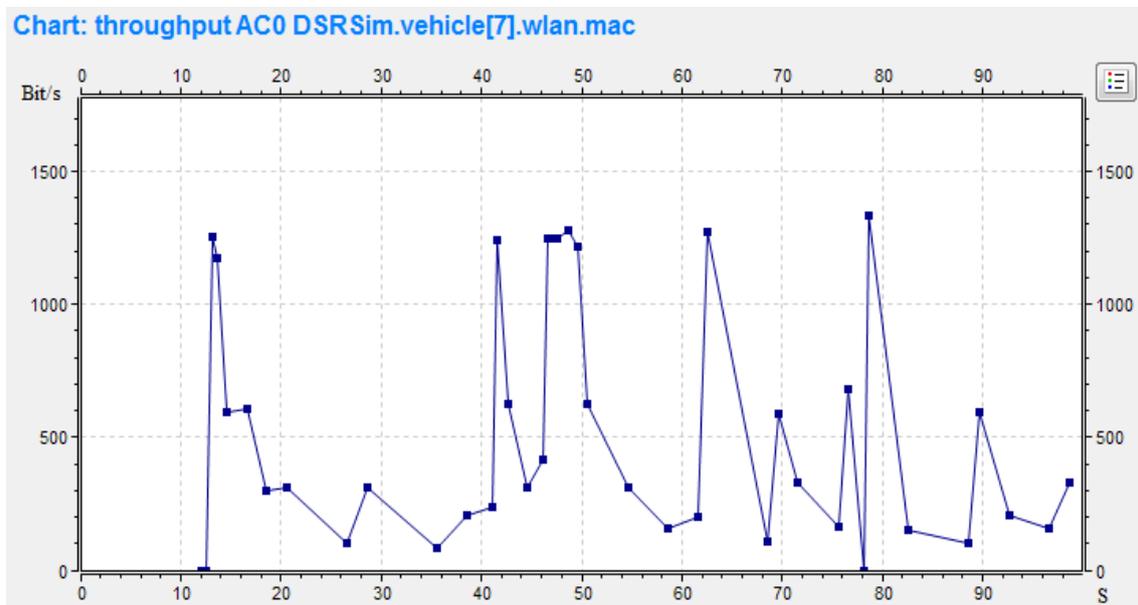


FIGURE 3.26 – Débit en milieu autoroutier avec le protocole DSR

D'après les figures 1.19, 1.20, 1.23, 1.24 on remarque que le taux de paquet perdu est plus grand avec le protocole AODV par contre il est négligeable avec le protocole DSR. Et d'après les figures 1.21, 1.22, 1.25, 1.26 on remarque que le débit est plus grand avec le protocole AODV par rapport au protocole DSR.

Conclusion

Dans ce chapitre nous avons évalué les performances des protocoles de routage DSR et AODV des réseaux véhiculaires dans deux milieux différents en terme de débit et de taux de paquet perdu, urbain sur un rondpoint de la ville de Bejaia et autoroutier sur l'autoroute Est-Ouest en utilisant deux métriques, on a pu constater que pour la majorité des cas le protocole DSR minimise au maximum la perte de paquet et il est plus performant en toutes circonstances et plus adéquat pour notre réseau dans les deux cas d'études.

Conclusion Générale et Perspectives

Le développement des nouvelles technologies a favorisé une formidable évolution des réseaux véhiculaires. Cette évolution vise à rendre les réseaux plus sûrs, plus efficaces, plus fiables et plus écologiques aussi bien du point de vue de l'industrie automobile que des opérateurs de réseaux et services. Les réseaux véhiculaires sont en effet une classe émergente des réseaux mobiles Ad Hoc, permettant des échanges de données entre véhicules ou encore véhicules et infrastructure. Ils suscitent un intérêt certain dans le but d'améliorer la sécurité et l'efficacité des transports routiers ainsi que l'amélioration du confort de l'utilisateur en lui fournissant une multitude des services d'information, d'aide à la décision, de guidage et d'accès à internet.

Notre projet d'étude est concentré sur l'évaluation des performances des protocoles de routage DSR et AODV pour les réseaux véhiculaires en milieu urbain et autoroutier en terme de taux de paquets perdus et de débit.

Pour atteindre notre objectif et obtenir des résultats de simulation proches de la réalité, nous nous sommes appuyées sur la simulation on utilisons le simulateur réseau OMNeT++ et le simulateur de mobilité SUMO. On a pu constater que pour les deux cas d'études le protocole DSR est plus performant et plus efficace.

À travers ce mémoire, nous avons été en mesure de comprendre et assimiler les différentes étapes par lesquelles un projet de recherche doit progresser. Nous avons également acquis une expérience interne et un bon aperçu des méthodes de travail d'un chercheur, avec une connaissance approfondie dans ce domaine. Tout au long de ce travail, nous avons utilisés des techniques et des nouvelles idées prometteuses qui vont nous permettre dans l'avenir proche

d'élargir ce sujet de recherche et s'attaquer aux problématiques plus pointues dans le domaine de réseau de VANET et potentiellement autres réseaux de nature proche avec des solutions plus génériques.

A

Annexe

A.1 Installation de OMNeT++4.6 sous Windows 7

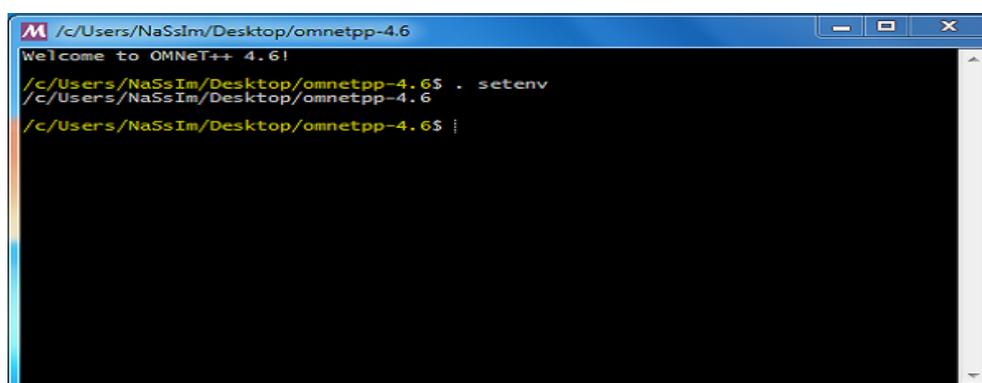
Les étapes suivies pour installer OMNeT++ sous Windows 7 :

tout d'abord on se rend sur la page de OMNeT++ pour le télécharger :

<https://omnetpp.org/omnetpp/>

alors on le télécharge en format.rar, après la fin du téléchargement on le décompresse en dossier. Avec l'invite de commande(Mingwenv) qui est dans le dossier **omnetpp-4.6** on entre la commande :

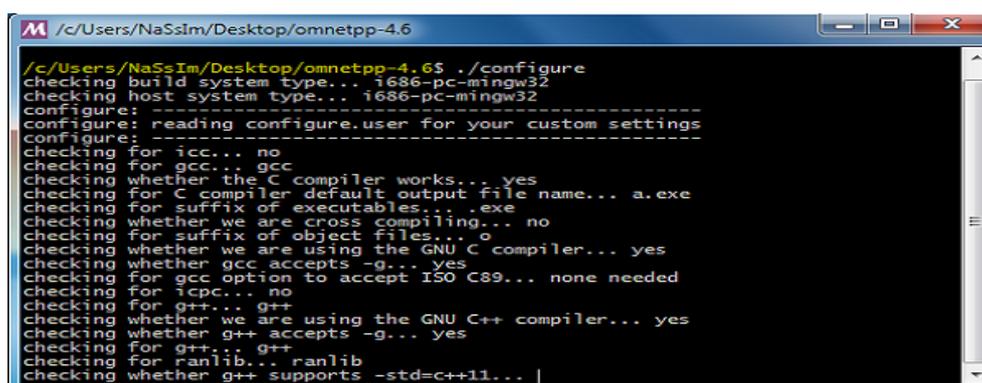
. **setenv** comme on peut le voir sur la figure A.1.



```
M /c/Users/NaSsIm/Desktop/omnetpp-4.6
Welcome to OMNeT++ 4.6!
/c/Users/NaSsIm/Desktop/omnetpp-4.6$ . setenv
/c/Users/NaSsIm/Desktop/omnetpp-4.6
/c/Users/NaSsIm/Desktop/omnetpp-4.6$ :
```

FIGURE A.1 – Exécution de la commande . **setenv**

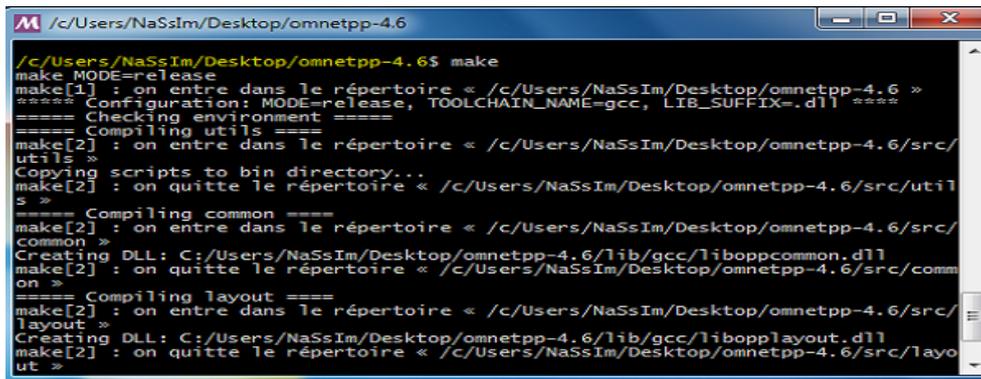
Puis on tape la commande : **./configure** comme sur la figure A.2.



```
M /c/Users/NaSsIm/Desktop/omnetpp-4.6
/c/Users/NaSsIm/Desktop/omnetpp-4.6$ ./configure
checking build system type... i686-pc-mingw32
checking host system type... i686-pc-mingw32
configure: reading configure.user for your custom settings
configure:
-----
checking for icc... no
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.exe
checking for suffix of executables... .exe
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking for icpc... no
checking for g++... g++
checking whether we are using the GNU C++ compiler... yes
checking whether g++ accepts -g... yes
checking for g++... g++
checking for ranlib... ranlib
checking whether g++ supports -std=c++11... |
```

FIGURE A.2 – Exécution de la commande **./configure** de OMNeT++

Après on tape la commande : **make** pour pouvoir installer OMNeT++ comme illustré dans la figure A.3.



```
/c/Users/NaSsIm/Desktop/omnetpp-4.6$ make
make MODE=release
make[1] : on entre dans le répertoire « /c/Users/NaSsIm/Desktop/omnetpp-4.6 »
==== Configuration: MODE=release, TOOLCHAIN_NAME=gcc, LIB_SUFFIX=.dll ====
==== Checking environment ====
==== Compiling utils ====
make[2] : on entre dans le répertoire « /c/Users/NaSsIm/Desktop/omnetpp-4.6/src/
utils »
Copying scripts to bin directory...
make[2] : on quitte le répertoire « /c/Users/NaSsIm/Desktop/omnetpp-4.6/src/uti
l »
==== Compiling common ====
make[2] : on entre dans le répertoire « /c/Users/NaSsIm/Desktop/omnetpp-4.6/src/
common »
Creating DLL: C:/Users/NaSsIm/Desktop/omnetpp-4.6/lib/gcc/liboppcommon.dll
make[2] : on quitte le répertoire « /c/Users/NaSsIm/Desktop/omnetpp-4.6/src/comm
on »
==== Compiling layout ====
make[2] : on entre dans le répertoire « /c/Users/NaSsIm/Desktop/omnetpp-4.6/src/
layout »
Creating DLL: C:/Users/NaSsIm/Desktop/omnetpp-4.6/lib/gcc/libopplayout.dll
make[2] : on quitte le répertoire « /c/Users/NaSsIm/Desktop/omnetpp-4.6/src/layo
ut »
```

FIGURE A.3 – Exécution de la commande **make** de OMNeT++

Et enfin on tape la commande : **omnetpp** pour lancé OMNeT++.

A.2 Importation d'un nouveau projet sous OMNeT++

Pour pouvoir importer un nouveau projet sous OMNeT++, on se rend dans **File**, puis **Import** et on choisi le premier dossier **General** et on clique sur **Existing Project into Workspace** comme on peut le voir sur la figure A.4.

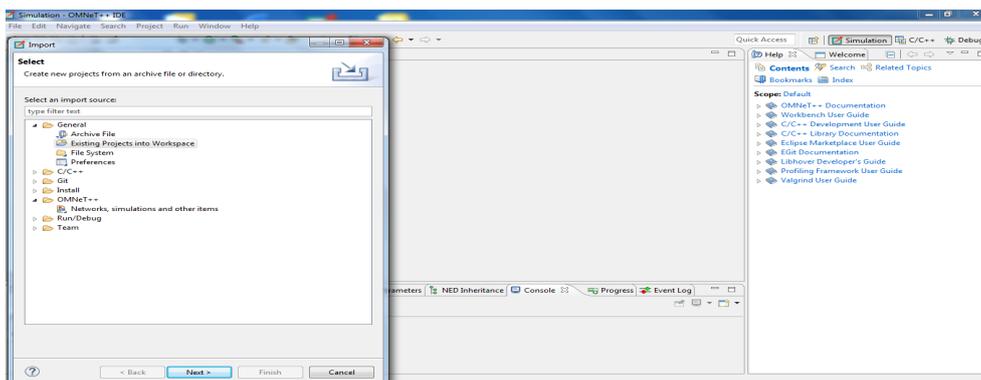


FIGURE A.4 – Première étape d'importation d'un nouveau projet sous OMNeT++

Après on clique sur **Browse** et on choisi le fichier a importer (Veins 4.4 par exemple), puis on coche **Copy projects into workspace** et on clique sur **Finish** comme sur la figure A.5.

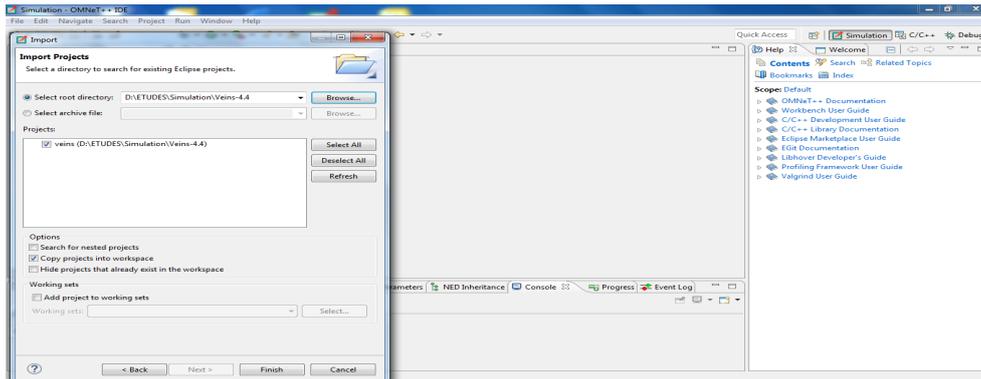


FIGURE A.5 – Deuxième étape d'importation d'un nouveau projet sous OMNeT++

A.3 Configuration de la variable d'environnement pour SUMO

Pour pouvoir exécuter un exemple sur SUMO on doit d'abord ajouter le chemin suivant : `D :/ETUDES/Simulation/sumo-0.19.0/bin` dans la variable PATH comme sur la figure A.6 suivante :

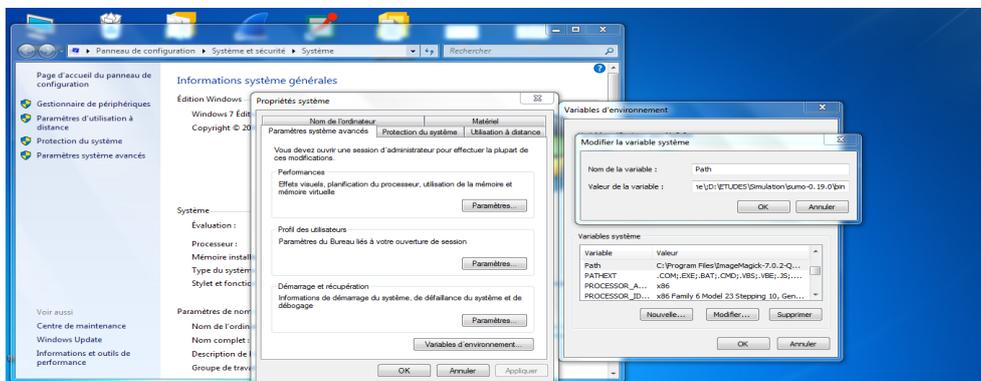


FIGURE A.6 – Ajout de la variable PATH de SUMO

Bibliographie

- [1] S.Rimour, « Généralités sur les réseaux : chapitre 1 », IUT C.F département informatique, mars 2002.
- [2] Hassnaa Moustafa « Routage unicast et multicast dans les réseaux mobiles Ad Hoc » Thèse de doctorat. Ecole Nationale Supérieure des Télécommunications de Paris, 2004.
- [3] Meraihi Yassine « Routage dans les réseaux véhiculaires (VANET) » Thèse de doctorat. Université M'hamed Bougara de Boumerdès, 2011.
- [5] R.Meraihi « Gestion de la qualité de service et contrôle de topologie dans les réseaux Ad Hoc » Thèse de doctorat. Ecole Nationale Supérieure des Télécommunications de Paris, 2005.
- [9] Hubaux Jean-Pierre, Capkun Srdjan and Luo Jun, The security and privacy of smart vehicles, IEEE Security and Privacy, No 3, pp. 49-55, 2004.
- [10] Ali Kahina Ait « Modélisation et étude de performances dans les réseaux VANET » Thèse de doctorat. Université de Technologie de Belfort-Montbéliard, 2012.
- [11] Jiang Daniel and Delgrossi Luca, IEEE 802.11 p : Towards an international standard for wireless access in vehicular environments, Vehicular Technology Conference, pp. 2036-2040, 2008.

[12] IEEE 802.11 Working Group, 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, IEEE Std 802.11, 2012.

[13] Zeadally Sherali, Hunt Ray, Chen Yuh-Shyan, Irwin A. and Hassan A., Vehicular ad hoc networks (VANETS) : status, results, and challenges, Telecommunication Systems, Vol. 50, No 4, pp. 217-241, 2012.

[14] Hasbullah Halabi, Ahmed Soomro Irshad and Ab Manan Jamalul-lail, Denial of service (dos) attack and its possible solutions in VANET, World Academy of Science, Engineering and Technology (WASET), Vol. 65, pp. 411 -415, 2010.

[15] Perkins Charles E. and Bhagwat Pravin, Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers, ACM SIGCOMM computer communication review, Vol. 24, No. 4, pp. 234-244, 1994.

[16] Sen Thomas and Jacquet Philippe, Optimized link state routing protocol (OLSR), No. RFC3626, 2003.

[17] Sedrati M., Aouragh L., Guettala L. and Bilami A., Etude des Performances des Protocoles de Routage dans les Réseaux Mobiles Ad-Hoc, 4 th Internationnal Conference on Computer integrated manufacturing CIP, 2007.

[18] Johnson David B., Maltz David A. and Hu Yih-Chun, The dynamic source routing protocol for mobile ad hoc networks, IETF MANET Working Group Internet Draft, 2004.

[19] Perkins Charles, Belding-Royer Elizabeth and Das Samir, Ad hoc on-demand distance vector (AODV) routing, 2003.

- [20] Park Vincent D. and Corson M., A performance comparison of the temporally ordered routing algorithm and ideal link-state routing, Computers and Communications, ISCC'98, Proceedings Third IEEE Symposium on, IEEE, pp. 592-598, 1998.
- [21] Wang Lan and Olariu Stephan, A two-zone hybrid routing protocol for mobile ad hoc networks, Parallel and Distributed Systems, IEEE Transactions on, Vol. 15, No 12, pp. 1105-1116, 2004.
- [22] Haas Zygmunt J., Pearlman Marc R. and Samar Prince, The zone routing protocol (ZRP) for ad hoc networks, draft-ietf-manet-zone-zrp-04. txt, 2002.
- [24] C. E. Perkins and E. Belding Royer, « Ad Hoc on Demand Distance Vector (AODV) Algorithm », in Proceedings of Systems and Applications(WMCSA'99) , pp. 90-100, 1999.
- [26] Maxim Raya and Jean-Pierre Hubaux : « Securing vehicular ad hoc networks » Journal of Computer Security 39-68 39 IOS Press, 2007.
- [27] Jerbi Moez « Protocoles pour les communications dans les réseaux de véhicules en environnement urbain : Routage et GeoCast basés sur les intersections » Thèse de doctorat. Evry-Val d'Essonne, 2008.
- [28] Yasinac Alec and Carter Stephen, Secure Position Aided Ad hoc Routing, Florida State University, 2002.
- [29] Karp Brad and Kung Hsiang-Tsung, GPSR : Greedy perimeter stateless routing for wireless networks, Proceedings of the 6th annual international conference on Mobile computing and networking, ACM, pp. 243-254, 2000.
- [30] Ko Young-Bae and Vaidya Nitin-H, Location-Aided Routing (LAR) in mobile ad hoc networks, Wireless Networks Vol. 6, No 4, pp. 307-321, 2000.

[31] Basagani Stefano, Chlamtac Imrich, Syrotiuk Violet R. and Woodward B. A., A distance routing effect algorithm for mobility (DREAM), Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking, ACM, pp. 76-84, 1998.

[32] Johnson David, Hu Y. and Maltz D., The dynamic source routing protocol (DSR) for mobile ad hoc networks for IPv4, No. RFC 4728, 2007.

[33] Perkins Charles, Belding-Royer Elizabeth and Das Samir, Ad hoc on-demand distance vector (AODV) routing, 2003.

[37] A.Varga. Using the OMNeT++ discrete event simulation system in education). IEEE Transactions on Education, 1999.

[39] D.Eckhoff and C.Sommer. A multi-channel IEEE 1609.4 and 802.11p EDCA model for the veins framwork). In 5th ACM/ICST, 2012.

[40] Kopke, A.Swigulski, M.Wessel, K.Willkomm, D.Haneveld, P.K.Parker, T.Visser, O.Lichte, and H.S.Valentin. Simulating wireless and mobile networks in OMNeT++ - The MiXiM vision. PhD thesis, ICST Brussels Belgium, 2008.

Webographie

[4] Anne Gégout « Les réseaux Ad Hoc ». <http://www.tdf.fr/medias/view/?id=704> (consulté en Mai 2016).

[6] Claude Chaidet, Isabelle Guérin Lassous « Routage QoS et réseau Ad hoc ». <http://hal.inria.fr/docs/00/07/18/86/PDF/RR-4700.pdf> (consulté en Mai 2016).

[7] Jérôme Vandermeersch « Hybridation entre les modes Ad Hoc et infrastructure dans les réseaux de type Wifi-Fi ». <http://code.ulb.ac.be/dbfiles/media832.pdf> (consulté en Mai 2016).

[8] Ken Leonard, Connected Vehicle Pilots Coming to a Region Near You, <https://www.dot.gov/fastlane/connected-vehicle-pilots-coming-region-near-you> (consulté en Mai 2016).

[23] Claude Chaidet, Isabelle Guérin Lassous « Routage QoS et réseau Ad hoc » <http://hal.inria.fr/docs/00/07/18/86/PDF/RR-4700.pdf> (consulté en Mai 2016).

[25] C.E.Perkins, E.Belding Royer and S.Das, « Ad hoc on demand distance vector (AODV) routing ». <http://www.ietf.org/rfc/rfc3561.txt> (consulté en Mai 2016).

[34] Protocole de routage. http://fr.wikipedia.org/wiki/Ad-hoc_on-demand_distance_vector (consulté en Mai 2016).

[35] NS. <https://en.wikipedia.org/wiki/Ns> (consulté en Mai 2016).

[36] OPNET. <https://en.wikipedia.org/wiki/OPNET> (consulté en Mai 2016).

[38] INET Framwork. <http://inet.omnetpp.org/index/php?n=Main.HomePage> (consulté en Mai 2016).

[41] VEINS. <http://veins.car2x.org> (consulté en Mai 2016).

[42] SUMO. <http://sourceforge.net/apps/mediawiki/sumo> (consulté en Mai 2016).

Résumé

Dans notre monde contemporain, les réseaux véhiculaires jouent un rôle significatif. C'est un domaine très intéressant pour toutes les sociétés de recherche et d'industrie. En effet, l'objectif de ces systèmes de transport intelligents est d'améliorer la sécurité et la sûreté des passagers et de fournir de nombreux services et facilités aux usagers routiers. Ces systèmes reposent sur des protocoles qui assurent l'échange d'informations et la communication entre les véhicules puisqu'il est évident que l'amélioration de la communication entre les véhicules revient à la détermination de l'efficacité de ces protocoles. Notre but dans ce mémoire était d'évaluer les performances des protocoles de routage DSR et AODV pour les réseaux véhiculaires en milieu urbain et autoroutier en terme de paquet perdu et de débit, pour pouvoir déterminer le protocole le plus adéquat et garantir la meilleure efficacité pour ce réseau. Tout d'abord, nous avons étudié les différents types de protocoles de routage et leurs facteurs de performance dans ces réseaux afin de déterminer le meilleur protocole pour ce type de réseaux. Par ailleurs, nous avons simulé ces deux protocoles dans un milieu urbain (ville de Bejaia) et autoroutier (autoroute Est-Ouest). Nous avons utilisés le simulateur réseau OMNeT++ et le simulateur de mobilité SUMO, Pour déterminer le meilleur protocole en termes de qualité, efficacité et adéquation d'un tel environnement.

Mots clés : DSR,AODV,VANET,OMNeT++,SUMO,protocole de routage,simulation.

Abstract

In our contemporary world, vehicular networks play a significant role. This is a very interesting area for all research organizations and industry. Indeed, the objective of these intelligent transport systems is to improve the safety and the security of passengers and provide many services and facilities to the road users. These systems are based on protocols that ensure information exchange and communication between vehicles since it is obvious that improved communication between the vehicle returns to the determination of the effectiveness of these protocols. Our aim in this paper was to evaluate the performance of DSR routing protocols AODV and vehicular networks in urban and motorway environment in terms of lost packet and flow, in order to determine the most appropriate protocol and ensure the best effectiveness in network. First, we studied the different types of routing protocols and performance factors in these networks to determine the best protocol for this type of network. Furthermore, we simulated these two protocols in an urban area (city of Bejaia) and motorway (East-West highway). We used simulator network OMNeT ++ and simulator of mobility SUMO to determine the best protocol in terms of quality, effectiveness and adequacy of such an environment.

Keywords : DSR,AODV,VANET,OMNeT++,SUMO,routing protocols,simulation.