

Table des matières

Dédicace	I
Remerciements	III
Résumé	V
Liste des figures	VIII
Liste des tableaux	IX
Liste des abréviations et des acronymes	X
Table des matières	XII
Introduction Générale	1

Chapitre I : vue l'ensemble des réseaux VANETs

I.1- Introduction	3
I.2- Réseaux Véhiculaires Ad hoc	3
I.2.1- Définition	3
I.2.2- L'architecture d'un réseau VANET	5
I.3- Les services offerts par les réseaux VANET	8
I.3.1- Les services liés à la sécurité routière	8
I.3.2- Les services liés au confort	8
I.4- Les modes de communication dans les réseaux VANET	8
I.4.1- Mode de communication Véhicule-à-Véhicule (V2V)	9
I.4.2- Mode de communication de Véhicule à Infrastructure (V2I)	9
I.5- Les caractéristiques des VANETS	10
I.6- Les applications des VANETS	11
I.7- Les Protocoles de routages	13
I.7.1- Classification des protocoles de routage dans les réseaux VANET	13
I.7.1.1- Le routage basé sur la topologie	13
I.7.1.1.1 Les protocoles proactifs	13
I.7.1.1.2 -Les protocoles réactifs	15
I.7.1.1.3 -Les protocoles hybrides	15
I.7.1.2- Les protocoles de routage basés sur la géographique	16
I.7.1.3- protocole à base de clusterisation	18
I.8- La mobilité	18
I.8.1- Définition	18
I.8.2- Le modèle de mobilité	18
I.8.3- Problèmes liés à la mobilité	19
I.9- Conclusion	20

Chapitre II : Etat de l'art des algorithmes de clusterisation

II.1- Introduction	21
II.2- Intérêts et couts de la clusterisation	21
II.3- classification des algorithmes de clusterisation	24
II.3.1- Rayon des clusters	26
II.3.2- Métrique de sélection des cluster-head	26
II.3.3- structure de l'ensemble des cluster-heads	27
II.4- Métriques d'évaluation des performances des algorithmes de clusterisation	28
II.5- Quelques approches de clusterisation	29
II.5.1- les algorithmes de clusterisation à 1 saut	30
II.5.1.1- les algorithmes à métrique arbitraire	30
II.5.1.2- Les algorithmes à métrique liée à la topologie	33

II.5.1.3-Les algorithmes à métrique spécifiques au nœud mobile	34
II.5.2- les algorithmes de clusterisation à K sauts.....	35
II.5.2.1- les algorithmes à métrique arbitraire	35
II.5.2.2- les algorithmes à métrique liée à la topologie	37
II.5.2.3- les algorithmes à métrique spécifique au nœud mobile	38
II.5.2.4- les algorithmes à métriques combinées	38
II.6- VANET, CLOUD et Clusterisation	39
II.6.1- Architecture des Réseaux CLOUDs	40
II.6.2- Clusterisation	41
II.7-Conclusion	41

Chapitre III : présentation des résultats de simulation

III.1-Introduction	42
III.2-La simulation du réseau VANET	42
III.3-Présentation de NS2	43
III.4-Utilisation de NS2	44
III.5-Architecture et Implémentation	45
III.5.1-Composants de la topologie	45
III.5.2-NAM (Network Animator)	46
III.5.3-Fichier trace (résultat de la simulation)	46
III.6-Préparation de l'environnement d'implémentation	47
III.7-Implémentation de la simulation	48
III.8-Résultats de la simulation	50
III.8.1-Comparaison entre les deux standards	50
III.8.1.1-Scénario de la zone1	50
III.8.1.2-Scénario de la zone2	54
III.8.1.3-Scénario de la zone3	57
III.9- Performances des protocoles de routage avec 802.11p	60
III.9.1-Scénario de La zone1	60
III.9.2-Scénario de la zone 2	63
III.9.3-Scénario de la zone 3	65
Conclusion	67
Conclusion Générale	68
Bibliographie	69

Introduction Générale

Nous assistons ces dernières années à une importante évolution dans le domaine des télécommunications sans fil, cette évolution est due essentiellement aux besoins actuels en termes de disponibilité et d'accès aux données à n'importe quel moment et depuis n'importe quel endroit.

De nombreuses applications ont depuis vu le jour afin d'améliorer notre vie quotidienne : dans nos maisons, nos sociétés, nos voitures... en somme partout.

Les réseaux VANETs consiste à renforcer la prévention routière et à munir nos voitures et nos routes de capacités permettant de rendre la route plus sûre (les informations sur le trafic, les accidents, les dangers, les déviations possibles, les informations météorologique, etc.), améliorer le confort des passagers et rendre le temps passé sur les routes plus conviviale (accès à internet, jeux interactifs entre les passagers des véhicules proches, service de chat, aider les personnes à se suivre sur la route...etc.). les réseaux VANETs sont l'exemple type de ce que nous appelons les systèmes de transports intelligents (ITS, Intelligent Transportation System) et dont le but est d'améliorer la sécurité, l'efficacité et la convivialité dans les transports routiers à travers l'utilisation des nouvelles technologies de l'information et de la communication (NTIC).

Avec le développement rapide des technologies de communications sans fil, une nouvelle architecture basée sur des communications dans les réseaux VANETs suscite ces dernières années un réel intérêt auprès des constructeurs automobiles et des opérateurs télécoms. Ce type d'architecture est formé par les véhicules eux même sans l'appui d'une infrastructure fixe pour le reliage des données et des messages.

Afin d'étudier les VANETs, le déploiement sur terrain n'est malheureusement pas envisageable à ce jour, d'où le recours à la simulation. Plusieurs simulateurs ont été mis à la disposition des chercheurs dans ce but (NS2, GloMoSim...). Lors d'une simulation, la mobilité est un paramètre à ne pas négliger, car les unités dans un VANET peuvent se déplacer à grande vitesse suivant un schéma de mobilité particulier.

Notre but de départ est d'étudier les performances des algorithmes de clusterisation pour les réseaux VANETs à fin de former des clouds. Pour cela nous avons prévu l'utilisation du simulateur OMNET++, nous avons rencontré beaucoup de difficultés et cela est dû principalement au manque de documentation et il nous a fallu plus de temps. Pour ces raisons nous avons fait une étude théorique finie par la proposition de l'approche de clusterisation pour les réseaux VANETs pour former le Cloud, en suite nous avons étudié les performances des protocoles de routage dans les normes 802.11 et 802.11p à l'aide du simulateur NS2. Pour atteindre nos objectifs, notre mémoire sera organisé en trois chapitres et de la façon suivante :

Dans le premier chapitre, nous allons donner un état de l'art des réseaux Ad Hoc véhiculaires (VANETs), leurs caractéristiques et les différents types de services offerts par ces réseaux.

Dans le deuxième chapitre, nous allons étudier la technique de clusterisation qui est largement utilisée dans les réseaux VANETs et nous allons présenter une classification des algorithmes qui met en évidence les principales caractéristiques des approches proposées.

Le troisième chapitre sera consacré à la simulation, interprétation et discussions des résultats, où nous allons tester les protocoles AODV, DSDV et DSR afin de voir l'impact du choix du protocole de routage sur les performances du réseau VANET.

Ce document sera clôturé par une conclusion générale et une référence bibliographique.

I.1- Introduction

Les réseaux véhiculaires Ad hoc (VANETs) est une forme de mise en application des réseaux ad hoc mobiles (MANETs). Ils constituent le noyau d'un système de transport intelligent (STI) ayant comme objectif principal l'amélioration de la sécurité routière en tirant profit de l'émergence de la technologie de communication et la baisse cout des dispositifs sans-fil. En effet, grâce à des capteurs installés au sein de véhicules, ou bien situés au bord des routes et des centres de contrôle, les communications véhiculaires permettront aux conducteurs d'être avertis suffisamment tôt de dangers éventuels.

De plus, ces réseaux ne se contenteront plus d'améliorer la sécurité routière seulement, mais ils permettront aussi d'offrir de nouveaux services aux usagers des routes rendant la route plus agréable.

Dans ce chapitre, nous présentons les réseaux VANETs, les différents types de services offerts par ces réseaux et les modes de communication existants, et nous décrivons les différentes caractéristiques, contraintes et les défis qui affronteront les concepteurs lors de la conception des protocoles dédiés à ce type de réseau.

I.2- Réseaux Véhiculaires Ad hoc

I.2.1- Définition

Un réseau VANET est une particularité des réseaux MANETs où les nœuds mobiles sont des véhicules (intelligents) équipés de calculateurs, de cartes réseau et de capteurs. Comme tout autre réseau Ad hoc, les véhicules peuvent communiquer entre eux (pour échanger les informations sur le trafic par exemple) ou avec des stations de base placées tout au long des routes (pour demander des informations ou accéder à internet...). Les réseaux véhiculaires regroupent deux grandes classes d'applications, à savoir les applications qui permettent de bâtir un système de transport intelligent ITS (Intelligent transport System) et celles liées au confort ou avertissement du conducteur et des éventuels passagers [1] [2].



Figure I.1 : Exemple d'un réseau VANET [3].

« La figure I.2 » représente la hiérarchie des réseaux sans fil, où elle schématise l'inclusion des réseaux véhiculaires Ad Hoc VANETs dans les réseaux mobile Ad Hoc MANETs, les MANETs dans les réseaux Mobiles ainsi que les réseaux mobiles dans les réseaux sans fil.

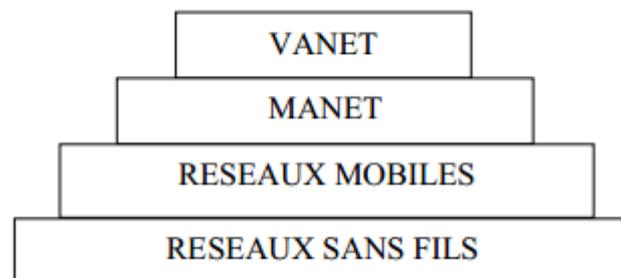


Figure I.2 : Hiérarchie des réseaux sans fil.

Pour la mise en place d'un tel réseau, certains équipements électroniques doivent être installés au sein de véhicules « figure I.3 », tel : les dispositifs de perception de l'environnement (radars, caméras), un système de localisation GPS, et bien sûr une plateforme de traitement. Plusieurs technologies peuvent être mises en œuvre pour l'établissement des communications véhiculaires, tel : les réseaux sans-fil de type 802.11, WIMAX, Bluetooth [4].

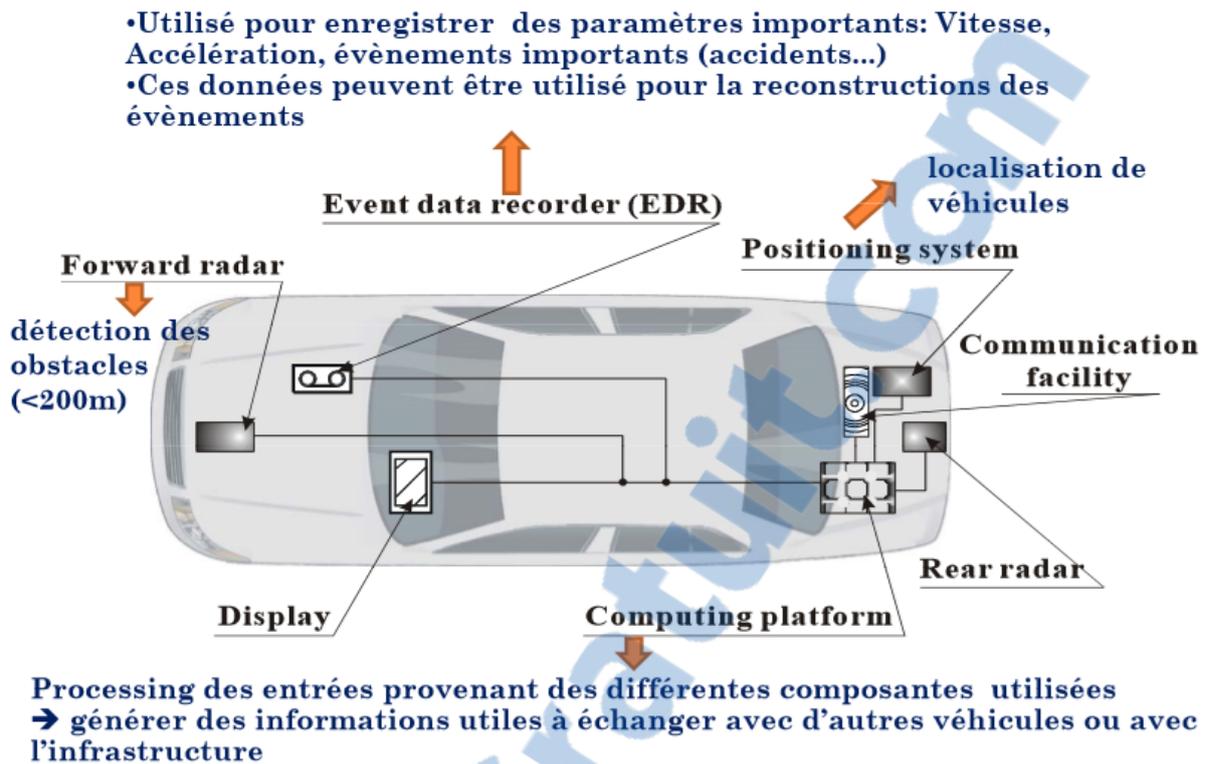


Figure I.3 : Les éléments constituant le véhicule intelligent [4].

I.2.2- L'architecture d'un réseau VANET

La communication entre véhicules, ou entre un véhicule et un RSU est réalisée à travers un milieu sans fil appelé WAVE. Cette méthode de communication offre une vaste gamme d'informations aux conducteurs et aux voyageurs et permet aux applications de sécurité d'améliorer la sécurité routière et de fournir une conduite confortable.

Les principaux composants du système sont UA (Unit Application), OBU (On Board Unit) et RSU (Unit Side Road). Typiquement, la RSU héberge une application qui fournit des services, et l'OBU est un dispositif de pairs qui utilise les services fournis. L'application peut résider dans la RSU ou dans l'OBU ; le dispositif qui héberge l'application est appelé le fournisseur, et le dispositif qui utilise l'application est décrit comme étant l'utilisateur. Chaque véhicule est équipé d'un OBU et un ensemble de capteurs pour collecter et traiter l'information puis de l'envoyer en tant que message à d'autres véhicules ou UAR à travers le support sans fil, elle comporte également un UA unique ou multiple qui utilise les applications fournies par le fournisseur en utilisant les capacités de connexion de l'OBU.

Le RSU peut également se connecter à Internet ou à un autre serveur qui permet de l'UA à partir de plusieurs véhicules de se connecter à l'Internet [5].

Unité embarquée (OBU) : Un OBU est un dispositif à ondes généralement monté à bord d'un véhicule utilisé pour échanger des informations avec les unités d'actions restreintes ou avec d'autres OBU. Il se compose d'un processeur de commande de ressources (RCP), et les ressources comprennent une mémoire d'écriture et de lecture utilisée pour stocker et récupérer des informations, d'une interface utilisateur, une interface spécialisée pour se connecter à d'autres OBU qui est un dispositif de réseau de communication sans fil de courte portée basée sur la norme IEEE 802.11p technologie radio. Il peut en outre comprendre un autre périphérique réseau pour les applications non sécurisées basées sur d'autres technologies radio telles que IEEE 802.11a / b / g / n. L'OBU se connecte à RSU ou à un autre OBU par une liaison sans fil sur la base du canal de fréquence radio IEEE 802.11p, et il est responsable de la communication avec les autres ou avec OBU ou UAS, il fournit également un service de communication aux données de l'UA et transmet le compte d'autres OBU sur le réseau. Les principales fonctions de l'OBU sont l'accès radio sans fil ad hoc, routage géographique, le contrôle de la congestion du réseau, transfert de messages fiable et la sécurité des données [5].

Unité d'application (UA) : l'UA est l'appareil équipé dans le véhicule qui utilise les applications fournies par le fournisseur en utilisant les capacités de communication de l'OBU, L'UA peut être un dispositif dédié pour les applications de sécurité ou un dispositif normal comme un assistant numérique personnel (PDA) pour exécuter l'Internet, l'UA peut être connecté à l'OBU via une connexion filaire ou sans fil et peut résider avec l'OBU dans une seule unité physique, la distinction entre l'UA et l'OBU est logique. L'UA communique avec le réseau uniquement par l'OBU qui assume la responsabilité de toutes les fonctions de mobilité et de routage [5].

Unité routière (RSU) : Le RSU est un dispositif d'onde fixe habituellement le long du bord de la route ou dans des endroits dédiés tels qu'aux intersections ou à proximité de places de stationnement. Le RSU est équipé d'un dispositif de réseau pour une communication à courte portée dédiée basée sur la technologie de radio IEEE 802.11p, et peut également être équipé d'autres dispositifs de réseau de manière à être utilisé aux fins de la communication au sein du réseau des infrastructures [5].

Selon C.C. Communication Consortium, les principales fonctions et les procédures associées à la RSU sont :

- Extension de la portée de communication du réseau ad hoc par redistribution de l'information à d'autres OBU et en envoyant les informations à d'autres unités d'actions restreintes afin de le transmettre à d'autres OBU.
- applications de sécurité de course, comme un avertissement de pont basse, l'alerte de l'accident ou de la zone de travail, en utilisant l'infrastructure de communication de véhicule (I2V) et agissant comme une source d'information.
- Fournir une connectivité Internet aux OBU.

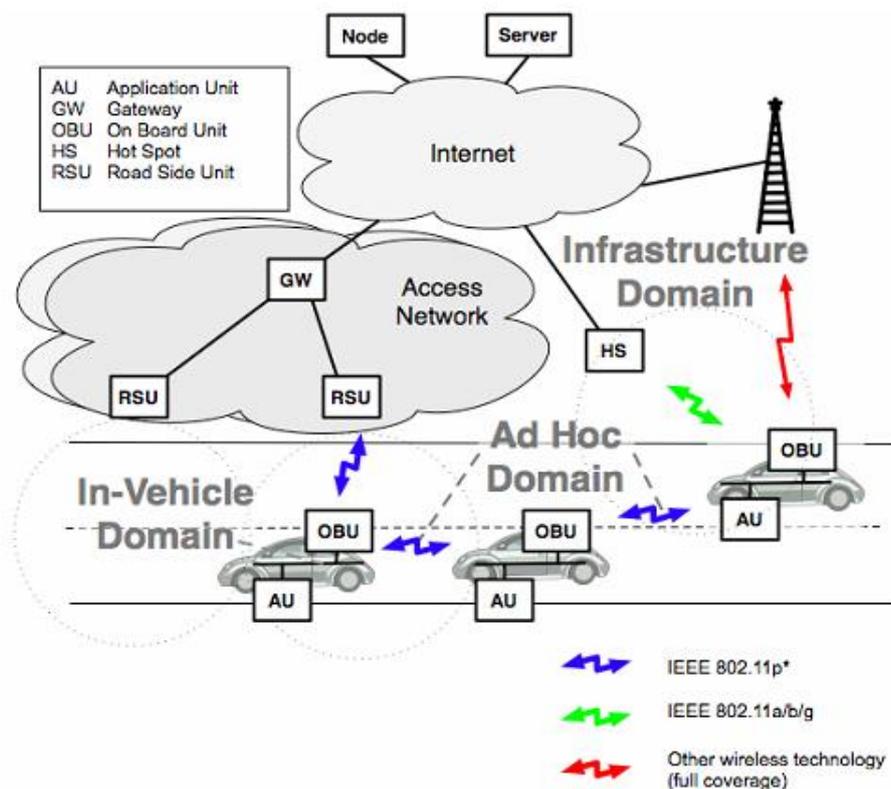


Figure I.4: Architecture de VANET [6].

I.3-Les services offerts par les réseaux VANETs

I.3.1- Les services liés à la sécurité routière

Ces services concernent les applications ayant un impact direct sur la sécurité des personnes et des biens, c'est à dire les applications qui permettent de réduire le nombre des accidents routiers et d'améliorer les conditions de circulation. Les services liés à la sécurité routière se basent sur la détection de l'environnement proche au moyen de capteurs (par exemple : les radars et les caméras) installés au niveau des véhicules ou bien au centre de contrôle, ainsi que la diffusion de messages fournissant des informations sur l'état du réseau routier (trafic, travaux, météo), ou rappelant au conducteur les limitations de vitesse, les distances de sécurité ou qu'il s'approche d'une intersection, avant même de la voir. Certains services peuvent automatiquement effectuer les actions appropriées pour éviter les accidents alors que d'autres services se contenteront d'assister les conducteurs [7].

I.3.2- Les services liés au confort

Les réseaux VANETs ne se contenteront pas seulement à offrir des services liés à la sécurité des véhicules et leurs occupants, mais permettront aussi d'assurer le confort de ces derniers durant leurs voyages, ces services comprennent entre autres : la messagerie instantanée, les jeux en réseau, l'accès à Internet, les paiements automatiques et la diffusion d'informations utiles sur la disponibilité de l'espace de stationnement dans les parkings en indiquant aux conducteurs les espaces libres. Le champ d'application de ces services, à ce stade est très large et offre de perspectives intéressantes aux opérateurs de télécommunication en leurs permettant de réaliser des bénéfices supplémentaires [7].

I.4-Les modes de communication dans les réseaux VANETs

Dans les réseaux de véhicules, on peut distinguer deux modes de communication, les communications Véhicule-à-Véhicule (V2V) et les communications Véhicule-à-Infrastructure (V2I) [8] [9]. Les véhicules peuvent utiliser un de ces deux modes ou bien les combiner s'ils ne peuvent pas communiquer directement avec les infrastructures. Dans cette section, nous présentons le principe et l'utilité de chaque mode.

I.4.1- Mode de communication Véhicule-à-Véhicule (V2V)

Ce mode de communication fonctionne suivant une architecture décentralisée, et représente un cas particulier des réseaux ad hoc mobiles, Il est basé sur la simple communication inter-véhicules ne nécessitant pas une infrastructure. En effet, un véhicule peut communiquer directement avec un autre véhicule s'il se situe dans sa zone radio, ou bien par le biais d'un protocole multi-sauts qui se charge de transmettre les messages de bout en bout en utilisant les nœuds voisins qui les séparent comme des relais. Dans ce mode, les supports de communication utilisés sont caractérisés par une petite latence et un grand débit de transmission.

Les communications V2V sont très efficaces pour le transfert des informations concernant les services liés à la sécurité routière, mais elles ne garantissent pas une connectivité permanente entre les véhicules [10] [11].

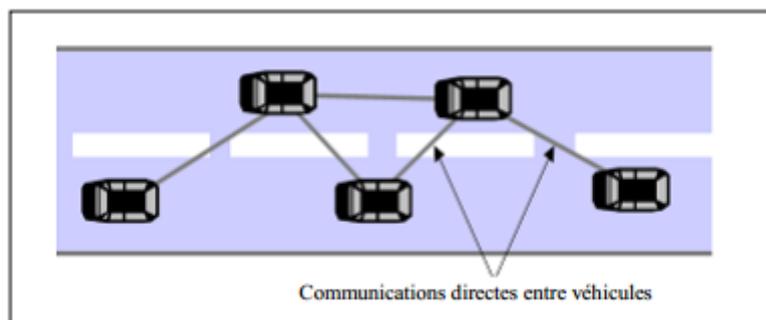


Figure I.5: Communication véhicule à véhicule [12].

I.4.2- Mode de communication de Véhicule à Infrastructure (V2I)

Ce mode de communication permet une meilleure utilisation des ressources partagées et démultiplie les services fournis (par exemple : accès à Internet, échange de données de voiture à domicile, communications de voiture à garage de réparation pour le diagnostic distant, ...etc.) grâce à des points d'accès RSU déployés aux bords des routes, ce mode est inadéquat pour les applications liées à la sécurité routière car les réseaux à infrastructure ne sont pas performants qu'aux délais d'acheminement [13].

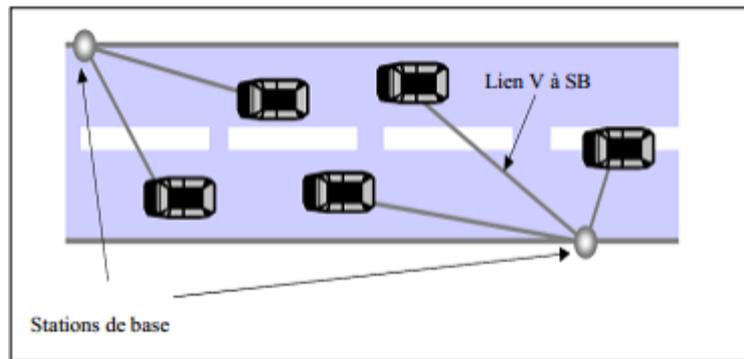


Figure I.6 : Communication véhicule à station de base [12].

I.5-Les caractéristiques des VANETS

Les réseaux véhiculaires ont des caractéristiques spécifiques qui les distinguent de réseaux ad hoc mobiles, Ces caractéristiques doivent être prises en compte lors de conception des protocoles pour les VANETS.

Dans cette partie, nous présentons quelques propriétés et contraintes concernant ce type de réseau.

- **La capacité d'énergie et stockage :** Contrairement au contexte des réseaux MANETs où la contrainte d'énergie représente un défi pour les chercheurs, les éléments du réseau VANET disposent suffisamment d'énergie qui peut alimenter les différents équipements électroniques d'une voiture intelligente. Donc, les nœuds sont censés d'avoir une grande capacité de traitement et de stockage de données [13].
- **La topologie et la connectivité :** Comme les réseaux ad hoc mobiles, les réseaux VANETs sont caractérisés par une connectivité sporadique, car un véhicule (nœud) peut rejoindre ou quitter un groupe de véhicules en un temps très court, ce qui nous mène ainsi à avoir une topologie très dynamique constituée de plusieurs îlots séparés [13].
- **Le modèle de mobilité:** plusieurs facteurs peuvent affecter la mobilité dans ces réseaux comme les infrastructures routières, par exemple : route, autoroute, panneaux de signalisation. En outre, la mobilité dans les VANETs est liée directement au comportement des conducteurs et leurs réactions face à des obstacles ou des situations différentes et complexes rencontrées, par exemple: les heures d'embouteillage, les accidents,... etc. [14].

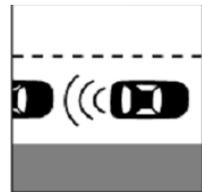
- **La sécurité et l'anonymat:** l'importance des informations échangées via les communications véhiculaires rend l'opération de sécurisation de ces réseaux cruciale qui constitue un pré requis au déploiement des VANETs [15].

I.6-Les applications des VANETs

Selon le DSRC, il ya plus d'une centaine de demandes recommandées dans VANETs. Ces applications sont de deux catégories, la sécurité et non liées à la sécurité. En outre, ils peuvent être classés en OBU à OBU ou applications OBU à RSU. Voici une liste de certaines de ces applications [16].

- **Avertissement coopératif de collision :**

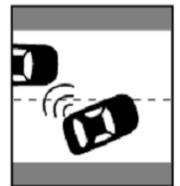
Avertissement coopératif de collision est une application de sécurité OBU à OBU, c'est – à-dire, en cas de tout changement brusque de vitesse ou de direction de conduite, le véhicule est considéré comme anormale et diffuse un message d'avertissement pour avertir tous les véhicules suivants du danger probable.



Cette application nécessite un algorithme de diffusion efficace avec une très faible latence.

- **Avertissement de changement de ruelle :**

Avertissement de changement de ruelle est une application de sécurité OBU à OBU, c'est-à-dire un conducteur de véhicule peut avertir d'autres véhicules de son intention de modifier la voie de déplacement et pour réserver un couloir vide dans la voie approchée.



Cette application dépend de radiodiffusion.

- **Avertissement de collision d'intersection:**

Avertissement de collision d'intersection est une application de sécurité OBU à RSU. Aux intersections, un nœud centralisé avertit les véhicules qui s'approchent des accidents possibles et les aides à déterminer la vitesse d'approche appropriée. Cette application n'utilise que des messages de diffusion.



- **L'approche de véhicules de secours :**

L'approche de véhicules de secours est une application de sécurité publique OBU à OBU. Les véhicules d'urgence à grande vitesse (ambulance ou une voiture de Police) peuvent Avertir les autres véhicules pour dégager leur ruelle. Encore une fois, cette application dépend sur la radiodiffusion.



- **Avertissement de renversement :**

Avertissement de renversement est une application de sécurité OBU à RSU. Un RSU localisé à courbes critiques peut diffuser des informations sur l'angle de la courbe et l'état de la route, de sorte qu'en s'approchant de véhicules, peut déterminer la vitesse maximale possible avant le retournement.

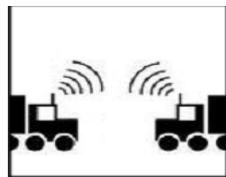


- **Avertissement de domaine d'activités :**

Avertissement de domaine d'activités est une application de sécurité OBU à RSU. Un RSU est monté dans les zones de travail pour avertir les véhicules entrants du danger probable et les mettre en garde pour diminuer la vitesse et de changer la voie de circulation.

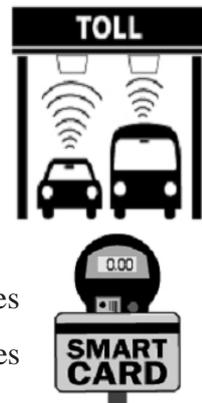
- **Communications inter-véhicule :**

La communication inter-véhicules est une application OBU à OBU non sécurisée qui permet aux voyageurs de communiquer entre eux en utilisant le transfert de fichier instantané, la vidéoconférence ou même le chat vidéo.



- **Collection électronique de péage :**

Télépéage est une application OBU à RSU non sécurisée qui prend en charge la collecte de paiement aux gares de péage, en utilisant des systèmes automatisés pour accroître l'efficacité opérationnelle. Les systèmes sont généralement constitués d'OBU qui est à la charge des cartes à puce prépayées. Ces OBU sont identifiés par UAR situés dans des voies réservées aux postes de péage.



- **Paiement de parking:**

Le paiement de parking est une application OBU à RSU non sécurisée qui fournit des avantages pour les exploitants de parcs de stationnement, simplifier le paiement pour les clients et de réduire la congestion aux entrées et sorties de parcs de stationnement.

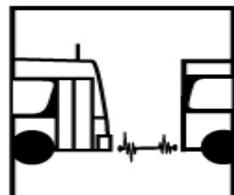
- **Gestion du trafic :**

Dans le véhicule de navigation est une application non sécurisée qui est conçu pour réduire le temps de conduite et la consommation de carburant en échangeant des informations en temps réel sur les conditions de circulation dans la voie de conduite.



- **Couplage / découplage :**

Système de couplage / découplage est une application OBU à OBU non sécurisée qui vise à relier plusieurs bus ou camions dans un train, afin de minimiser le temps de distance de progrès et de voyage et de diminuer les collisions arrières.



I.7-Les Protocoles de routages

Le routage joue un rôle très important dans les VANETs puisque tous les services supportés, unicast ou multicast, se basent sur des communications multi-saut pour l'acheminement des données, Les transferts de fichiers et les jeux. Les communications multicast sont utilisées dans les applications de sécurité et de gestion de trafic telles que l'avertissement de collision et le platooning. Pour réaliser les échanges, les protocoles de routage utilisent des informations locales sur le voisinage immédiat ou globales, concernant tout le réseau, afin de déterminer les nœuds relais qui participent à l'acheminement des données. Les communications unicast sont généralement utilisées dans les applications de confort [17].

I.7.1-Classification des protocoles de routage dans les réseaux VANET

Les réseaux véhiculaires ont comme caractéristique principale une forte mobilité qui entraîne une topologie très dynamique. Cette caractéristique fait que les protocoles de routage traditionnels des MANETS sont pour la plupart inadaptée aux VANETs. En effet, dans les VANETs, la vitesse peut être beaucoup plus élevée que les MANETS dans certains environnements de communication comme les autoroutes. Différentes solutions ont été proposées pour le routage dans les réseaux VANETs, nous distinguons deux classes : les protocoles basés sur la Unicast (topologie) qui sont divisés en protocoles proactifs, réactifs et hybrides pour créer des itinéraires et les protocoles basés sur la localisation (géographique) qui utilisent la position physique des nœuds mobiles pour configurer le routage [17].

I.7.1.1-Le routage basé sur la topologie

I.7.1.1.1- les protocoles proactifs

Le principe de base des protocoles proactifs est de calculer les routes à l'avance, en continu ainsi de maintenir à jour les tables de routage, de tel sorte que lorsqu'un nœud désire envoyer un paquet à un autre nœud, une route soit immédiatement connue. Les nœuds dans les réseaux mobiles Ad Hoc peuvent apparaître et disparaître de manière aléatoire d'où la

nécessité de mise en place d'un système d'échange continu des paquets de contrôle. Les tables de routage sont donc maintenues grâce à ces paquets. Cette manière de procéder permet aux nœuds de construire de façon distribuée la topologie du réseau, lorsqu'un nœud reçoit un paquet de contrôle, il met à jour ses tables de routage. Ainsi, de nouvelles routes seront construites sur la base des informations topologiques transportées par les trames de contrôle. Ce processus est déclenché à chaque changement de topologie pour reconstruire à nouveau les routes. Un des avantages de ces protocoles est la disponibilité immédiate de la route lors du besoin. Cependant, la bande passante diminue de part le trafic généré par l'échange de paquets de contrôle [18] [19].

Le protocole FSR (Fisheye State Routing)

Il est similaire au Protocole de liaison de l'état de routage (LSR). Chaque nœud maintient une table de topologie basée sur les dernières informations reçues des nœuds de quartier. Il utilise période d'échange différent pour différentes entrées dans la table de routage afin de réduire la taille des messages de commande dans les grands réseaux. L'inconvénient du FSR, est que la taille de la table de routage augmente avec l'augmentation de la taille du réseau, la découverte de route peut échouer si le nœud de destination se trouve hors de portée du nœud de source. En raison de la grande mobilité dans VANET, la route vers la destination à distance deviennent moins précise [20].

Le protocol OLSR (Optimized Link State Routing Protocol)

C'est un protocole de routage destiné aux réseaux maillés, sans fil ou mobiles. Le protocole est une optimisation de l'algorithme d'état de liaison pure. Le concept clé utilisé dans le protocole est l'utilisation des relais multipoints (MPR). L'ensemble MPR est choisi de sorte qu'il couvre tous les nœuds qui sont à deux sauts de suite, Il fonctionne comme un protocole proactif, les informations de topologie avec d'autres nœuds du réseau sont échangées régulièrement [21].

I.7.1.1.2 -Les protocoles réactifs

Le principe des protocoles réactifs également appelés protocoles de routage à la demande est de lancer le processus de recherche de routes uniquement en cas de besoin, cela permet d'économiser de la bande passante et de l'énergie. Lorsqu'un paquet doit être envoyé d'un nœud source vers un nœud cible, le protocole de routage va rechercher 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. En général, cette recherche se fait par inondation (un paquet de recherche de route est transmis de proche en proche dans tout ou partie du réseau). Dans ce cas, la bande passante est plus large. Cependant, du fait que l'on ne dispose pas immédiatement de la route vers la destination, le délai d'établissement de la route est plus important en comparaison avec les protocoles proactifs [18] [19].

✚ Le protocole AODV (Ad hoc On Demand Distance Vector)

C'est un protocole de routage destiné aux réseaux ad hoc. Il est à la fois capable de routage unicast et multicast. Il est libre de boucle, auto démarré et s'accommode d'un grand nombre de nœuds mobiles.

Lorsqu'un nœud source demande une route, il crée les routes à la volée et les maintient tant que la source en a besoin. Pour les groupes multicast, AODV construit une arborescence.

Ce protocole de routage est peu gourmand en énergie et ne nécessite pas de grande puissance de calcul, il est donc facile à installer sur de petits équipements mobiles [22].

✚ Le protocole DSR (Dynamic Source Routing)

C'est un protocole de routage pour les réseaux maillés sans fils (wireless mesh networks). Il est similaire à AODV dans le sens qu'il forme une route à la demande, quand un ordinateur veut transmettre. Cependant, il utilise le routage à la source au lieu de se baser sur la table de routage de chaque routeur intermédiaire [22].

I.7.1.1.3 -Les protocoles hybrides

Les protocoles hybrides combinent les deux approches précédentes. Ils utilisent une technique proactive dans un petit périmètre autour de la source où le nombre de sauts est assez petit (par exemple trois à quatre sauts) et réactive pour les nœuds plus éloignés.

L'avantage des protocoles hybrides est le fait qu'ils s'adaptent mieux aux réseaux de grandes tailles. Cependant, ce type de protocole cumule les inconvénients des protocoles proactifs et ceux des protocoles réactifs, tels que l'échange de paquets de contrôle réguliers et inondation de l'ensemble de réseau pour chercher une route vers un nœud éloigné [18][19].

Le protocole ZRP (Zone Routing Protocol)

C'est un modèle hybride entre un schéma proactif et un schéma réactif. Le principal problème dans l'élaboration d'un protocole de routage pour réseau ad hoc réside dans le fait que pour déterminer le parcours d'un paquet de données, le nœud source doit au moins connaître les informations permettant d'atteindre ses proches voisins. D'un autre côté, la topologie d'un tel réseau change fréquemment. De plus, comme le nombre de nœuds peut être élevé, le nombre de destinations potentielles peut également l'être, ce qui requiert des échanges de données important et fréquents. Donc la quantité de données de mise à jour du trafic peut être conséquente. Cela est en contradiction avec le fait que toutes les mises à jour dans un réseau interconnecté ad hoc circulent dans l'air et donc sont coûteuses en ressources [23].

Le protocole HARP

Il divise l'ensemble du réseau en zones non-chevauchées. Il vise à établir un itinéraire stable depuis une source vers une destination à améliorer le retard. Il s'applique dans la découverte de route entre les zones pour limiter les inondations dans le réseau et de choisir le meilleur itinéraire en fonction des critères de stabilité.

HARP est effectué à deux niveaux: intra-zone et inter-zone en fonction de la position de destination, Il utilise des protocoles proactifs et réactifs en intra-zone et inter-zone routage respectivement. Il n'est pas applicable dans les réseaux ad hoc haute mobilité [24].

I.7.1.2- Les protocoles de routage basés sur la géographique [25]

Les protocoles de routage géographiques sont les plus adaptés pour les réseaux ad hoc de véhicules, puisque le mécanisme de routage se base sur les données géographiques des nœuds.

Le protocole GSR (Geographic Source Routing)

Le GSR est un protocole de routage géographique qui combine entre le routage basé sur la position et le routage basé sur la topologie. Son routage est basé sur la topologie car il calcule et stocke l'image de la topologie générale du réseau actuel est le met à jour périodiquement

en cas de déplacement des nœuds du réseau, à partir de cette image et des informations géographique (routage basé sur la position) il assure l'acheminement le plus court à l'aide aussi des algorithmes de recherches.

Le protocole A-STAR (Anchor-based Street and Traffic Aware Routing)

C'est un Protocol basé sur la position pour un environnement de communication inter-véhiculaire métropolitain, Il utilise particulièrement les informations sur les itinéraires d'autobus de ville pour identifier une route d'ancre (Anchor route) avec une connectivité élevée pour l'acheminement des paquets. A-STAR est similaire au protocole GSR en adoptant une approche de routage basée sur l'ancre (Anchorbased) qui tient compte des caractéristiques des rues. Cependant, contrairement à GSR il calcule les "Anchor Paths" en fonction du trafic (trafics de bus, véhicules, etc.). Un point est assigné à chaque rue en fonction de sa capacité. Nous trouvons qu'une des perspectives à ce travail consiste à donner un poids dynamique et qui changerait en fonction de ces informations collectées et du trafic à un moment donné, afin de fournir une meilleure qualité de calcul d'ancres.

Le protocole Gy-TAR (Greedy Traffic-Aware Routing Protocol)

Le protocole de routage Gy-TAR est un protocole géographique basé sur la position et adapté aux réseaux véhiculaires, capable de trouver des chemins robustes dans un environnement urbain. Chaque véhicule connaît sa propre position grâce à l'unité du système de positionnement global (GPS) et l'émetteur connaît la position de destinataire grâce à des services de localisation tel que GLS (Grid Localisation Service) et peut déterminer la position des nœuds voisins à un saut à travers des cartes numériques.

Le protocole DVCAST (Distributed Vehical Broadsact Protocol)

C'est un protocole basé sur la diffusion distribué, il divise les véhicules en trois catégories selon leur connectivité locale : connectés, peu connectés/zone de voisinage ou bien totalement déconnectés. Chaque véhicule utilise un drapeau (flag) pour vérifier si le paquet est redondant ou non en utilisant les informations locales de la topologie en inondant le réseau par des messages périodiques « HELLO ».

I.7.1.3- protocole à base de clusterisation

Dans les protocoles de routage véhicules à base de cluster, près les uns des autres forment un cluster. Chaque groupe a un cluster-head (grappe tête), qui est responsable des fonctions de gestion pour intra et inter-clusters. Les nœuds intra-cluster communiquent entre eux en utilisant les liens directs, alors inter-cluster communication est réalisée via cluster head [26].

Le protocole HCB (Hard Core Bitch)

C'est un protocole de routage basé sur cluster hiérarchique conçu pour les réseaux ad hoc hautement mobile. Son architecture de communication est basée sur deux couches. Dans la couche 1 pour la plupart des nœuds ont une interface radio unique et ils communiquent les uns avec les autres par la voie multi-hop. Parmi ces nœuds certains ont également une autre interface avec la gamme de communication à longue radio appelée super nœuds, qui existent à la fois sur la couche 1 et 2. Super nœuds sont capables de communiquer les uns avec les autres via la station de base en couche 2. Au cours de la formation de grappes, chaque nœud sera privilégié la plus proche tête de cluster et des super nœuds deviendront pôle têtes de couche 1. Dans HCB, le routage intra-cluster est effectué indépendamment dans chaque grappe.

I.8- La mobilité

I.8.1- Définition

Le terme mobilité est la capacité ou la facilité d'un objet ou d'une personne à se déplacer par rapport à un lieu, à une position ou à un ensemble d'objets de même nature. Dans le domaine de réseaux, la mobilité se traduit par la possibilité que certaines entités peuvent passer d'une cellule à une autre sans perdre la liaison [27].

I.8.2- Le modèle de mobilité

La mobilité est très importante dans la simulation de réseaux de véhicules. En effet, c'est elle qui va déterminer si deux pairs sont proches et peuvent communiquer.

Afin de définir un modèle de mobilité adéquat pour les réseaux de véhicules [27], on distingue les environnements suivants :

- **Autoroute** : environnement ouvert caractérisé par une grande vitesse de déplacement avec des limites (vitesse min et vitesse max), avec des dépassements de véhicules et une densité de nœuds fonction de l'heure de la journée.
- **Ville** : vitesse modérée avec une probabilité d'intersection plus grande. Il existe des endroits d'arrêt aux feux, une grande densité de voitures et l'existence de certaines routes plus fréquentées que d'autres (routes principales, endroit commercial ou touristique par exemple).
- **Rase campagne** : caractérisé par des vitesses moins importantes avec une densité de voitures plus faible.

I.8.3- Problèmes liés à la mobilité [12]

- **Nommage et adressage des nœuds mobiles** : dans les réseaux fixes, l'adresse d'un nœud est souvent confondue avec son nom. Par contre, nous ne pouvons pas nommer un nœud mobile par son adresse car il change de cellules et donc utilise plusieurs points d'accès, d'où le changement fréquent d'adresses.
- **Routage** : la mobilité des nœuds et le changement fréquent de la topologie nécessitent des protocoles de routage qui doivent réagir rapidement à ces changements, afin de permettre le bon acheminement des paquets à leurs destinations finales.
- **Récupération des adresses** : la déconnexion (accidentelle ou volontaire) d'un nœud mobile nécessite l'adaptation des protocoles de récupération des données. Il est aussi important de supprimer les données résiduelles stockées pour éviter la saturation de l'espace mémoire.
- **Diffusion de données** : dans le cas d'une émission multidestinataires (diffusion sélective) d'un message, ce dernier peut être reçu par des stations différentes à des moments différents. Donc, il est possible qu'un message diffusé ne soit pas reçu par une unité mobile destinataire, car son entrée en cellule a eu lieu après la diffusion du message ou sa sortie a eu lieu avant la diffusion du message. Une unité mobile peut recevoir plusieurs copies du même message à travers des stations de base différentes.

- **Dépendance de la localisation** : dans les réseaux mobiles à stations de base, un nœud mobile peut avoir besoin des services des réseaux filaires (Internet par exemple), il est donc préférable de faire appel à la station de base la plus proche de sa localisation.

I.9-Conclusion

Dans ce chapitre, nous avons montré que les communications dans les réseaux VANETs nous permettent d'améliorer d'une part, la sécurité routière grâce aux messages échangés entre les véhicules et de rendre d'autre part les routes plus agréables grâce à la diversité des services offerts. Toutes ces applications nécessitent des concepteurs en tenant compte de l'importance de l'échange d'informations entre les véhicules.

Dans la suite de notre travail, nous allons s'intéresser à la technique de clusterisation qui a été couramment utilisée dans les réseaux VANETs et nous proposons également des algorithmes qui créent et maintiennent une structure stable des clusters pour prolonger la validité des routes.

II.1- Introduction

La clusterisation consiste à diviser le réseau en un ensemble de nœuds géographiquement proches. Elle présente alors une solution intéressante pour simplifier et optimiser les fonctions et les services du réseau. En particulier, elle permet au protocole de routage de fonctionner plus efficacement en réduisant le trafic de contrôle dans le réseau et en simplifiant le processus d'aiguillage des données. Plusieurs algorithmes de clusterisation ont été proposés et évalués. Ces algorithmes ont des caractéristiques différentes et sont conçus pour satisfaire certains objectifs selon le contexte dans lequel la clusterisation est déployée (routage, sécurité, conservation de l'énergie, etc.). Plusieurs études se sont intéressées à établir une classification des approches existantes de clusterisation.

II.2-Intérêts et couts de la clusterisation

La technique de clusterisation consiste à organiser les nœuds du réseau en des groupes virtuels appelés « clusters ». Les nœuds géographiquement voisins, sont regroupés dans un même cluster selon certaines règles. Dans un cluster, on retrouve généralement trois types de nœuds [28] comme le montre «la figure II.1 » :« cluter-head », nœuds « passerelles » et nœuds « ordinaires » dite aussi membre. Dans chaque cluster, un nœud est élu comme chef de groupe appelée cluster-head, qui possède des fonctions supplémentaires (routage, accès au médium, etc.). Une passerelle est un nœud non-cluster-head qui possède des liens inter-clusters et peut donc accéder à des clusters voisins et acheminer les données entre eux, tandis qu'un nœud ordinaire est un nœud non-cluster-head qui ne possède pas des liens avec les autres clusters.

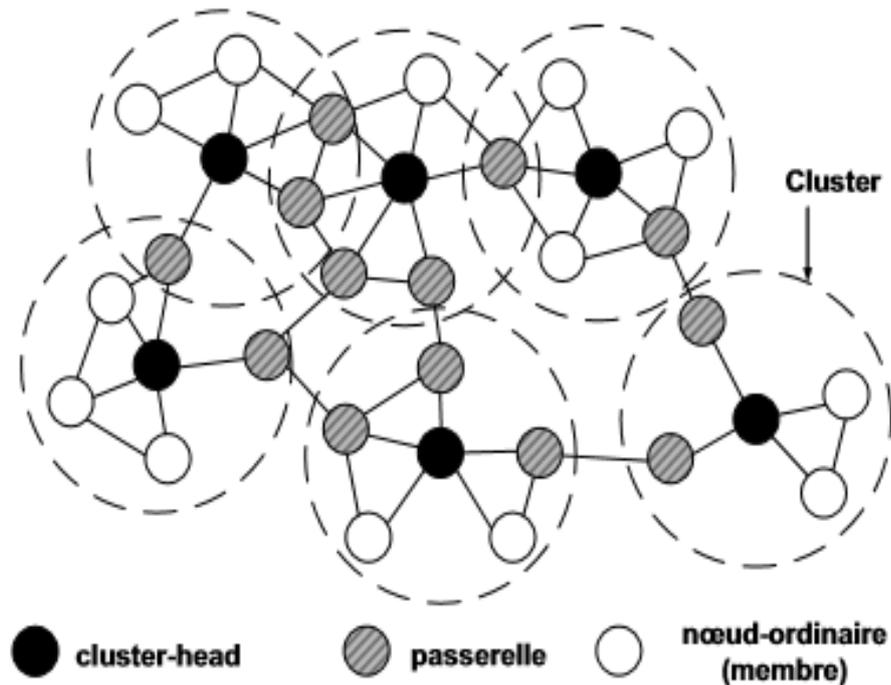


Figure II.1 : Exemple d'une structure d'un cluster.

La clusterisation représente une solution prometteuse pour les réseaux « auto-organisables » à grand nombre de nœuds. Cette technique de structuration du réseau possède au moins trois avantages :

- La structure de clusters permet la réutilisation des ressources de réseau [29] [30], deux clusters peuvent utiliser la même fréquence ou le même jeu de code s'ils ne sont pas voisins, en plus, les nœuds de chaque cluster sont supervisés par leur cluster-head qui peut coordonner l'accès au canal, épargnant ainsi les ressources gaspillées dans la retransmission due aux collisions.
- L'ensemble des cluster-head et des passerelles peut former une dorsale virtuelle pour le routage inter-cluster. Ainsi, la génération et la diffusion des informations de routage peuvent se limiter à cet ensemble de nœuds [31] [32] ce qui évite les diffusions fréquentes des informations de routage pouvant surcharger le réseau et dégrader considérablement ses performances.

- La structure de cluster fait apparaître le réseau plus petit et plus stable aux yeux de chaque nœud du réseau, quand un nœud mobile change son cluster d'attache, seuls les nœuds faisant partie du même cluster du nœud ayant quitté et actualisent leurs informations, donc les changements locaux n'affectent pas le réseau entier mais uniquement une partie. Ainsi, l'information traitée et stockée par chaque nœud sera réduite.

La clusterisation a l'avantage d'être une alternative attrayante pour remédier aux problèmes de passage à l'échelle. Cependant, cette technique possède des inconvénients. En effet, construire et maintenir une structure de clusters demande un coût supplémentaire comparé à un réseau plat. L'analyse du coût de la clusterisation permet d'évaluer son efficacité et son passage à l'échelle. Cette analyse peut être quantitative ou qualitative.

- Le trafic de contrôle périodique requis pour la maintenance de la structure des clusters [33] constitue une charge supplémentaire du réseau. Comme le nombre de messages de contrôle croît avec le nombre des nœuds dans un réseau dense et large, le trafic considérable de contrôle va puiser une partie des ressources limitées en termes de bande passante, de temps de traitement et d'énergie. Les messages de contrôle vont concurrencer conjointement avec le trafic de données pour accéder au canal, surchargeant ainsi le réseau. Dans de telles conditions, les applications de couches supérieures ne peuvent plus fonctionner correctement.
- L'effet cascade de la ré-clusterisation est aperçu dans certaines propositions de clusterisation et peut affecter considérablement les performances des protocoles des couches supérieures. Cet effet consiste à déclencher la reconstruction de la totalité de la structure des clusters sur tout le réseau si un certain événement local survient (le mouvement ou le départ d'un nœud) et résulte à la réélection du cluster-head [33]. Cet effet aura lieu lorsqu'une réélection d'un seul cluster-head affecte la structure de plusieurs clusters et déclenche une chaîne de réélection des cluster-head sur tout le réseau.
- La phase de formation des clusters est une phase de collecte d'information et de prise de décision du rôle à jouer par chaque nœud. La durée de cette phase mesurée en nombre de tours demeure une métrique importante et représente un coût considérable dans un algorithme de clusterisation. Certaines propositions supposent que les nœuds

restent statiques pendant la phase de formation. Ainsi, plus cette phase est longue plus le temps où les nœuds restent non mobiles est long.

Pour la majorité des approches de clusterisation, la procédure de formation peut s'exécuter en parallèle sur tout le réseau ce qui rend la convergence de formation des clusters plus rapides. Mais dans ces approches, les nœuds ne prennent pas le même temps de décision de leur rôle. En plus, pour les mécanismes de clusterisation où la décision d'un nœud se base sur les décisions diffusées dans son voisinage, parfois des nœuds attendent indéfiniment dans des réseaux denses ou à grande envergure. Ainsi, la phase de formation peut ne pas être bornée et peut prendre des valeurs variables en fonction de la topologie du réseau.

II.3-classification des algorithmes de clusterisation

Plusieurs classifications des mécanismes de clusterisation ont été proposées dans la littérature. La classification de l'approche existante peut se baser sur plusieurs critères. La première classification évidente repose sur les critères et les métriques utilisés dans le choix du cluster-head. Dans cet axe, certains mécanismes de clusterisation ont choisi des critères simples comme l'identifiant. Alors que d'autres approches ont adopté des sélections plus raffinées. Ces sélections combinent certains critères dans le but de sélectionner les nœuds les plus appropriés pour jouer le rôle de cluster-head, une autre classification peut prendre en compte le nombre de sauts qui séparent un nœud ordinaire du cluster-head auquel il est rattaché. Dans cette classification, on retrouve deux catégories : les algorithmes à 1 saut et ceux à plusieurs sauts [34].

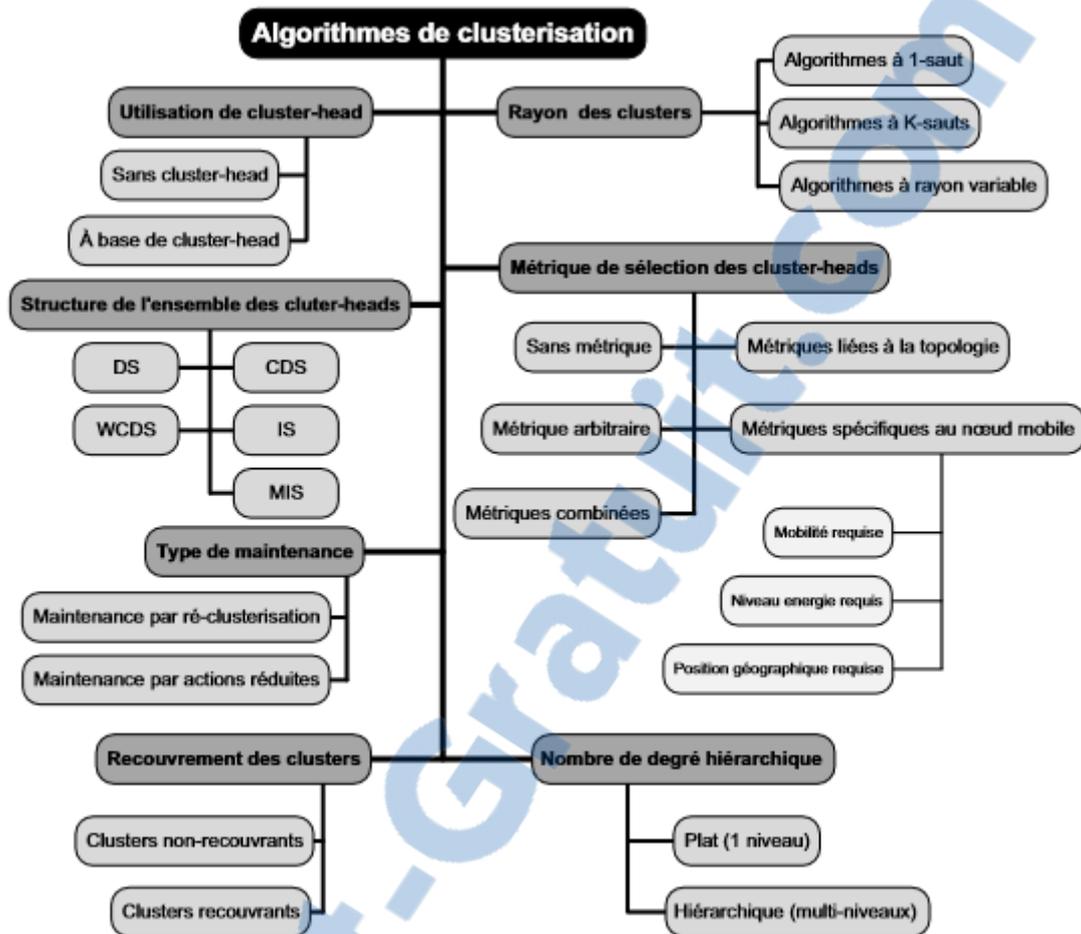


Figure II.2 : Classification des algorithmes de clusterisation [34].

L'étude de classe des algorithmes de clusterisation distingue six catégories d'algorithmes : les algorithmes se basant sur la construction d'un ensemble dominant, les algorithmes à charges équilibrée, les algorithmes à faible coût de maintenance, les algorithmes conscients de la mobilité, les algorithmes d'économie d'énergie et les algorithmes à métriques combinées.

Nous allons présenter et distinguer des critères de classification des approches existantes de clusterisation illustrés dans la figure II.2.

II.3.1-Rayon des clusters

Le rayon d'un cluster exprime la valeur maximale de la distance qui sépare le cluster-head à l'ensemble de ses membres, Cette distance est exprimée en nombre de sauts. Nous distinguons trois classes d'algorithmes : les algorithmes à 1 saut, les algorithmes à K sauts et les algorithmes à rayon variable [34].

II.3.2-Métrique de sélection des cluster-head :

Plusieurs métriques ont été proposées dans la littérature pour élire l'ensemble des cluster-head. Nous proposons de classer les algorithmes de clusterisation dans cinq classes selon le type des métriques employé lors de la sélection des cluster-head [34]. Ces cinq classes sont les suivantes :

- **Sans métrique** : regroupe les algorithmes qui déclarent les cluster-heads sans avoir recours à aucune métrique de sélection.
- **Métrique arbitraire** : représente une valeur choisie arbitrairement et généralement non significative. Dans cette classe, nous trouvons les algorithmes qui utilisent des valeurs aléatoires ou l'identifiant des nœuds.
- **Métriques liées à la topologie** : dans cette classe nous regroupons tous les algorithmes qui utilisent une métrique issue de la topologie du réseau. Parmi les métriques qui appartiennent à cette classe nous citons : le degré de connectivité, le K-degré de connectivité et la K-densité.
- **Métriques spécifiques aux nœuds mobiles** : les algorithmes de cette classe utilisent des métriques très spécifiques aux nœuds mobiles. Dans le cas général, ces algorithmes supposent connaître certaines informations sur chaque nœud tel que sa mobilité, la puissance des signaux reçus, le niveau d'énergie ou encore sa position géographique. Dans cette classe il existe trois sous classes:
 - **Mobilité requise** : regroupe les algorithmes qui supposent que les nœuds ont une connaissance de leur mobilité.
 - **Niveau d'énergie requis** : regroupe tous les algorithmes qui utilisent la valeur d'énergie restante dans les batteries des nœuds comme métrique de sélection des cluster-heads.

- Position géographique requise : cette classe d'algorithmes suppose que chaque nœud a la capacité de déterminer sa position géographique par exemple à travers un système GPS.
- **Métriques combinées** : plusieurs algorithmes combinent plusieurs métriques de différents types pour élire l'ensemble de cluster-heads.

II.3.3-structure de l'ensemble des cluster-heads

Les cluster-heads élus forment des structures particulières [34] parmi les quelles nous citons :

- **Ensemble dominant DS (Dominating Set)** : est un ensemble de nœuds tel que tout nœud du réseau est voisin d'au moins un nœud de cet ensemble.
- **Ensemble dominant connecté CDS (Connected Dominating Set)** : est un ensemble dominant où tous les nœuds dominants sont directement connectés.
- **Ensemble dominant faiblement connecté WDS (Weakly Connected Dominating Set)** : est un ensemble dominant tel que l'ensemble des dominants avec les arêtes dont une des extrémités est un dominant forme un ensemble connexe.
- **Ensemble indépendant IS (Independent Set)** : est un ensemble de nœuds du graphe tel qu'aucune paire de nœuds de l'IS ne se trouvent voisins dans le graphe.
- **Ensemble indépendant maximal MIS (Maximum Independent Set)** : est un ensemble indépendant présentant une cardinalité maximale.

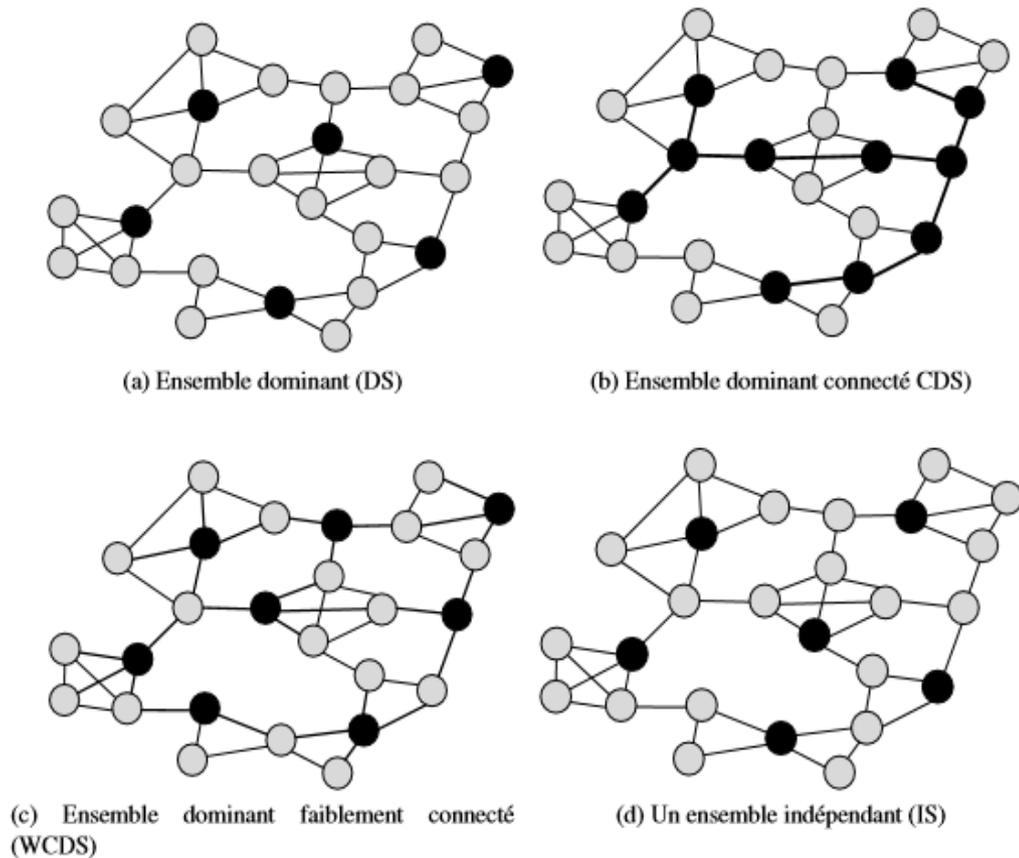


Figure II.3 : Exemple de structures de l'ensemble des cluster-heads [34].

II.4-Métriques d'évaluation des performances des algorithmes de clusterisation

Vu la multitude et la diversité des algorithmes de clusterisation proposés, il est difficile de trouver les bonnes métriques d'évaluation de performance [34]. Les groupes de métriques sont les suivants :

➤ Messages explicites du processus de clusterisation

- Overhead : l'overhead encouru durant le temps d'opération du réseau.
- Complexité en message pour un changement de topologie : le nombre de messages échangés entre les nœuds pour accomplir une réorganisation valide des clusters après un changement dans la topologie du réseau.

➤ Caractéristique des clusters

- Taille moyenne d'un cluster : le nombre moyen de nœuds gérés par un cluster-head.

- Distance totale aux cluster-heads : la somme des distances entre les nœuds et leurs cluster-heads.

➤ **Complexité des algorithmes**

Complexité en temps pour un changement de topologie qui représente le temps passé pour accomplir une réorganisation valide des clusters après un changement dans la topologie du réseau.

➤ **Stabilité de la structure des clusters**

- Durée moyenne de vie d'un cluster-head : la durée moyenne pendant laquelle un nœud joue le rôle de cluster-head.
- Durée moyenne de vie d'un membre : la durée moyenne pendant laquelle un nœud est associé à son cluster-head.
- Nombre de changements de cluster-head : le nombre de fois qu'un cluster-head change de statut.
- Nombre de mise à jour de l'ensemble dominant : l'ensemble des cluster-heads est appelé un ensemble dominant si tous les nœuds dans le réseau sont dans cet ensemble ou sont voisins des nœuds de cet ensemble.
- Nombre de ré-affiliation : le compteur de ré-affiliation est incrémenté quand un nœud est dissocié de son cluster-head et se rattache comme membre à un autre cluster-head à l'intérieur de l'ensemble dominant courant.

➤ **Facteur d'équilibre de charge**

- Durée moyenne de vie d'un cluster-head : la durée moyenne pendant laquelle un nœud joue le rôle de cluster-head.
- Variance de la durée de vie d'un cluster-head : la variance de la durée pendant laquelle un nœud joue le rôle de cluster-head.

II.5-Quelques approches de clusterisation

De nombreuses approches de clusterisation ont été proposées. Une grande partie des ces approches construisent des clusters à 1 saut où chaque nœud est à un saut de son cluster-head. D'autres algorithmes génèrent des clusters à K-sauts où chaque nœud est à au plus K sauts de son cluster-head.

La majorité des algorithmes proposent l'utilisation d'une métrique de base qui permet d'élire un cluster-head et invoque parfois d'autres métriques pour rompre les égalités. Néanmoins, il existe des algorithmes qui n'utilisent aucune métrique particulière pour structurer le réseau en clusters.

Nous avons distingué quatre types de métriques de base utilisées par les algorithmes proposés. La métrique arbitraire est une métrique qui est donnée de façon arbitraire ou aléatoire à un nœud comme l'identifiant d'un nœud. Le deuxième type de métrique concerne les métriques calculées à partir de la topologie du réseau comme la connectivité d'un nœud appelée aussi degré d'un nœud.

D'autres métriques sont plutôt spécifiques au nœud mobile comme sa mobilité relative et son énergie résiduelle. Il y a des algorithmes qui combinent plusieurs métriques pour calculer un poids utilisé dans l'élection du cluster-head [34].

II.5.1- les algorithmes de clusterisation à 1 saut

II.5.1.1- les algorithmes à métrique arbitraire

L'un des algorithmes de clusterisation les plus anciens est LCA (Linked Cluster Algorithm) [35], connu sous le nom de l'algorithme du plus petit identifiant (ID) ou du plus petit ID. chaque nœud se déclare cluster-head ou non, en se basent sur son identifiant et ceux de ses voisins. Un nœud peut être dans l'un de ces quatre statuts : ordinaire, cluster-head, membre, ou passerelle. Initialement, tous les nœuds ont un statut de nœud ordinaire, les nœuds diffusent périodiquement la liste des nœuds qu'ils sont capables de détecter. La formation des clusters suit les règles suivantes :

- Si un nœud U possède le plus petit identifiant dans son voisinage à un saut, il se déclarera comme cluster-head et ses voisins à un saut dont les identifiants sont supérieurs à celui du cluster-head le rejoignent et deviennent des nœuds membres.
- Si non, il attendra que tous ses voisins à un saut déclareront leurs statuts. Ainsi si un parmi eux se déclare cluster-head lors le nœud U déclare à son voisinage à un saut son statut de nœud membre.

- Une fois que tous les nœuds ont, soit le statut de membre ou de cluster-head, alors si un nœud a parmi ses voisins à un saut plus qu'un cluster-head, il se déclarera passerelle.

L'algorithme du plus petit ID considère uniquement les nœuds ayant le plus petit identifiant qui sont choisis arbitrairement sans prendre en considération d'autres caractéristiques sur les nœuds élus comme cluster-head. Comme l'identifiant du nœud ne change pas au cours du temps, les nœuds ayant un petit identifiant sont les plus probables à être choisis comme cluster-head que les ayant un grand identifiant. Ainsi, si ces cluster-heads gardent leur statut de cluster-head pour une longue durée, il en résulte qu'ils consomment rapidement leur énergie et par conséquent ils causent des goulets d'étranglement dans leurs clusters.

La maintenance de la structure des clusters créée par LCA s'avère coûteuse car le mouvement d'un nœud peut engendrer des créations en chaîne et nécessiter la reconstruction totale de la structure. LCC (Least Cluster Change) [36] est alors proposé comme une version améliorée de l'algorithme LCA qui ajoute une étape de maintenance permettant de minimiser le coût de la restructuration du réseau des clusters. L'algorithme LCC distingue deux phases : la phase de formation et la phase de maintenance des clusters.

La formation des clusters est la même que celle dans LCA, c'est-à-dire que les nœuds ayant le plus petit identifiant dans leur voisinage sont élus des clusters-heads. Les clusters ne sont reconstruits que dans les deux cas suivants :

- Si deux cluster-head i et j se trouvent voisins, alors l'un des deux (i ou j) devra abandonner le rôle de cluster-head selon le critère « Plus Grand Degré » et/ou « Plus Petit ID ».
- Si un nœud non-cluster-head se déplace en dehors des clusters formés et ne migre pas vers un cluster existant alors il deviendra cluster-head et formera un nouveau cluster.

Ainsi, si un nœud non-cluster-head u migre d'un cluster $C(i)$ vers un cluster $C(j)$ alors la réélection du cluster-head dans le cluster $C(j)$ n'aura pas lieu même si le nœud u possède les capacités d'être cluster-head dans $C(j)$. De cette façon, LCC améliore considérablement la stabilité des clusters. Mais le coût de la reconstruction du réseau reste toujours un peu onéreux car les réactions en chaînes de reconstruction ne sont que limitées et ne sont pas

complètement supprimées vu qu'un seul nœud peut relancer le processus de clusterisation s'il n'existe pas de cluster-head dans son voisinage.

L'algorithme Adaptive clustering (AC) utilise le statut de cluster-head uniquement pour la formation des clusters. Une fois les clusters formés, la notion de cluster-head disparaît et les nœuds du cluster jouent le même rôle. La motivation est que les cluster-heads peuvent devenir des goulots d'étranglement et dépensent leurs ressources rapidement que les autres nœuds. Cet algorithme utilise la même métrique que l'algorithme LCA (le plus petit identifiant) pour le choix des cluster-heads.

Pour construire les clusters, chaque nœud i maintient initialement son propre ID et les IDs de ses voisins à 1-saut dans un ensemble noté Γ_i . L'algorithme AC est différent de l'algorithme LCA par le non recouvrement des clusters. En plus, l'algorithme se termine après que chaque nœud envoie un seul message de clusterisation. En effet, chaque nœud n'est autorisé à diffuser son état (cluster-head, membre) que s'il possède le plus petit identifiant dans tous les nœuds de l'ensemble Γ_i . Les règles de formation de l'algorithme AC sont les suivantes :

- Un nœud i se déclare cluster-head s'il possède l'identifiant le plus petit dans Γ_i . Si c'est le cas, il doit diffuser sa décision à tous ses voisins dans un message d'état (ID, C-ID).
- A la réception d'un message d'état d'un nœud i , les voisins de i suppriment i de leur ensemble Γ .
- Si un nœud i annonce qu'il est cluster-head (ID=C-ID), ses voisins s'attachent à lui si leur statut est non spécifié ou si le cluster-head auquel ils sont membres a un identifiant plus grand que i .
- Le processus continue jusqu'à ce que l'ensemble Γ de chaque nœud soit vide. Après la phase de formation, les cluster-heads ne jouent plus aucun rôle dans la phase de maintenance. Chaque nœud doit connaître son voisinage à 2-sauts. Par suite, il sait si les membres des clusters sont à 2-saut de lui, si un nœud i trouve la distance entre lui et un nœud j du même cluster devient à 3-saut, il invoque la procédure de maintenance dont les règles sont :
 - Si i est un nœud à 1 saut du nœud ayant la plus haute connectivité, il reste dans le cluster et le nœud j quitte le cluster et rejoint un cluster voisin.
 - Si non, i rejoint un cluster voisin.
 - S'il n'existe pas un cluster convenable, i crée son propre cluster.

Bien que n'utilisant pas la notion de cluster-head, la maintenance de cet algorithme maintient le nœud de plus grand degré au centre du cluster, ce qui peut revenir au même que l'élire comme cluster-head.

II.5.1.2- Les algorithmes à métrique liée à la topologie

L'algorithme HCC (Highest Connectivity Cluster) choisit le nœud ayant le plus grand nombre de voisins (plus grand degré) comme cluster-head. Initialement, tous les nœuds sont non couverts [37]. Chaque nœud diffuse la liste des nœuds qu'il peut détecter. La phase de formation des clusters applique les règles suivantes :

- Un nœud est élu cluster-head s'il a le plus grand nombre de voisins. Dans le cas d'égalité, le nœud ayant le plus petit identifiant ID deviendra cluster-head.
- Un nœud qui n'a pas encore choisi son cluster-head demeure dans son état non couvert, si non il devient un nœud couvert.
- Un cluster-head qui a élu un autre nœud comme son cluster-head, abandonne son rôle de cluster-head.
- Les nœuds appartenant à plusieurs clusters jouent le rôle de passerelle.

L'idée de HCC est que des nœuds ayant un grand degré sont de bons candidats pour être cluster-head car ils couvrent un grand nombre de nœuds et ainsi de nombre de clusters sera réduit. Dans un environnement mobile, cet algorithme produit des cluster-head qui ne sont pas susceptibles de jouer leur rôle comme cluster-head pour très longtemps puisque leur degré change très fréquemment contrairement à l'algorithme du plus petit ID où les nœuds de faible identifiant ont tendance à garder le statut de cluster-head plus longtemps.

L'algorithme LCC déjà présenté possède deux versions LCC-lowest-ID et LCC-hc. Dans la première version, c'est la métrique identifiant qui utilisée alors que dans la deuxième version, le degré d'un nœud est employé pour élire le cluster-head. C'est ainsi que LCC-hc est un algorithme à métrique liée à la topologie tandis que LCC-lowest-ID est à métrique arbitraire.

Les algorithmes de clusterisation génèrent soit des clusters recouvrant ou des clusters non-recouvrant. Dans des clusters recouvrant, un nœud peut appartenir à plusieurs clusters. L'inconvénient des algorithmes à clusters recouvrant est que la ré-affiliation d'un nœud mobile d'un cluster à un autre provoque la restructuration des clusters concernées et dans certains cas la restructuration de tout le réseau. Pour remédier à ce problème, des algorithmes de clusterisation ont été proposés, permettant la création de clusters non-recouvrant c'est-à-dire un nœud ne peut appartenir qu'à un seul cluster. Ceci permet de limiter l'opération de maintenance au niveau d'un seul cluster lors de ré-affiliation d'un nœud d'un cluster à un autre.

II.5.1.3-Les algorithmes à métrique spécifiques au nœud mobile

MOBIC (Lowest Relative Mobility Clustering Algorithm) est un algorithme distribué qui applique le même algorithme que LCA et HCC mais implique une autre métrique qui est la mobilité relative des nœuds. La mobilité relative d'un nœud représente le rapport des niveaux de puissance des transmissions successives reçus par un nœud de ses voisins. Ainsi, le nœud ayant la plus faible mobilité dans son voisinage gardera un voisinage plus stable au cours du temps et sera donc un bon candidat pour être cluster-head.

MOBIC utilise la même procédure de maintenance que LCC à laquelle il ajoute une règle supplémentaire pour minimiser le coût de l'opération de maintenance. Si deux cluster-heads i et j se trouvent voisins, alors l'un des deux n'abandonnera son statut de cluster-head qu'après une certaine période Δt , sinon ils garderont tous les deux leur statut. Cela permet de ne pas faire appel à l'opération de maintenance une fois que deux cluster-heads se trouvent dans le même voisinage. L'optimisation introduite dans MOBIC dans la procédure de maintenance a contribué de réduire à plus du tiers le nombre de changements des cluster-heads relativement au protocole LCC. Cependant, les limitations de l'algorithme LCC ne sont pas totalement éliminées. De plus, MOBIC exige que les nœuds doivent être capables d'estimer le niveau de signal avec leurs voisins pour calculer la mobilité relative [38].

II.5.2- les algorithmes de clusterisation à K sauts

II.5.2.1- les algorithmes à métrique arbitraire

Chaque nœud doit connaître ses voisins à k sauts, et tout nœud non affilié possédant le plus petit ID parmi ses k-voisins non encore affiliés devient cluster-head. Lorsque tous les nœuds de son k-voisinage ayant un plus petit identifiant que lui ont diffusé leur décision d'être cluster-head ou de s'attacher à un autre cluster-head, un nœud peut prendre sa propre décision de s'attacher au nœud de son k-voisinage du plus petit identifiant s'étant déclaré cluster-head s'il existe, ou de créer son propre cluster. Si non, L'opération de maintenance prend en compte le rayon du cluster. Cependant, nous retrouvons toujours les mêmes inconvénients que présentent les algorithmes générant des 1-clusters, à savoir un petit changement dans la topologie du réseau peut provoquer une restructuration de tout le réseau. En outre, dans les réseaux denses tels que les réseaux de capteurs, la taille des clusters peut être importante, ce qui peut épuiser rapidement les batteries des clusters heads et par la suite causer des goulets d'étranglement dans les clusters.

L'algorithme Max-Min-d-clusters [33] construit des d-clusters non recouvrants, avec d'un paramètre de l'heuristique. Cet algorithme utilise l'identifiant du nœud comme métrique d'élection du cluster-head et se déroule en trois tours. Lors du premier tour, chaque nœud diffuse son identifiant à ses voisins à d sauts et collecte leurs identifiant. Il engage le plus grand des identifiants qu'il le diffuse de nouveau dans la deuxième tour. Chaque nœud garde le plus petit des identifiants reçus dans le deuxième tour (le plus petit parmi les plus grands). Le troisième tour consiste au choix de cluster-head basé sur les identifiants enregistrés lors des deux tours précédents. Si un nœud u a reçu son identifiant dans la deuxième phase, il devient cluster-head. Sinon, si u a reçu un identifiant durant chacun des tours 1 et 2, il choisira le nœud ayant cet identifiant comme cluster-head. Sinon, u élit cluster-head le nœud ayant le plus grand identifiant dans son voisinage à d-sauts.

Cet algorithme produit une structure de clusters robuste, cependant, la durée du déroulement de l'algorithme n'est pas négligeable. En outre, cette technique augmente considérablement l'overhead du réseau puisqu'elle utilise un nombre important de messages pour élire les cluster-head. D'autre part cette technique ne tient pas compte de la mobilité des nœuds et du

changement de la topologie. Alors une notre version de l'algorithme Max-Min d-cluster a été améliorée appelée Cluster-head Load-Balancing. Le but est d'assurer un équilibre de charge entre les nœuds en évitant qu'un nœud reste cluster-head pour une longue durée et épuise ainsi ses ressources rapidement mais en assurant aussi qu'il lui reste suffisamment pour apporter une stabilité à la structure.

L'algorithme Cluster-head Load-Balancing introduit une notion d'identifiant virtuel VID (Virtual IDentifier) utilisée pour l'élection des clusters-heads. Ce VID est à la base l'identifiant du nœud auquel est ajoutée une information de temps liée à la période pour laquelle un nœud peut rester cluster-head. Initialement, l'identifiant virtuel de chaque nœud est initialisé par son propre identifiant. A chaque période de temps, chaque nœud non cluster-head incrémente son identifiant virtuel jusqu'à ce que son VID atteigne une valeur maximale seuil MAX-VID. Le nœud ayant le plus grand identifiant virtuel dans son k- voisinage devient cluster-head. En cas d'égalité, le nœud ayant opéré le moins de fois comme cluster-head sera élu cluster-head et en cas d'égalité toujours, c'est le nœud qui a le plus grand identifiant ID deviendra cluster-head. Un nœud joue le rôle de cluster-head pour une période de temps Time-CH et après cette période, il initialisera son VID à zéro et abandonnera le statut de cluster-head.

Lorsque deux clusters-heads se trouvent à moins de k sauts l'un de l'autre, le cluster-head ayant le plus faible VID abandonnera son statut. Cet algorithme assure une certaine stabilité de la structure des k-clusters mais il nécessite durant l'élection des clusters-heads une synchronisation des nœuds (non cluster-heads) pour l'incrémementation de leurs VID et des cluster-heads pour comptabiliser la période pour jouer ce rôle.

Cette synchronisation est couteuse en nombre de messages échangés entre les différents nœuds. Alors un algorithme CLUBS été proposé qui forme des clusters à partir des diffusions locales et converge dans un temps proportionnel à la densité locale des nœuds [34].

La formation des clusters dans CLUBS est basée sur les trois caractéristiques suivantes :

- Chaque nœud dans le réseau doit être connecté à un cluster.
- Le diamètre maximal de tous les clusters dans le réseau doit être le même.
- Les clusters doivent supporter la communication intra-cluster.

L'algorithme forme des clusters avec un maximum de deux sauts. Chaque nœud dans le réseau participe à la formation des clusters en choisissant un nombre aléatoire dans un intervalle d'entiers fixe. Ensuite, il compte à rebours de ce nombre. Si le compte régressif n'avait pas été interrompu par un autre nœud voisin et il atteint le zéro, il s'annonce comme cluster-head et diffuse un message d'invitation. Quand un nœud voisin reçoit le message d'invitation qui vient depuis un diamètre de deux sauts. Il arrête le compte à rebours, accepte l'invitation et joint le cluster. Un nœud qui a joint un cluster « partisan » n'a plus le droit d'être cluster-head. Comme CLUBS autorise le chevauchement des clusters, les nœuds partisans restent en écoute des messages d'invitation supplémentaires et peuvent être partisan à plus qu'un cluster-head.

Si un nœud est en concurrence pour devenir cluster-head détecte une collision ou a reçu un message corrompu, il devient un nœud partisan et assume que plusieurs cluster-heads essayent de le recruter en même temps. Il peut trouver son cluster-head plus tard. L'algorithme se termine quand tous les nœuds dans le réseau joignent quelques clusters comme cluster-head ou comme un partisan. Quand deux cluster-heads se retrouvent à un saut. Leurs clusters s'écrouleront et le processus d'élection de cluster-head sera relancer.

II.5.2.2- les algorithmes à métrique liée à la topologie

Dans ces algorithmes une métrique est appelée associativité qui représente la stabilité relative des nœuds dans leurs voisinage. Pour chaque nœud, l'associativité comptabilise le temps que chacun des nœuds de son voisinage reste effectivement dans son voisinage et en fait la somme sur chaque voisin. A chaque période de temps, un nœud u considère quels sont ses voisins actuels déjà présents lors de la période précédente et ajoute 1 à la valeur associée à chacun d'eux. Si un voisin apparaît, il prend la valeur 1 et si un voisin a disparu, la valeur qui lui était associé prend 0. A chaque période de temps, l'associativité de u est la somme des valeurs associées à chacun de ses voisins. Cette valeur prend donc en compte la stabilité de u (si u est relativement stable dans son voisinage, il aura une forte associativité) et le degré des nœuds.

La formation des clusters suit le procédé suivant, un nœud considère les nœuds de son k-voisinage ayant un degré supérieur à une valeur seuil et élit parmi eux celui ayant la plus forte associativité. Dans le cas d'égalité, il choisit celui ayant le plus fort degré et si encore il y'a égalité, c'est celui ayant le plus faible identifiants qui sera élu. Les clusters résultants sont des k-clusters recouvrants mais qui visent à être plus stables dans le temps et dans l'espace que ceux se basant sur le simple degré [34].

II.5.2.3- les algorithmes à métrique spécifique au nœud mobile

HEED est un algorithme distribué de clusterisation proposé pour les réseaux des capteurs. Il construit un graphe connecté de cluster-heads à plusieurs sauts, les clusters formés sont disjoints. Les cluster-heads sont élus en se basant sur l'énergie résiduelle de chaque nœud. Les nœuds ayant une haute énergie résiduelle deviennent cluster-heads.

Cet algorithme invoque aussi une métrique secondaire utilisée pour rompre l'égalité. Le cas d'égalité dans ce contexte inclut le cas quand un nœud se trouve à proximité de deux cluster-heads et quand deux cluster-heads se trouvent dans le même rayon. Cette métrique est une métrique de cout de communication qui peut être fonction de la proximité des voisins ou la densité du cluster [39].

II.5.2.4- les algorithmes à métriques combinées

Mob Dhop (Mobility-based D-hop) [40] est un algorithme distribué qui forme des clusters de diamètre variable avec la mobilité du nœud. Cet algorithme suppose que chaque nœud peut mesurer la puissance du signal reçu. Cinq paramètres doivent être calculés :

- La distance estimée entre les nœuds : basée sur la puissance du signal reçu du voisin.
- La mobilité relative entre les nœuds est la différence de la distance estimée d'un nœud pour les deux moments successifs. Elle indique si les deux nœuds s'éloignent ou se rapprochent.
- La variation de la distance estimée en fonction du temps.

- La stabilité locale, elle est fonction de la variation de la distance estimée et de la mobilité relative.
- La distance moyenne estimée.

Cet algorithme se déroule en trois étapes : étape découverte, de fusion et de maintenance. L'étape de découverte a pour but de former des clusters à 2-sauts. Dans cette étape, les nœuds échangent périodiquement des messages Hello qui incluent la valeur de stabilité locale (initialisée à l'infini). Après, chaque nœud calcule sa valeur de stabilité locale et la diffuse à ses voisins. Le nœud ayant la plus petite valeur de stabilité locale est élu cluster-head et sa valeur de stabilité locale sera la valeur de stabilité du groupe GS. Le nœud qui entend des messages de plusieurs cluster-heads devient gateway, sinon il est membre. L'étape de fusion peut être initiée par un nœud non-spécifié qui demande de se joindre des clusters voisins ou quand deux gateways voisins demandent de fusionner leur clusters. Les deux conditions de fusion doivent être accomplies :

- La variation de la distance estimée entre deux nœuds fusionnés doit être inférieure ou égale au minimum de la valeur de la stabilité du groupe des deux clusters.
- La distance moyenne entre deux gateways doit être inférieure ou égale au maximum de la distance moyenne estimée des deux clusters.

La maintenance des clusters est invoquée quand un nœud arrive ou quitte le réseau.

- Quand un nœud arrive, il y'a fusion.
- Quand un cluster-head quitte le réseau, ses voisins exécutent l'étape de découverte et un nouveau cluster-head est élu. Durant la période sans cluster-head, les voisins à 2-sauts initialisent le processus de fusion pour joindre d'autres clusters si les conditions de fusion sont satisfaites.

II.6- VANET, CLOUD et Clusterisation

Les réseaux VANETs offrent plusieurs services (la sécurité routière, confort), malgré ça les capacités des véhicules (calcul, stockage) sont sous utilisés, d'où vient la proposition de former des clouds à partir des réseaux VANETs, qui va apporter de la valeur à ces réseaux.

Les réseaux CLOUDs offrent des services traditionnelles, classés sous trois catégories (SaaS : Software as a Service, PaaS : Plateform as a Service, IaaS : Infrastructure as a Service).

Le réseau CLOUD est composé de plusieurs machines interconnectées pour augmenter les capacités de calcul et de stockage, et ces machines sont installées dans des centres de calcul (Data Center) [41].

II.6.1- Architecture des réseaux CLOUDs

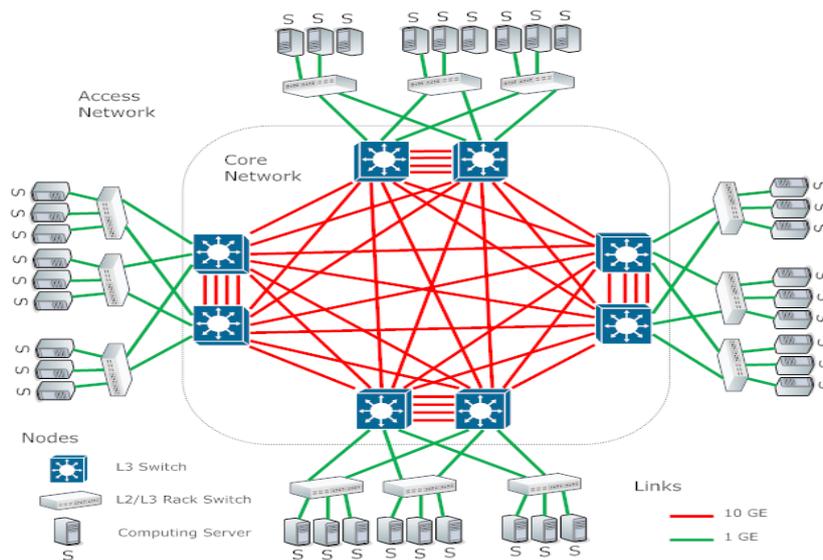


Figure II.4 : Architecture Two-tier [41].

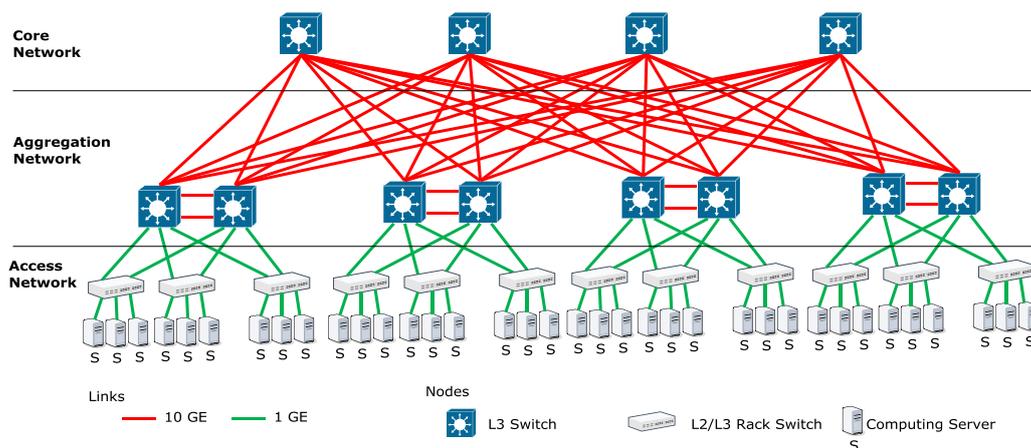


Figure II.5 : Architecture Three-tier [41].

II.6.2- Clusterisation

La clusterisation est déjà utilisée dans les réseaux VANETs (routage). Pour les réseaux VLOUDS (Vanet Cloud Computing) la clusterisation est un outil qui permet de structurer et organiser les véhicules pour former un cloud (vcloud datacenter).

La simulation est l'outil qui nous aide à étudier les performances des algorithmes de clusterisation pour les VLOUDS.

II.7- Conclusion

Dans ce chapitre, nous avons étudié la technique de clusterisation qui est largement utilisée dans les réseaux VANETs. Nous avons présenté une classification des algorithmes de clusterisation qui met en évidence les principales caractéristiques des approches proposées et nous avons donné une analyse détaillée du fonctionnement de ces algorithmes.

L'inconvénient majeur de la plupart des approches est de ne pas garder la structure des clusters stable dans un environnement dense, mobile et à large échelle. Fournir un mécanisme de clusterisation stable est important pour le bon fonctionnement des applications. La stabilité de la structure des clusters permet de garantir la stabilité des routes et donc réduire le coût de leur maintenance et améliorer les performances du réseau.

Le prochain chapitre sera consacré à la présentation de l'outil de simulation NS2 et les résultats obtenus lors de la simulation.

III.1-Introduction

Ce dernier chapitre sera réservé à la présentation des résultats de simulation. L'objectif principal de notre simulation est l'étude des performances des protocoles de routages AODV, DSDV et DSR appliqués aux réseaux VANETs. Dans le but d'évaluer ces protocoles, nous étions amenés à l'intégrer sous le simulateur NS2. Pour cela, il nous a été nécessaire dans un premier temps de présenter l'outil de simulation NS2 et son architecture et ses composants. Puis on passe à la description de scénario de simulation.

III.2-La simulation du réseau VANET [12]

La simulation est l'implantation d'un modèle simplifié du système à l'aide d'un programme de simulation adéquat. Cette méthode traduit le comportement du système à évaluer d'une manière réaliste. La simulation permet de tester et visualiser à moindre coût les résultats sous forme de graphes faciles à analyser et à interpréter, elle n'est pas une méthode exacte et nécessite de prêter une attention particulière aux interprétations des résultats de simulation.

Il existe plusieurs outils de simulation, citons à titre d'exemple : NS2 (Network Simulator 2), GloMoSim (Global Mobile Simulator), Opnet (Open Network) et JistSwan. Pour faciliter le choix d'un simulateur, des aspects peuvent être considérés tels que :

- Bibliothèque de modèles : Si l'on souhaite utiliser un protocole déjà inclus dans la bibliothèque, il est alors inutile de l'implémenter.
- Fiabilité du simulateur et des protocoles simulés : La fiabilité des protocoles inclus dans le simulateur est primordiale pour rendre la mesure de performances d'un protocole la plus fidèle à la réalité.
- Performances brutes : Se mesurent en temps d'exécution et en utilisation de la mémoire. Si nous souhaitons simuler un réseau comportant un grand nombre de nœuds, le temps d'exécution doit rester raisonnable et la mémoire utilisée adaptée à la machine exécutant le simulateur.
- Facilité d'extension : La facilité d'ajout de nouveaux modèles au simulateur est primordiale pour en évaluer les performances.

- Mesure de performances : Certains simulateurs incluent la génération automatique de statistiques en fonction de différentes métriques. Type de réseau Architecture : (filaire ou ad hoc) ou ses applications.
- Licence de distribution : Définit les droits d'utilisation du logiciel, les droits de diffusion (duplication) et les droits de modification.

Notre choix s'est porté sur le Network Simulator NS2 et ceci pour différentes raisons qui sont:

- Son modèle libre permet l'ajout très rapide de modèles correspondant à des technologies émergentes. Ceci le rend meilleur logiciel de simulation par événements discrets d'après les spécialistes des télécommunications.
- Le logiciel NS2 est fourni avec une interface graphique (NAM) permettant de démontrer le fonctionnement des réseaux.
- Il permet d'étendre le simulateur et de le personnaliser, et il donne une liberté de programmation vu qu'il est fourni avec son code source qu'on peut modifier et recompiler autant qu'on le souhaite.

Pour la plate forme notre choix s'est porté sur la plate forme de Linux pour la simple raison que NS2 est nettement plus facile à installer et à configurer sous Linux que sous Windows. Nous avons travaillé avec ns-allinone-2.34 sous Linux ubuntu 12.04.

III.3-Présentation de NS2

Le simulateur du réseau NS2 est un outil logiciel de simulation de réseaux informatiques. Il est principalement bâti avec les idées de la conception par objets, de réutilisation du code et de modularité. NS2 est écrit en C++ et utilise le langage OTCL (Object Tools Command Language) dérivé de TCL. A travers OTCL, l'utilisateur décrit les conditions de la simulation : la topologie du réseau, les caractéristiques des liens physiques, les protocoles utilisés et les communications qui ont lieu. La simulation doit d'abord être saisie sous forme de fichier que NS va utiliser pour produire un fichier contenant les résultats. Mais l'utilisation de l'Otel permet aussi à l'utilisateur de créer ses propres procédures (par exemple s'il souhaite enregistrer dans un fichier l'évolution d'une variable caractéristique du réseau au cours du temps).

Il contient les fonctionnalités nécessaires à l'étude des algorithmes de routage unicast ou multicast, des protocoles de transport, de réservation, des services intégrés et des protocoles d'application. De plus le simulateur possède déjà une palette de systèmes de transmission, d'ordonnanceurs et de politiques de gestion de files d'attente pour effectuer des études de contrôle de congestion [42]. La liste des principaux composants actuellement disponible dans NS par catégorie est :

- Application Web, ftp, telnet, générateur de trafic (CBR, ...).
- Transport TCP, UDP, RTP, SRM.
- Routage statique ou dynamique (vecteur de distance).
- Routage Multicast (DVMRP, PIM).
- Gestion de file d'attente : RED, DropTail, Token bucket, etc.
- Discipline de service : CBQ, SFQ, DRR, Fair Queueing.
- Système de transmission : CSMA/CD, CSMA/CA, lien point à point.

III.4-Utilisation de NS2

NS2 permet à l'utilisateur de définir un réseau et de simuler des communications entre les nœuds de ce réseau, il utilise le langage OTCL (une extension objet du langage tcl) dérivé de TCL [12].

- TCL (Tools Command Language) est un langage interprété, traité par un interpréteur TCL (NS par exemple). Les programmes écrits en Tcl sont en fait des fichiers texte constitués de commandes. Ces fichiers sont nommés scripts.
- OTCL est une extension orientée objet de Tcl, il utilise les fonctions standards du langage Tcl mais également ajoute des commandes à l'interpréteur.
- Tk (Tool kit) est un langage interprété permet de construire des interfaces graphiques.

III.5-Architecture et Implémentation

L'architecture réseau de NS2 est fortement basée sur le modèle des couches OSI, il s'agit de la décomposition de la pile réseau en couches. Au plus bas niveau de NS2, il y a six classes qui définissent l'ensemble de la structure du Programme et fournissent les méthodes élémentaires, il s'agit des classes TCL, TclObject, TclClass, TclCommand, EmbeddedTcl, InstVar. Elles définissent entre autres les méthodes utilisées par C++ pour accéder à l'interpréteur, la hiérarchie. Les principales commandes de haut niveau et les Méthodes d'accès aux variables C++ et Otcl.

La simulation est configurée, contrôlée et gérée à l'aide des interfaces fournies par la classe OTclSimulator, cette classe fournit des procédures pour créer et gérer la topologie, initialiser le format des paquets et choisir le planificateur d'événements, elle stocke intérieurement des références à chaque élément de la topologie, un script donc devra toujours commencer par l'instanciation d'une variable de cette classe. L'utilisateur crée ensuite la topologie à travers OTcl en utilisant les classes Node et Link, composants essentiels de la topologie. Ces éléments sont décrits dans la sous section suivante [12].

III.5.1-Composants de la topologie [43] [44]

La topologie NS2 est essentiellement composée de nœuds et de liens :

- **Les Nœuds :** La définition des nœuds se fait dans un premier temps à travers l'instance de Simulator puis à travers l'instance de la classe Node, la fonction d'un nœud est de recevoir des paquets, les examiner et les mapper à ses interfaces sortantes appropriées. Cette classes est composée d'un classificateur et de méthodes pour configurer un nœud, les méthodes proposées sont des fonctions de contrôle, de gestion d'adresse et de port, de gestion d'agents et de repérage des voisins, le classificateur est la partie du nœud qui traite chaque segment des paquets reçus, il en existe donc plusieurs, chacun étant spécifique au champs examiné.
- **Les liens :** Les liens constituent la deuxième partie de la topologie, les liens entre les nœuds sont définit dans la classe Link et Simple Link plus précisément lorsqu'il s'agit

de relier deux nœuds. Plusieurs types de liaisons sont supportés, comme le point à point, le broadcast ou les liaisons sans fil pour la mobilité.

- **La gestion des files d'attente :** La gestion des files d'attente et la simulation des délais sur les liens sont implémentés dans les classes Queue et Link Delay respectivement. Les files d'attente actuellement disponible dans NS sont :
 - FIFO.
 - RED buffer management.
 - CBQ (priorité et circulaire) ...etc.
- **Les agents :** L'agent est un autre composant d'un nœud. Il modélise les constructeurs et les consommateurs de paquets IP. La classe agent fournit des méthodes utiles au développement de la couche transport et à d'autres protocoles du plan de signalisation ou de gestion. Cette classe est à la fois dans l'interpréteur et dans le simulateur. C'est la classe de base pour définir des nouveaux protocoles dans NS. Elle fournit l'adresse locale et de destination, les fonctions pour générer les paquets, l'interface à la classe application. Actuellement NS comporte de nombreux agents citons: UDP, protocoles de routage, différentes versions de TCP, RTP, ...etc.

III.5.2-NAM (Network Animator)

Le NAM est un outil d'animation basé sur Tcl/TK, utilisé dans NS afin d'interpréter et de visualiser le tracé de simulation des réseaux, ainsi que le tracé de données. Le NAM peut être démarré en tapant la commande « nam <nom du fichier NAM (.nam)> », dans le Shell de Linux [12].

III.5.3-Fichier trace (résultat de la simulation)

Le fichier trace est un fichier qui retrace tous les évènements qui se sont produits pendant la simulation à savoir émission, réception, perte de paquets ainsi que le passage des paquets par les différentes couches du modèle TCP/IP. Il existe deux formats du fichier trace l'ancien format et le nouveau format. Nous avons opté pour le nouveau format car il est beaucoup plus explicite que l'ancien [12].

Ce type de fichier est généré en spécifiant dans le script Tcl, la commande : `$ns use-newtrace`
Avec « ns » est le nom de l'instance de simulation. Chaque valeur notée dans ce fichier est précédée de sa signification. Il contient des informations sur :

- N: Propriétés des nœuds.
- I: Informations sur le niveau IP du paquet.
- H: Informations sur le nœud suivant.
- M: Informations sur le niveau MAC du paquet.

III.6-Préparation de l'environnement d'implémentation

La préparation de l'environnement d'implémentation consiste à installer le simulateur de réseau NS2 sous le système d'exploitation linux Nous avons utilisé la version NS-2.34 [45].

L'installation s'effectue en étapes :

- Décompresser le package `ns-allinone-2.34.tar.gz`
- A partir du répertoire `ns-allinone-2.34` qui résulte de la décompression, lancer la console de commandes (ou la touche F4).
- Lancer la commande « `./install` », Cette commande spécifie les chemins des fichiers et répertoires de NS selon le homeloin de l'utilisateur, i.e. le chemin de décompression de NS. Cependant, NS n'est encore pas utilisable. Pour pouvoir utiliser NS, il faut définir ces chemins dans les variables d'environnement de LINUX.
- bibliothèques : « `gedit ~/.bashrc` » si on est sur GNOME ou « `kate ~/.bashrc &` » si on est sur KATE.
- Après enregistrement du fichier, nous lançons la commande `$ source ~/.bashrc`
- Pour vérifier que NS est accessible sur Linux, la commande « `ns` » doit afficher « % ».
- A partir du répertoire « `Home/home login/ns-allinone-2.31/ns-2.31` », lancer la dernière commande « `./validate` » qui génère les fichiers « .o » (object) de NS et permet ainsi d'exécuter des simulations.



III.7-Implémentation de la simulation

L'objectif principal de notre simulation est l'étude des performances des protocoles de routage AODV, DSDV et DSR dans les deux standards 802.11 et 802.11p appliqués aux réseaux VANETs. Par conséquent, pour pouvoir obtenir des résultats de simulation nous avons comparé ces protocoles suivant les paramètres cités ci-dessous :

- La bande passante : c'est le volume total d'informations qui peut absorber un lien entre deux nœuds sans créer de file d'attente.
- Le Délai moyen de transfert d'un paquet de donnée : C'est le délai moyen pris par un paquet de données pour transiter d'une application à une autre.
- La mobilité : Elle indique le mouvement des nœuds, elle peut être faible ou forte. Le calcul se fait en mesurant le mouvement relatif d'un nœud par rapport aux autres

Dans notre simulation, chacun des protocoles (DSDV, AODV et DSR) va être testé sous des conditions différentes afin de prédire lequel des trois est meilleur dans un monde nomade. En se basant sur deux paramètres qui sont la bande passante et le délai, dans les deux standards :

- Le standard 802.11 : c'est un ensemble de normes concernant les réseaux sans fil locaux (le wifi). Qui ont été mises au point par le groupe de travail 11 du comité de normalisation LAN/NAM de l'IEEE (IEEE.802). son but est d'offrir une connectivité sans fil à des stations fixes ou mobiles qui demandent un déploiement rapide au sein d'une zone locale en utilisant différentes bandes de fréquences [17].
- Le standard 802.11p : Cette extension du standard concerne les communications de service entre des véhicules et une infrastructure ou entre véhicules, dans la bande 5,9 GHz dédiée à cet usage. Une contrainte est un temps de réponse très court afin de réaliser une transaction en moins de 50 ms sur une courte distance [17].

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



Figure III.1: Les zones urbaines pris en compte dans chaque scénario.

Zones	Nombre de nœuds	Terrain de simulation
Zone 1	10, 15, 20	X1=600, Y1=400
Zone 2	20, 30, 40	X2=600, Y2=500
Zone 3	40, 45, 60	X3=600, Y3=700

Tableau III.1 : Paramètres des zones.

Paramètres	Valeur
Simulateur	NS-2.34
paramètres simulés	Bande passante, Délai
Protocoles	AODV, DSDV, DSR
Zones	Z1, Z2, Z3
Standards	802.11, 802.11.p
Taille des paquets	100
Temps de simulation	180

Tableau III.2 : Tableau récapitulatif des paramètres de simulation.

III.8-Résultats de la simulation

III.8.1-Comparaison entre les deux standards

Nous allons comparer les deux standards dans trois zones différentes par rapport à un seul paramètre qui est la bande passante.

III.8.1.1-Scénario de la zone1

- Le protocole AODV

	10	15	20
80211	0,03449	0,034914	0,030208
802.11P	0,03442	0	0,030208

Tableau III.3 : La valeur moyenne de la bande passante pour AODV.

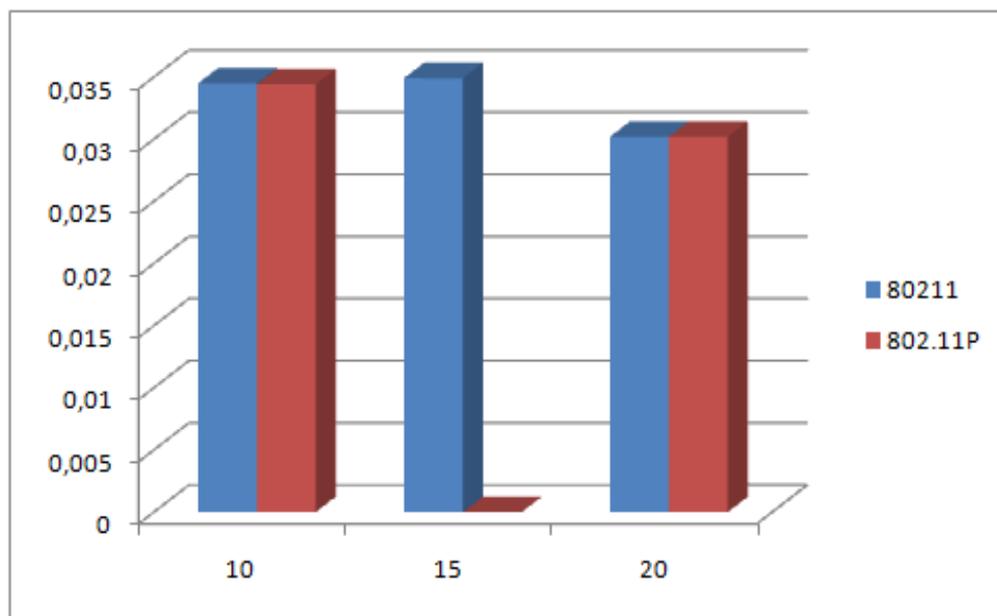


Figure III.2: Résultats de simulation de la bande passante pour AODV.

« Figure III.2 » met en évidence les performances des standards 802.11 et 802.11p par rapport à la bande passante avec un nombre variable des nœuds 10,15 et 20. De la figure, nous observons que les standards 802.11 et 802.11p ont donnés les mêmes résultats par rapport aux nombres des nœuds 10 et 20, par contre dans le nombre des nœuds 15 le standard 802.11 a donné de meilleurs résultats que le 802.11p.

- Le protocole DSDV

	10	15	20
802.11	0,03449	0,034396	0,033784
802.11P	0,034373	0,034655	0,034467

Tableau III.4: La valeur moyenne de la bande passante pour DSDV.

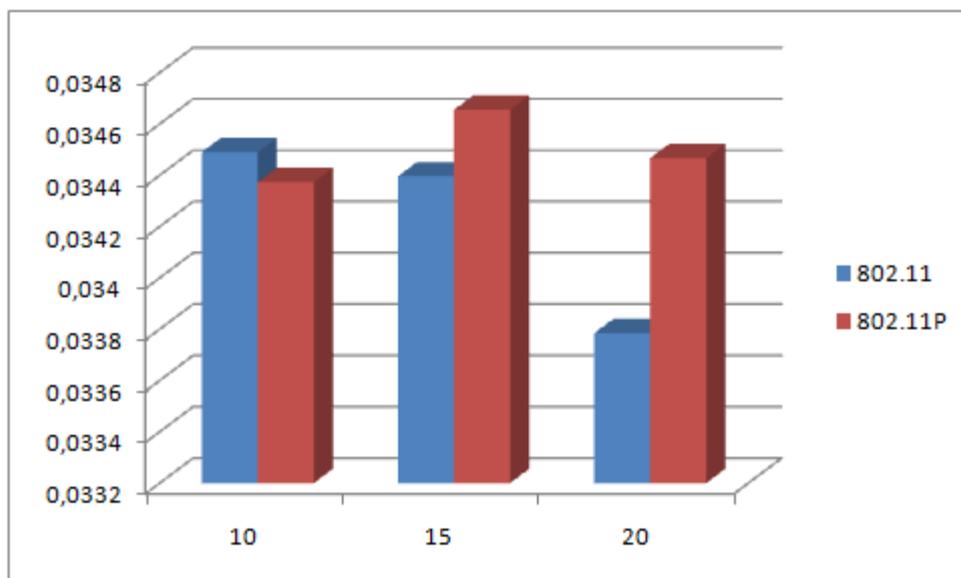


Figure III.3: Résultats de simulation de la bande passante pour DSDV.

« La figure III.3 » met en évidence les performances relatives des standards 802.11 et 802.11p par rapport à la bande passante avec différents nombres des nœuds 10, 15 et 20. De la figure, nous observons que le standard 802.11p a donné de meilleurs résultats dans les nœuds 15 et 20. Contrairement au nombre des nœuds 10, le standard 802.11 a donné de meilleurs résultats.

- Le protocole DSR

	10	15	20
802.11	0,032945	0,032763	0
802.11P	0,032945	0,032446	0,033103

Tableau III.5 : La valeur moyenne de la bande passante pour DSR.

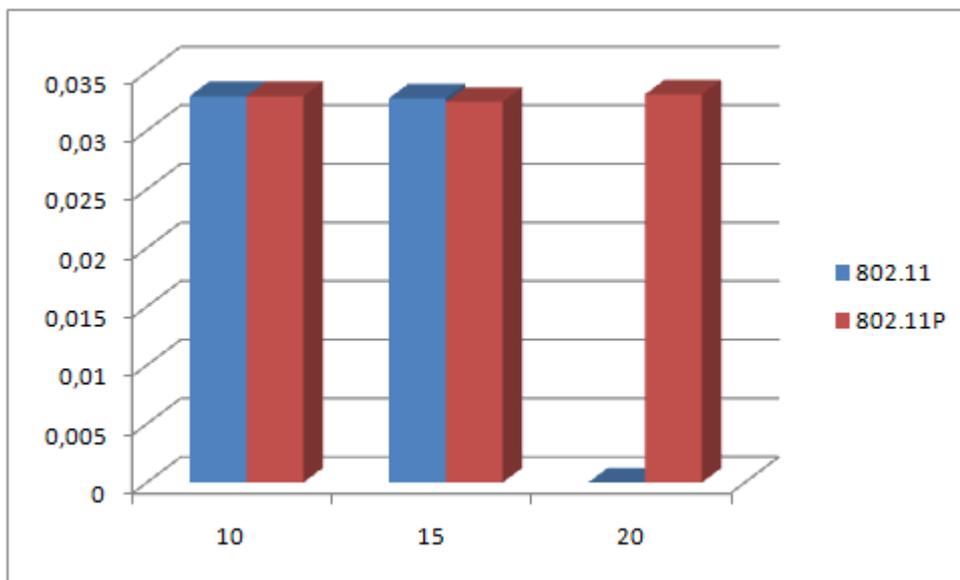


Figure III.4: Résultats de simulation de la bande passante pour DSR.

« La figure III.4 » met en évidence les performances relatives des standards 802.11 et 802.11p Par rapport à la bande passante avec différents nombres des nœuds 10, 15 et 20. De la figure nous observons que le standard 802.11p a donné de meilleurs résultats que le standard 802.11.

- Dans le scénario de la zone 1, les résultats obtenus par le standard 802.11p sont meilleurs, le standard délivre un volume important d’informations dans les liens entre les nœuds sans créer la file d’attente.

III.8.1.2-Scénario de la zone2 :

- Le protocole AODV

	20	30	40
802.11	0,034161	0,027856	0,024162
802.11P	0,035173	0,034396	0,034208

Tableau III.6 : La valeur moyenne de la bande passante pour AODV.

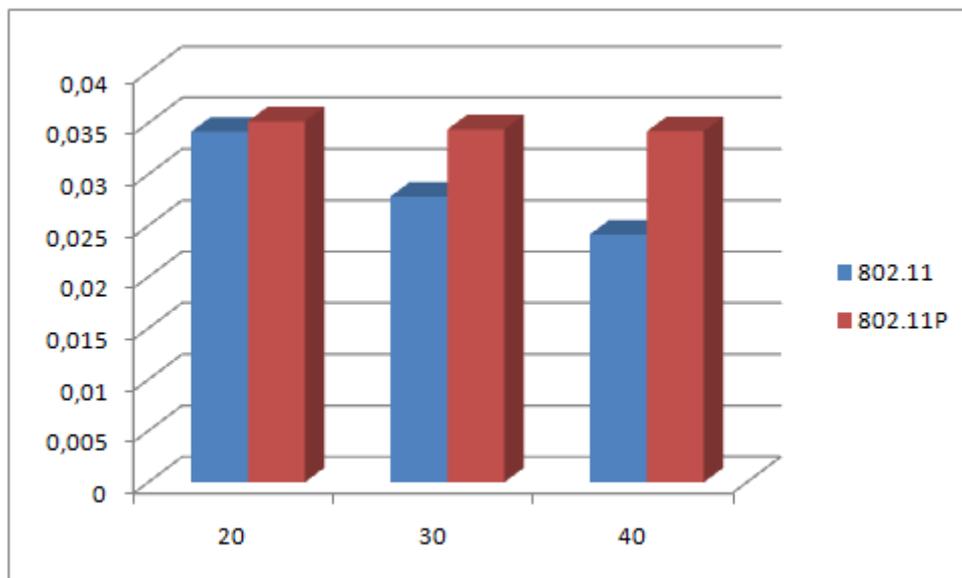


Figure III.5: Résultats de simulation de la bande passante pour AODV.

« La figure III.5 » met en évidence les performances relatives des standards 802.11 et 802.11p par rapport à la bande passante avec un nombre variable des nœuds 20, 30 et 40. De la figure nous observons que le standard 802.11p a donné de meilleurs résultats que le standard 802.11 et ceci est valable pour les différents nombres des nœuds.

- Le protocole DSDV

	20	30	40
802.11	4,513489	0,025903	0,022515
802.11P	4,572283	0,03449	0,03449

Tableau III.7 : La valeur moyenne de la bande passante pour DSDV.

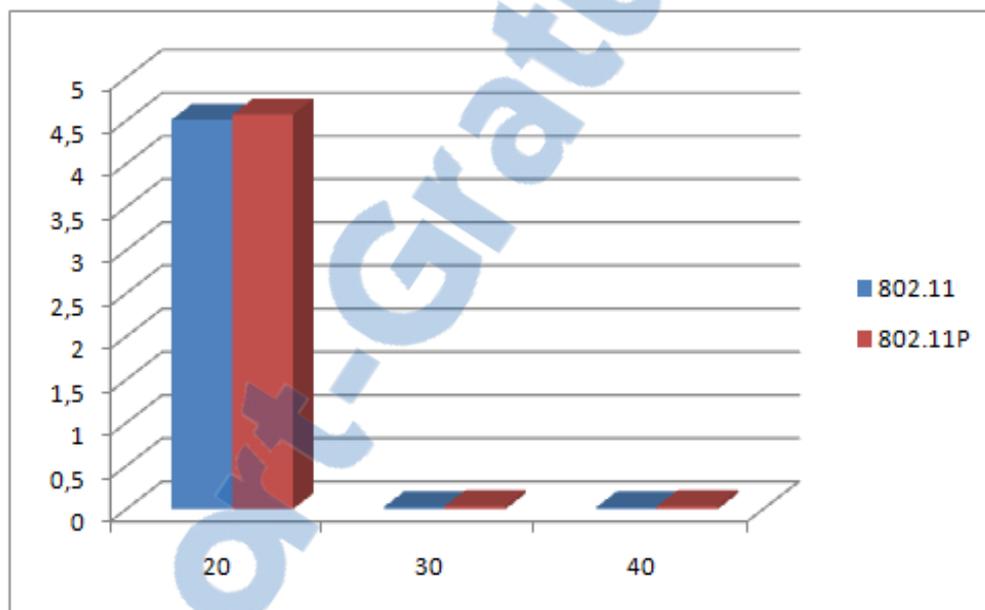


Figure III.6: Résultats de simulation de la bande passante pour DSDV.

« La figure III.6 » met en évidence les performances relatives des standards 802.11 et 802.11p par rapport à la bande passante avec un nombre variable des nœuds 20, 30 et 40. De la figure nous observons que dans le nombre des nœuds 20, les deux standards ont donnés de meilleurs résultats, mais pour le nombre des nœuds 30 et 40 les résultats sont nuls, ceci signifie qu'aucun paquet n'a été reçu par la destination et donc que le protocole DSDV a été incapable d'acheminer les paquets jusqu'à la destination.

- Le protocole DSR

	20	30	40
802.11	0,032922	0,023639	0,022257
802.11P	0,033375	0,033443	0,033194

Tableau III.8 : La valeur moyenne de la bande passante pour DSR.

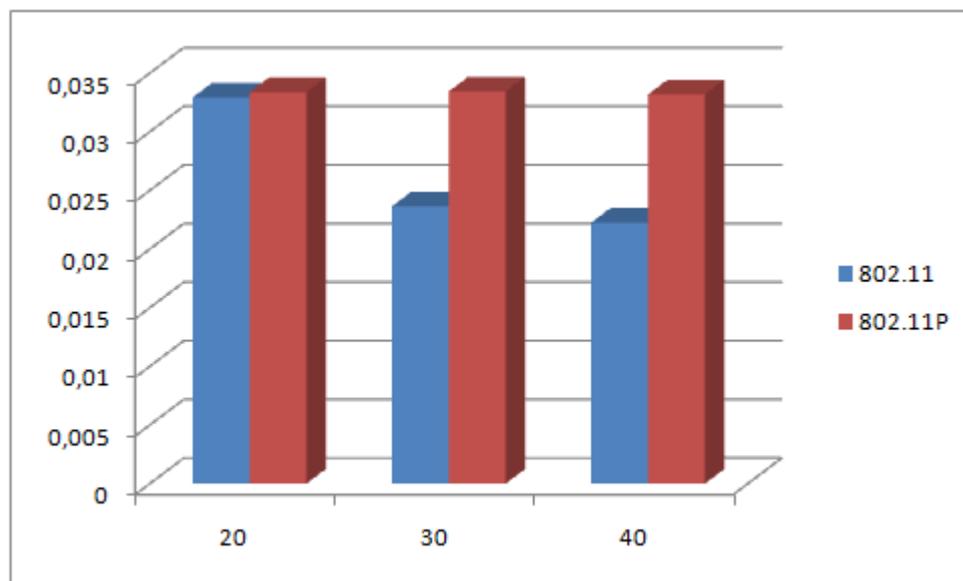


Figure III.7: Résultats de simulation de la bande passante pour DSR.

« La figure III.7 » met en évidence les performances relatives des standards 802.11 et 802.11p par rapport à la bande passante avec un nombre variable des nœuds 20, 30 et 40. De la figure nous observons que le standard 802.11p a donné de meilleurs résultats que le standard 802.11.

- Dans le scénario de la zone 2, les résultats obtenus par le standard 802.11p sont meilleurs, le standard délivre un volume important d'informations dans les liens entre les nœuds sans créer la file d'attente.

III.8.1.3-Scénario de la zone3 :

- Le protocole AODV

	30	45	60
802.11	0,30091	0,034349	0
802.11P	0,034373	0,03449	0,000235

Tableau III.9: La valeur moyenne de la bande passante pour AODV.

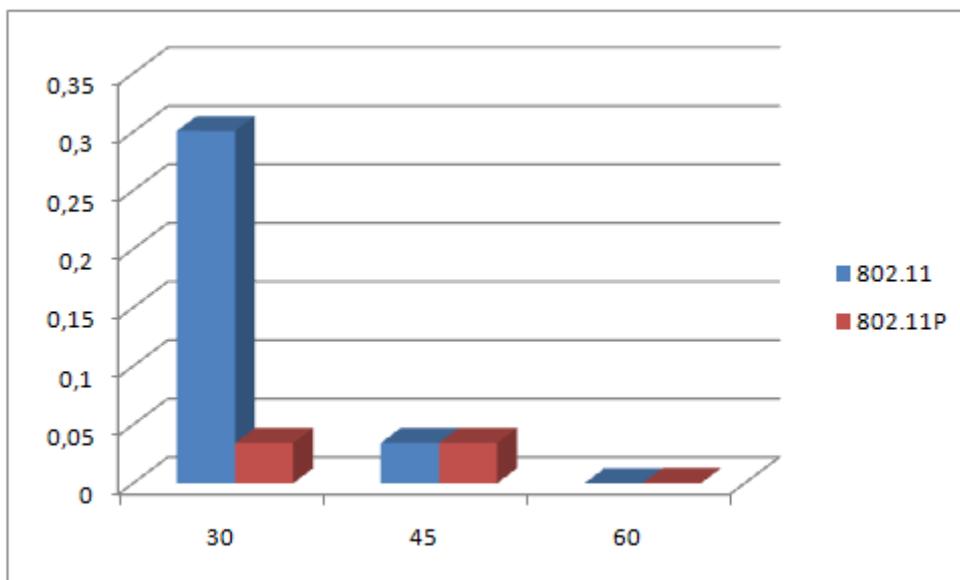


Figure III.8: Résultats de simulation de la bande passante pour AODV.

« La figure III.8 » met en évidence les performances relatives des standards 802.11 et 802.11p par rapport à la bande passante avec un nombre variable des nœuds 30, 45 et 60. De la figure nous observons que pour le nombre des nœuds 30, le standard 802.11 a donné de meilleurs résultats mais pour le nombre des nœuds 45 les résultats sont les mêmes dans les deux standards, par contre dans le nombre des nœuds 60 les résultats sont nuls, ceci signifie qu'aucun paquet n'a été reçu par la destination et donc que le protocole AODV a été incapable d'acheminer un seul paquet jusqu'à la destination.



- Le protocole DSDV

	30	45	60
802.11	0,030561	0,034326	0,021386
802.11P	0,034867	0,034608	0,03449

Tableau III.10 : La valeur moyenne de la bande passante pour DSDV

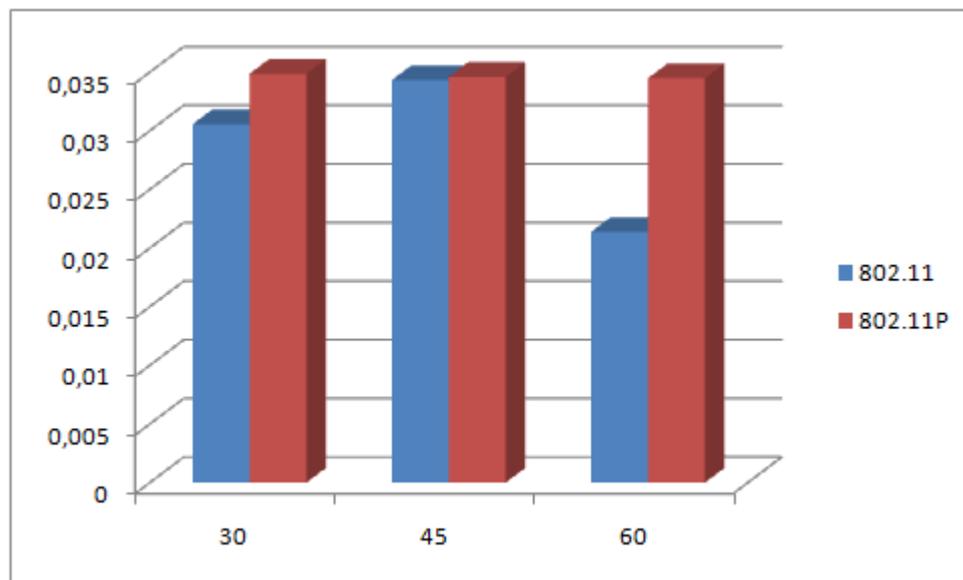


Figure III.9: Résultats de simulation de la bande passante pour DSDV.

« La figure III.9 » met en évidence les performances relatives des standards 802.11 et 802.11p par rapport à la bande passante avec un nombre variable des nœuds 30, 45 et 60. De la figure nous observons que le standard 802.11p a donné de meilleurs résultats que le standard 802.11.

- Le protocole DSR

	30	45	60
802.11	0,029888	0,033511	0,018816
802.11P	0,032809	0,033171	0,0322

Tableau III.11 : La valeur moyenne de la bande passante pour DSR.

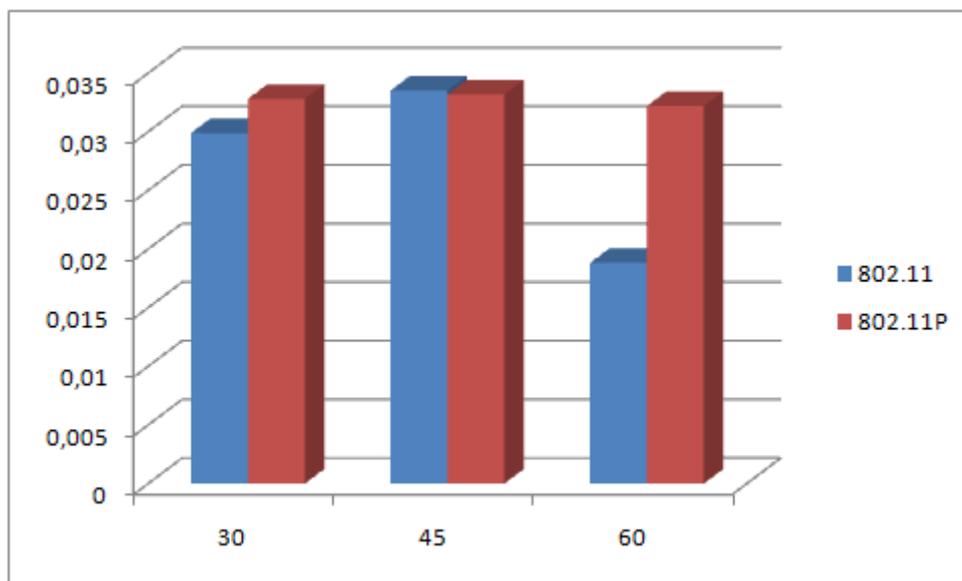


Figure III.10: Résultats de simulation de la bande passante pour DSR.

« La figure III.10 » met en évidence les performances relatives des standards 802.11 et 802.11p par rapport à la bande passante avec un nombre variable des nœuds 30, 45 et 60. De la figure nous observons que le standard 802.11p a donné de meilleurs résultats que le standard 802.11 dans les différents nombres des nœuds.

- Dans le scénario de la zone 3, les résultats obtenus par le standard 802.11.p sont meilleurs, le standard délivre un volume important d'informations dans les liens entre les nœuds sans créer la file d'attente.

Cette comparaison a été amenée pour évaluer les performances des deux standards par rapport au trois protocoles AODV, DSDV et DSR. Ces deux standards ont été comparés en terme de bande passante dans trois zones différentes.

D'après les résultats obtenus dans les trois zones, on conclut que le standard 802.11p a des meilleures performances que le standard 802.11 pour la communication dans les réseaux VANETs. En se basant sur ces résultats nous avons appliqué cette norme dans la suite de notre simulation.

III.9- Performances des protocoles de routage avec 802.11p

Nous allons comparer les trois protocoles de routages dans trois zones différentes suivant deux paramètres : la bande passante et le délai.

III.9.1-Scénario de La zone1

- la bande passante

	10	15	20
AODV	0,03442	0	0,030208
DSDV	0,034373	0,034655	0,034467
DSR	0,032945	0,032446	0,033103

Tableau III.12 : La valeur moyenne de la bande passante dans la zone 1.

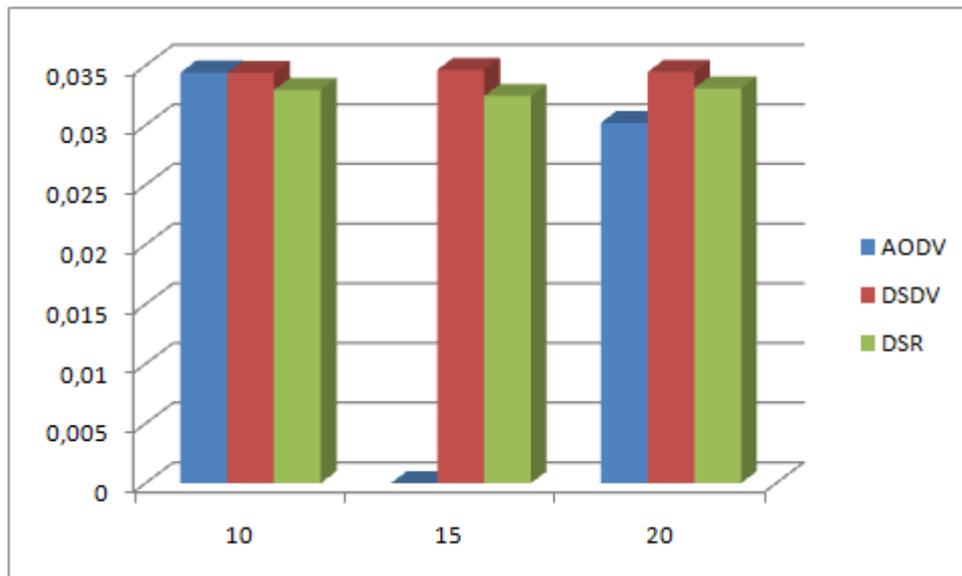


Figure III.11: Résultats de simulation de la bande passante dans la zone 1.

« La figure III.11 » met en évidence les performances relatives des protocoles de routages AODV, DSDV et DSR par rapport à la valeur moyenne de la bande passante avec un nombre variable des nœuds 10, 15 et 20. De la figure nous observons que les deux protocoles DSDV et DSR ont donnés des meilleurs résultats dans les différents nombres des nœuds, et pour le protocole AODV a donné des bons résultats pour les nœuds 10 et 20 et un résultat nul dans le nombre des nœuds 15.

- le délai

	10	15	20
AODV	0,023674	0	0,02948
DSDV	0,026205	0,02293	0,023591
DSR	0,02374	0,023636	0,023661

Tableau III.13 : La valeur moyenne du délai dans la zone 1.

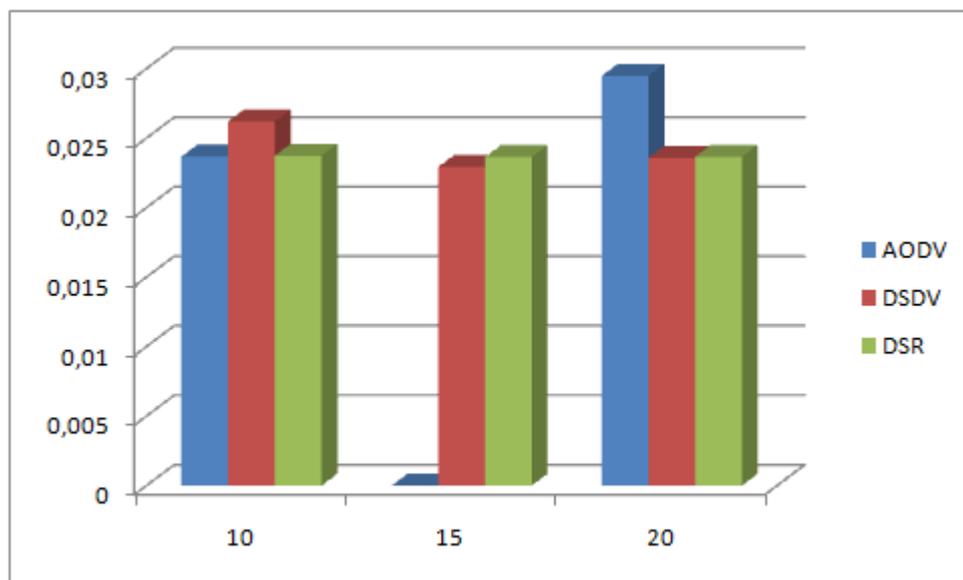


Figure III.12: Résultats de simulation du délai dans la zone 1.

« La figure III.12 » met en évidence les performances relatives des protocoles de routages AODV, DSDV et DSR par rapport à la valeur moyenne du délai avec un nombre variable des nœuds 10, 15 et 20. de la figure nous observons que les deux protocoles DSDV et DSR ont donnés de meilleurs résultats dans les différents nombres des nœuds, et pour le protocole AODV a donné de bons résultats pour les nœuds 10 et 20 et un résultat nul dans le nombre des nœuds 15.

III.9.2-Scénario de la zone 2 :

- la bande passante

	20	30	40
AODV	0,035173	0,034396	0,034208
DSDV	4,572283	0,03449	0,034279
DSR	0,033375	0,033443	0,033194

Tableau III.14 : La valeur moyenne de la bande passante dans la zone 2.

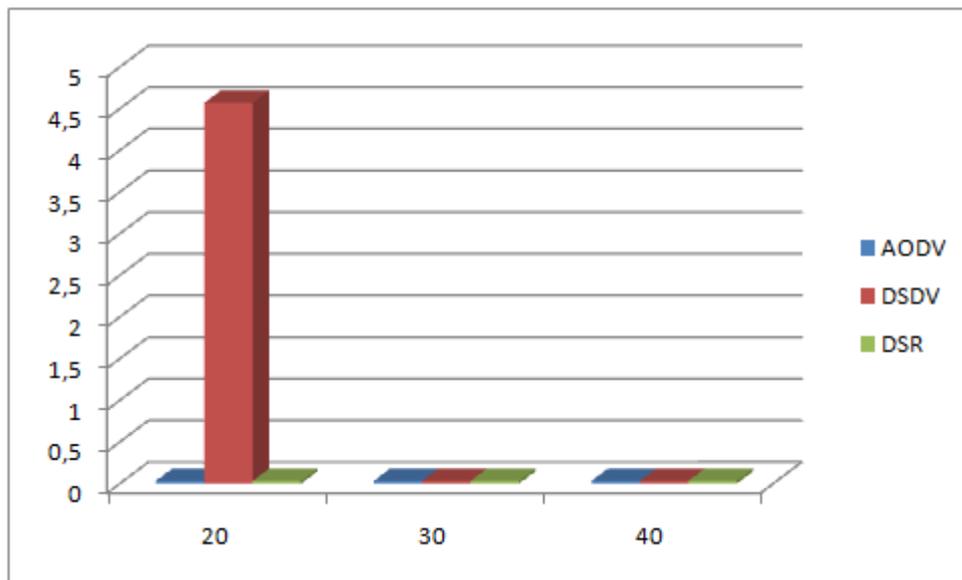


Figure III.13: Résultats de simulation de la bande passante dans la zone 2.

« La figure III.13 » met en évidence les performances relatives des protocoles de routages AODV, DSDV et DSR par rapport à la valeur moyenne de la bande passante avec un nombre variable des nœuds 20, 30 et 40. De la figure nous observons que dans les trois protocoles les résultats sont nuls dans les différents nœuds, cela signifie qu'aucun paquet n'a été reçu par la destination et donc que les protocoles ont été incapables d'acheminer un seul paquet jusqu'à la destination, mais uniquement dans le nombre des nœuds 20, le protocole AODV a donné de bons résultats.

- le délai

	20	30	40
AODV	0,022197	0,032121	0,038865
DSDV	0,024437	0,037338	0,038799
DSR	0,022385	0,034283	0,038758

Tableau III.15 : La valeur moyenne du délai dans la zone 2.

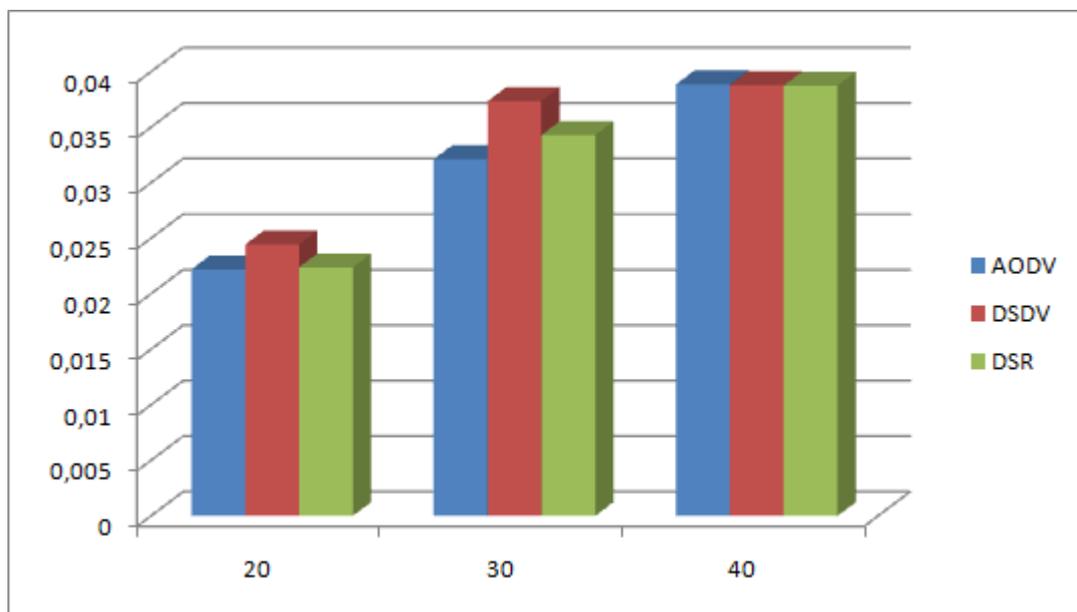


Figure III.14: Résultats de simulation du délai dans la zone 2.

« La figure III.14 » met en évidence les performances relatives des protocoles de routages AODV, DSDV et DSR par rapport à la valeur moyenne du délai avec un nombre variable des nœuds 20, 30 et 40. De la figure nous observons que les résultats sont symétriques dans les trois protocoles dans le nombre des nœuds 40 et pour le nombre des nœuds 20 et 30 c'est le protocole DSDV qui donne de meilleurs résultats.

III.9.3-Scénario de la zone 3 :

- la bande passante

	30	45	60
AODV	0,034373	0,03449	0,000235
DSDV	0,034867	0,03449	0,03449
DSR	0,022608	0,033171	0,0322

Tableau III.16 : La valeur moyenne de la bande passante dans la zone 3.

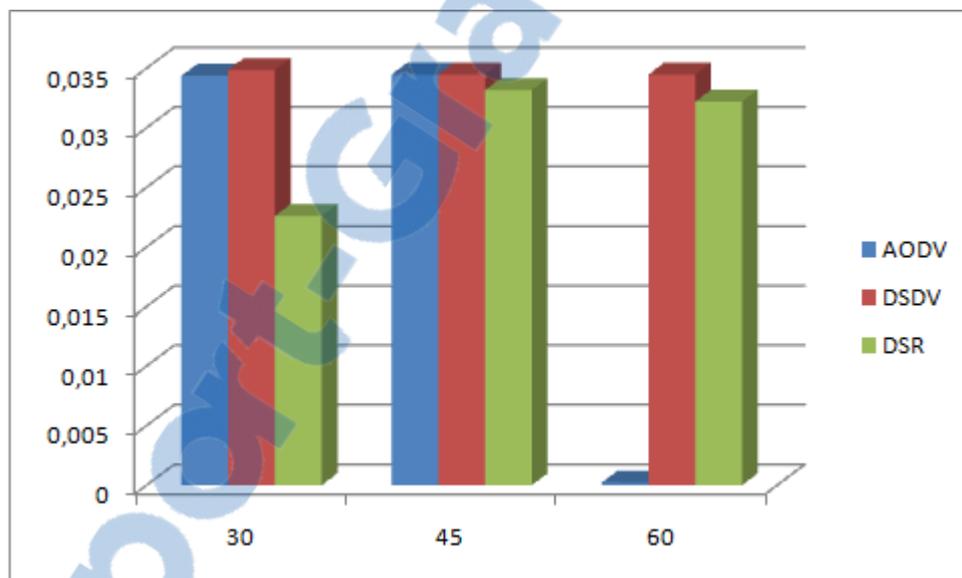


Figure III.15: Résultats de simulation de la bande passante dans la zone 3.

« La figure III.15 » met en évidence les performances relatives des protocoles de routages AODV, DSDV et DSR par rapport à la valeur moyenne de la bande passante avec un nombre variable des nœuds 30, 45 et 60. De la figure nous observons que les deux protocoles DSDV et DSR ont donnés de meilleurs résultats dans les différents nombres des nœuds, et pour le protocole AODV a donné de meilleurs résultats pour les nœuds 30 et 45, et un résultat nul dans le nombre des nœuds 60.

- Pour le délai

	30	45	60
AODV	0,023072	0,038741	0,0114259
DSDV	0,02421	0,034531	0,02624
DSR	0,022608	0,038756	0,031187

Tableau III.17 : La valeur moyenne du délai dans la zone 3.

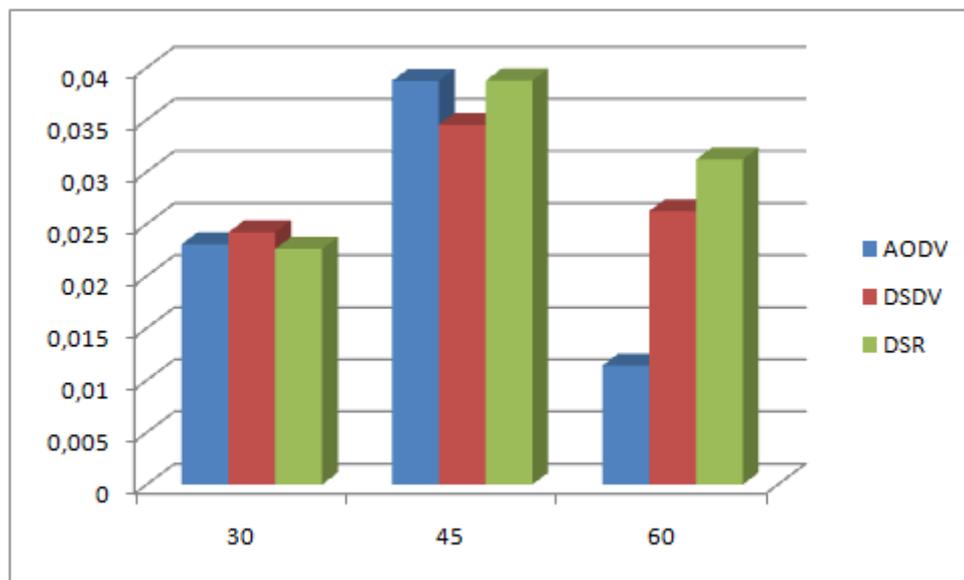


Figure III.16: Résultats de simulation du délai dans la zone 3.

« La figure III.16 » met en évidence les performances relatives des protocoles de routages AODV, DSDV et DSR par rapport à la valeur moyenne du délai avec un nombre variable des nœuds 30, 45 et 60. De la figure, nous observons que les protocoles ont donné des résultats symétriques dans le nombre des nœuds 30, 45, et un résultat différent pour le nombre des nœuds 60. Ou le protocole AODV a donné des résultats moins importants.

III.10- Conclusion

Notre but à travers la simulation été de tester les performances des protocoles de routage dans les deux standards 802.11 et 802.11p.

Suite aux résultats obtenus nous avons constaté que les protocoles de routage les plus courants dans les réseaux VANETs (AODV, DSDV et DSR) qu'il n y a pas un protocole qui donne des bons résultats, ce qui nous mène à déduire que ces protocoles (tel qu'ils sont définit) ne sont pas vraiment adaptés aux réseaux VANETs à cause de la nature de ces réseaux (forte mobilité, les composants tel que OBU, RSU).

Conclusion générale

Les systèmes de transport intelligents ne sont qu'à leurs balbutiements. À termes, le développement des nouvelles technologies a favorisé une formidable évolution des réseaux véhiculaires. Cette évolution vise à rendre les réseaux plus efficaces, plus fiables, plus sûrs et plus écologiques aussi bien du point de vue de l'industrie automobile que des opérateurs de réseaux et de services. Les réseaux véhiculaires sont en effet une extension des réseaux MANET, permettant des échanges de données véhicules to véhicules ou encore véhicules et infrastructure. Ils visent l'amélioration de la sécurité et l'efficacité du transport routier ainsi que l'amélioration du confort de l'utilisateur en offrant différents services tels que l'accès à Internet, l'aide à la décision et le guidage.

Notre objectif a été d'analyser des mécanismes et des protocoles de communication adaptés aux caractéristiques de ces réseaux de véhicules et aux applications ciblées.

Dans un premier temps, nous avons présenté les principaux concepts, spécificités et challenges liés aux réseaux de véhicules. Nous avons également présenté la dynamique autour de ce type de réseaux mais surtout d'assimiler les bases nécessaires qui permettent de comprendre le fonctionnement et les mécanismes de bases des réseaux véhiculaires.

D'après les résultats qu'on a obtenus, on a déduit que le routage d'information joue un rôle très important dans l'étude des performances d'un réseau Véhicules. C'est pourquoi, il est nécessaire de choisir un protocole de routage pour que les résultats de la simulation soient fiables.

Notre domaine d'étude est concentré à l'étude des performances des protocoles de routage (AODV, DSDV et DSR) dans un réseau VANET. Pour atteindre notre objectif et obtenir des résultats de simulation proches de la réalité, nous avons testé dans ce réseau le protocole AODV, DSDV et DSR.

Bibliographie

- [1] A. Bachir, A. Benslimane, «A multicast protocol in ad hoc networks inter-vehicle geocast», Vehicular Technology Conference, 2003. The 57th IEEE Semiannual, Volume: 4, On page(s): 2456- 2460.
- [2] James Bernsen, and D. Manivannan, « Unicast Routing Protocols for Vehicular Ad Hoc Networks: A Critical Comparison and Classification», Elsevier Journal of Parvasive and Mobile Computing, vol. 5, pp. 1-18, 2009.
- [3] Ken Leonard. « Connected Vehicle Pilots Coming to a Region Near Yo », 2014. <http://www.transportation.gov/fastlane/connected-vehicle-pilots-coming-regionnear-you>.
- [4] J.-P. Hubaux. « The Security and Privacy of Smart Vehicles ». Presentation at ZISC Information Security Colloquium, 2004.
- [5] Saif Al-Sultan n, Moath M. Al-Doori, Ali H. Al-Bayatti, Hussien Zedan. « A comprehensive survey on vehicular Ad Hoc network», Journal of Network and Computer Applications. Article history Received 20 June 2012 Received in revised form 21 February 2013.
- [6] C2C-CC Manifesto, « Overview of the C2C-CC System version1.1 », 28th August 2007 by the technical committee.
- [7] Nouredine CHAIB. « La sécurité des communications dans les réseaux VANET ». Mémoire présenté en vue de l'obtention du diplôme Magister en Informatique. Université ELHADJ LAKHDER - BATNA.
- [8] M. Burmester, E. Magkos, and V. Chrissikopoulos. « Strengthening Privacy Protection in VANETs », in IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WIMOB), Avignon, 2008.
- [9] J. Choi, S. Jung, Y. Kim, and M. Yoo, « A Fast and efficient Handover Authentication Achieving Conditional Privacy in V2I Networks », in Proceedings of the 9th International Conference on Smart Spaces and Next Generation Wired/Wireless Networking and Second Conference on Smart Spaces, St. Petersburg, Russia, 2009.
- [10] Q. Xu and D. Jiang, « Design and analysis of highway safety communication protocol in 5.9 GHz dedicated short range communication spectrum », Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual, vol. 4, pp. 2451-2455, Apr. 2003.
- [11] J. Santa, A. F. Gómez-Skarmeta, and M. Sánchez-Artigas, « Architecture and Evaluation of a Unified V2V and V2I Communication System based on Cellular Networks », Computer Communications, vol. 31, no. 12, pp. 2850-2861, Jul. 2008.

- [12] Meraihi yacine. « Routage dans les Réseaux Véhiculaire (VANETs) cas d'un environnement type Ville », Département : Automatisation et Electrification des Procédés Industriels, Université M'HAMED BOUGARA - BOUMERDES. 2011.
- [13] M. JERBI, « Protocoles pour les Communications dans les Réseaux de Véhicules en environnement Urbain : Routage et GeoCast basés sur les intersections », Université D'EVRY VAL D'ESSONNE, thèse de doctorat, 2008.
- [14] M. Fiore, J. Harri, F. Filali, and C. Bonnet, « Vehicular Mobility Simulation for VANETs », in Proceedings of the 40th Annual Simulation Symposium, Norfolk, VA, 2007.
- [15] S. N. Pathak and U. Shrawankar, « Secured Communication in Real Time VANET », in Proceedings of the 2009 Second International Conference on Emerging Trends in Engineering & Technology, Nagpur, 2009, pp. 1151-1155.
- [16] Mostafa M. I. Taha B.Sc. « Electrical Engineering, Broadcasting Protocols in Vehicular Ad-Hoc Networks (VANETs)», Assiut University, 2004.
- [17] Ait Ali Kahina. « Modélisation et Etude de Performances dans Les Réseaux VANET ». Thèse de doctorat de l'Université de Technologie de Belfort-Montbéliard. 2012.
- [18] Tayeb Lemlouma, « Le routage dans les réseaux mobiles ad hoc »
[www.opera.fr/people/TayebLemlouma/papers/AdHoc_presentation .PDF](http://www.opera.fr/people/TayebLemlouma/papers/AdHoc_presentation.PDF)
- [19] Amina Naimi-Meraihi, « Délai de routage dans les réseaux Ad Hoc 802.1 », Thèse doctorat. INRIA Rocquencourt, 2005.
- [20] M. Gerla, X. Hong, G. Pei, « Fisheye State Routing Protocol (FSR) », IETF Internet Draft, work in progress, draft ietfmanet- fsr-03.txt, July 2002.
- [21] T. Clausen et al. « Optimized Link State Routing Protocol (OLSR) », RFC 3626, Network Working Group, 2003.
- [22] Catherine Loison, Thomas Ruocco, Camille Rives, « Routage Multicast dans les Réseaux Véhiculaires (VANET) ». Université d'Avignon, 2013.
- [23] Z. J. Haas, « The Zone Routing Protocol (ZRP) for ad hoc networks », Internet Draft, 1997.
- [24] Navid Nikaein, Christian Bonnet and Neda Nikaein, « HARP - Hybrid Ad Hoc Routing Protocol », in proceeding of IST 2001: International Symposium on Telecommunications, Iran/Tehran 2001.
- [25] AOUES Abdelaziz, HAMMOUDI Meryam, BENAÏSSA Yousef, BENSALDANE Nadjet. « Les Réseaux Véhiculaires VANET », Département d'informatique L3 GTR Technologie Réseau 2014/2015, Université des sciences et de technologie HOUARI BOUMEDIENE.

- [26] Yang Xia, Chai Kiat Yeo, Bu Sung Lee, « Hierarchical Cluster Based Routing for Highly Mobile Heterogeneous MANET », Communications and Information Technologies, ISCIT'07, International Symposium on, pp. 936–941, Oct. 2007.
- [27] Sorin Paun Laurentiu « Gestion de la mobilité dans les réseaux ambiants ». Thèse de doctorat, Institut National Polytechnique de Grenoble 2005.
- [28] Elizabeth M. Belding-Royer. Hierarchical Routing in Ad Hoc Mobile Networks. Wireless Communications and Mobile Computing, 2(5) :515–32, 2002.
- [29] Chun-hung Richard Lin and Mario Gerla. « A Distributed Control Scheme in Multi-hop Packet Radio Networks for Voice/Data Traffic Support », In Proceedings of the 1995 IEEE International Conference on Communications (ICC'95), volume 2, pages 1238–1242, Seattle, WA, USA, June 18-22 1995. ISBN 0-7803-2486-2.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=524297.
- [30] Chunhung Richard Lin and Mario Gerla. « Adaptive Clustering for Mobile Wireless Networks », IEEE Journal on Selected Areas in Communications, 15(7) :1265–75, 1997.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=622910.
- [31] U.C. Kozat and al. Virtual dynamic backbone for mobile ad hoc networks. IEEE ICC'01, 1:1201–10, 2001.
- [32] Marc R. Pearlman and Zygmunt J. Haas. « Determining the Optimal Configuration for the Zone Routing Protocol », IEEE Journal on Selected Areas in Communications, 17 :1395–1414, 1999. ISSN 0733-8716.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=779922
- [33] Alan Amis, Ravi Prakash, Dung Huynh, and Thai Vuong. « Max-Min D-Cluster Formation in Wireless Ad Hoc Networks », In Proceedings of the 2000 IEEE Computer and Communications Societies Conference on Computer Communications (INFOCOM'00), volume 1, pages 32–41, Los Alamitos, California, USA, 26–30 March 2000. IEEE.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=832171.
- [34] Badreddine Guizani. « Algorithmes de Clusterisation et Routage dans les Réseaux Ad Hoc », Computers and Society. Université de Technologie de Belfort-Montbéliard; Ecole Nationale des Sciences de l'Informatique de Manouba, 2012. French.
- [35] Anthony Ephremides, Jeffrey E. Wieselthier, and D.J. Baker. « A Design Concept for Reliable Mobile Radio Networks With Frequency Hopping Signaling ». Proceedings of the IEEE, 75(1) :56 – 73, January 1987. ISSN 0018-9219.
- [36] Ching Chuan Chiang, Hsiao-Kuang Wu, Winston Liu, and Mario Gerla. « Routing In Clustered Multihop, Mobile Wireless Networks With Fading Channel », In Proceedings of the IEEE Singapore International Conference on Networks (SICON'97), pages 197–212, Singapore, August 19 1997. IEEE.
URL <http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.50.8359>.

- [37] Mario Gerla and Jack Tzu-Chieh Tsai. « Multicluster, Mobile, Multimedia Radio Network », *Wireless Networks*, 1(3) :255–265, 1995.
URL <http://dx.doi.org/10.1007/BF01200845>.
- [38] P. Basu, N. Khan, and T.D.C. Little. « A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks », In *Proceedings of the International Conference on Distributed Computing Systems Workshop (ICDCSW'01)*, pages 413–418, Mesa, Arizona, USA, April 2001. IEEE. ISBN 0-7695-1080-9.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=918738.
- [39] Oussama Younes and Sonia Fahmy. « HEED : A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks », *IEEE Transactions on Mobile Computing*, 3 :366–379, October 2004. ISSN 1536-1233.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1347100.
- [40] Inn ER and Winston K.G. Seah. « Mobility-based d-Hop Clustering Algorithm for Mobile Ad Hoc Networks », In *Proceedings of the 2004 IEEE Wireless Communications and Networking Conference (WCNC'04)*, volume 4, pages 2359–2364, Atlanta, Georgia, USA, March 21-25 2004. IEEE.
- [41] Dzmitry Kliazovich, Pascal Bouvry, Yury Audzevich, and Samee Ullah Kha. « GREENCLOUD: A PACKET-LEVEL SIMULATOR OF ENERGY-AWARE CLOUD COMPUTING DATA CENTERS », Université de Luxembourg. 2010.
- [42] Mariam Dawoud. « Analyse du protocole AODV ». DEA d'Informatique .Université de Paul Sabatier-I.R.I.T. 2006.
- [43] Tutorial de NS disponible sur : <http://titan.cs.uni-bonn.de/~greis/ns/>.
<http://titan.cs.uni-bonn.de/~greis/ns/nstutorial.tar.gz>.
- [44] Kevin Fil Kannan, «The ns manual ». URL <http://www.isi.edu/nsnam/ns/ns-documentation>.
- [45] TALAI. « Meriem Etude et implémentation d'algorithme de mobilité de groupe et application au P-learnings », Université BADJI MOKHTAR- ANNABA. Faculté Des Sciences de L'ingénieur Département D'informatique. 2010.

Liste des Abréviations

AODV	<i>Ad-hoc On Demand Distanc Vector</i>
A-STAR	<i>Anchor Based Street and Traffic Aware Routing</i>
CDS	<i>Connected Dominating Set</i>
C.C	<i>Communication Consortium</i>
DS	<i>Dominating Set</i>
DSR	<i>Dynamic Source Routing</i>
DSDV	<i>Destination Sequence Distance Vector</i>
DVCOST	<i>Distributed Vehicle Broadcast Protocol</i>
FSR	<i>Fisheye State Routing</i>
GloMoSim	<i>Global Mobite Simulator</i>
GLS	<i>Grid Localisation Service</i>
GPS	<i>Global Positioning System</i>
GSR	<i>Geographic Source Routing</i>
GY-STAR	<i>Greedy Traffic- Aware Routing Protocol</i>
HCB	<i>Hard Core Bitch</i>
IaaS	<i>Infrastructure as a Service</i>
IS	<i>Independent Set</i>
ITS	<i>Intelligent Transport System</i>
MANET	<i>Mobile Ad Hoc Network</i>
MPR	<i>Multipoint Protocol Relai</i>
NAM	<i>Network Animator</i>
NS2	<i>Network Simulator 2</i>

<i>OBU</i>	<i>On Board Unit</i>
<i>OLSR</i>	<i>Optimized Link State Routing Protocol</i>
<i>OPT</i>	<i>Open Network</i>
<i>OTCL</i>	<i>Object Tools Command Langage</i>
<i>PaaS</i>	<i>Plateform as a Service</i>
<i>PDA</i>	<i>Personal Digital Assistant</i>
<i>RSU</i>	<i>Road Side Unit</i>
<i>RCP</i>	<i>Resource Command Processor</i>
<i>SaaS</i>	<i>Software as a Service</i>
<i>TCL</i>	<i>Tools Command Langage</i>
<i>Tk</i>	<i>Tool Kit</i>
<i>UA</i>	<i>Unit Application</i>
<i>VANET</i>	<i>Vehicle Ad hoc Network</i>
<i>VLOUD</i>	<i>Vanet Cloud Computing</i>
<i>V2I</i>	<i>Vehicle To Infrastructur</i>
<i>V2V</i>	<i>Vehicle To Vehicle</i>
<i>WDS</i>	<i>Weackley Dominating Set</i>
<i>WIMAX</i>	<i>Worldwide Interoperablity For Microwave Access</i>
<i>ZRP</i>	<i>Zone Routing Protocol</i>

Liste des figures

Figure I.1 : Exemple d'un réseau VANET.....	2
Figure I. 2 : Hiérarchie des réseaux sans fil	2
Figure I.3 : Les éléments constituant le véhicule intelligent	3
Figure I.4 : Architecture de VANET	5
Figure I.5 : Communication véhicule à véhicule	7
Figure I.6 : Communication véhicule à station de base	7
Figure II.1 : Exemple d'une structure d'un cluster	20
Figure II.2 : Classification des algorithmes de clusterisation	23
Figure II.3 : Exemple de structures de l'ensemble des cluster-heads	26
Figure II.4 : Architecture Two-tier.....	40
Figure II.5 : Architecture three-tier.....	40
Figure III.1 : Les zones urbaines pris en compte dans chaque scénario	47
Figure III.2 : Résultats de simulation de la bande passante pour le protocole AODV	49
Figure III.3 : Résultats de simulation de la bande passante pour le protocole DSDV	50
Figure III.4 : Résultats de simulation de la bande passante pour le protocole DSR	51
Figure III.5 : Résultats de simulation de la bande passante pour le protocole AODV	52
Figure III.6 : Résultats de simulation de la bande passante pour le protocole DSDV	53
Figure III.7 : Résultats de la simulation de la bande passante pour le protocole DSR	54
Figure III.8 : Résultats de simulation de la bande passante pour le protocole AODV	55
Figure III.9 : Résultats de simulation de la bande passante pour le protocole DSDV	56
Figure III.10 : Résultats de simulation de la bande passante pour le protocole DSR	57
Figure III.11 : Résultats de simulation de la bande passante dans la zone 1	58
Figure III.12 : Résultats de simulation du délai dans la zone 1	59
Figure III.13 : Résultats de simulation de la bande passante dans la zone 2	60
Figure III.14 : Résultats de simulation du délai dans la zone 2	61
Figure III.15 : Résultats de simulation de la bande passante dans la zone 3.....	63
Figure III.16 : Résultats de simulation du délai dans la zone 3.....	63

Liste des tableaux

Tableau III.1 : Paramètres des zones	47
Tableau III.2 : Tableau récapitulatif des paramètres de simulation	48
Tableau III.3 : La valeur moyenne de la bande passante pour AODV	49
Tableau III.4 : La valeur moyenne de la bande passante pour DSDV	50
Tableau III.5 : La valeur moyenne de la bande passante pour DSR	51
Tableau III.6 : La valeur moyenne de la bande passante pour AODV	52
Tableau III.7 : La valeur moyenne de la bande passante pour DSDV	53
Tableau III.8 : La valeur moyenne de la bande passante pour DSR	54
Tableau III.9 : La valeur moyenne de la bande passante pour AODV	55
Tableau III.10 : La valeur moyenne de la bande passante pour DSDV.....	56
Tableau III.11 : La valeur moyenne de la bande passante pour DSR	57
Tableau III.12 : La valeur moyenne de la bande passante dans la zone 1	58
Tableau III.13 : La valeur moyenne du délai dans la zone 1	59
Tableau III.14 : La valeur moyenne de la bande passante dans la zone 2	60
Tableau III.15 : La valeur moyenne du délai dans la zone 2	61
Tableau III.16 : La valeur moyenne de la bande passante dans la zone 3	62
Tableau III.17 : La valeur moyenne du délai dans la zone 3	63

Résumé

La rapide progression des technologies sans fil au cours de ces dernières années a vu naître de nouveaux systèmes de communication dont les réseaux véhiculaires. Ces réseaux visent à intégrer les nouvelles technologies de l'information et de la communication dans le domaine automobile en vue d'améliorer la sécurité et le confort sur le réseau routier. Le routage constitue un élément crucial dans les réseaux VANETs, car définissant la façon dont les différentes entités échangent des messages. Le but de notre travail est d'étudier les protocoles de routages (AODV, DSDV et DSR) et leurs performances en utilisant le simulateur NS2.

Mots clés: Réseaux VANET, protocoles de routage, NS2, simulation, clusterisation.

Abstract

The rapid growth of wireless technologies in recent years has seen the birth of new communication systems including vehicular networks. These networks aim to integrate new information and communications technology in the automotive industry to improve safety and comfort on the road network . The routing is a crucial element in VANETs networks because defining how the various entities exchange messages . The aim of our work is to study the routing protocols (AODV , DSR and DSDV) and their performance using the simulator NS2

Keywords: VANET, Routing Protocol, NS2, Simulation, clusterisation.

ملخص

شهد النمو السريع للتقنيات الاتصال اللاسلكي في السنوات الأخيرة ولادة أنظمة الاتصالات الجديدة بما في ذلك شبكات المركبات. وتهدف هذه الشبكات إلى دمج تكنولوجيا المعلومات والاتصالات الجديدة في صناعة السيارات لتحسين السلامة والراحة على شبكة الطرق . التوجيه هو عنصر حاسم في شبكات VANETs لتحديد كيفية مختلف الرسائل الكيانات الصرّف. والهدف من عملنا هو لدراسة بروتوكولات التوجيه (AODV ، DSR و DSDV) و أدائها باستخدام جهاز محاكاة NS2.

كلمات البحث: شبكات VANET، بروتوكول التوجيه والمحاكاة NS2

