

الجمهورية الجزائرية الديمقراطية الشعبية
RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
وزارة التعليم العالي والبحث العلمي
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE
جامعة عمار تليجي بالأغواط
UNIVERSITÉ AMMAR TELIDJI LAGHOUAT



كلية العلوم والهندسة
FACULTÉ DES SCIENCES ET SCIENCES DE L'INGÉNIEUR
DÉPARTEMENT DE GÉNIE INFORMATIQUE

MÉMOIRE DE MASTER

DOMAINE : MATHÉMATIQUES ET INFORMATIQUE (MI)

FILIÈRE : INFORMATIQUE

OPTION. : RÉSEAUX, SYSTÈMES ET APPLICATIONS RÉPARTIS (ReSar)

Présenté par :

Ouled Amar Djamel

Thème :

Impact des protocoles de routage sur les applications multimédias temps réel dans les réseaux à forte mobilité

Soutenu publiquement devant le jury composé de :

| | | |
|--------------|---------------------------|--------------------------|
| Y. Ouinten | Maître de Conférences (A) | U. Laghouat (Président) |
| M. Djoudi | Maître-assistant (A) | U. Laghouat (Examineur) |
| O.J. Lakhdar | Maître-assistant (B) | U. Laghouat (Examineur) |
| M.B. Yagoubi | Maître de Conférences (A) | U. Laghouat (Rapporteur) |
| N. Lagraa | Maître de Conférences (A) | U. Laghouat (Rapporteur) |

ANNÉE UNIVERSITAIRE 2010/2011

Remerciements

Louange à notre Seigneur "ALLAH " qui nous a dotés de la merveilleuse faculté de raisonnement. Louange à notre créateur qui nous a incités à acquérir le savoir, c'est à lui que nous adressons toutes notre gratitude en premier lieu.

Je suis profondément reconnaissant à mon encadreur Dr. Yagoubi Mohamed Bachir, et à mon co-encadreur Dr. Nasreddine Lagraa pour m'avoir soutenu tout au long de mon projet, et pour la gentillesse qu'ils m'ont toujours éprouvé, ainsi que pour la somme de leurs conseils et de leurs recommandations, qui ont été sans doute décisifs dans l'accomplissement de ce travail,

Je tiens également à exprimer l'honneur que me font les membres du jury pour avoir accepté de me prêter leur attention et évaluer mon travail.

Je remercie sincèrement tous les membres de ma famille, dont l'aide et l'encouragement m'ont permis de continuer mes études.

Je remercie aussi tou(te)s mes collègues et mes ami(e)s pour leurs supports durant tout ce travail et pour tous les bons moments passés ensembles.

Je ne peux conclure ces remerciements sans exprimer ma reconnaissance à tous ceux qui m'ont aidé et encouragé durant ce projet.

Table des matières

| | |
|--|----------|
| Introduction générale | 7 |
| 1 Les exigences des applications multimédias en temps réel | 9 |
| 1.1 Introduction | 9 |
| 1.2 Définition de la qualité de service | 10 |
| 1.2.1 Paramètres de la QOS | 10 |
| 1.2.2 L'impact de la congestion | 10 |
| 1.2.3 Les modèles de la QOS | 10 |
| 1.2.3.1 Modèle Best Effort | 10 |
| 1.2.3.2 Modèle IntServ (Integrated Services) | 11 |
| 1.2.3.3 Modèle DiffServ (Differentiated Services) | 11 |
| 1.2.3.4 Quel modèle choisir pour VANET ? | 11 |
| 1.3 Classification des trafics | 11 |
| 1.3.1 Applications élastiques | 12 |
| 1.3.2 Applications non élastiques (temps réel) | 12 |
| 1.3.2.1 Applications temps réel interactives | 13 |
| 1.3.2.2 Applications temps réel adaptatives | 13 |
| 1.3.2.3 Exigences des jeux multi-joueurs | 13 |
| 1.3.2.3.a Le jeu de tir à la première personne FPS (First person shooter) | 13 |
| 1.3.2.3.b Jeux de rôle massivement multi-joueurs MMORPG | 14 |
| 1.3.2.3.c Jeu de stratégie en temps réel RTS (Real Time Strategy Games) | 15 |
| 1.3.2.3.d Les jeux de sport | 15 |
| 1.3.2.3.e Jeux au tour par tour | 15 |
| 1.3.2.4 Les jeux en VANET | 15 |
| 1.4 Le protocole de la couche transport | 16 |
| 1.5 Conclusion | 17 |

| | | |
|-----------|--|-----------|
| 2 | L'analyse des protocoles de routage dans les réseaux VANETs | 18 |
| 2.1 | Introduction | 18 |
| 2.2 | Types de communications | 19 |
| 2.2.1 | le mode V2V | 19 |
| 2.2.2 | le mode hybride | 19 |
| 2.3 | Caractéristiques | 20 |
| 2.3.1 | L'impact de mobilité | 20 |
| 2.4 | Classification des applications (ITS) | 20 |
| 2.4.1 | Applications de sécurité | 20 |
| 2.4.2 | Applications de confort | 21 |
| 2.5 | Les protocoles à base de topologie | 21 |
| 2.5.1 | Les protocoles proactifs | 21 |
| 2.5.1.1 | Optimized Link State Routing- OLSR | 22 |
| 2.5.1.1.a | Avantages | 22 |
| 2.5.1.1.b | Inconvénients | 23 |
| 2.5.2 | Les protocoles réactifs | 24 |
| 2.5.2.1 | Ad Hoc On Demand Distance Vector Routing- AODV | 24 |
| 2.5.2.1.a | Découverte de route de l'AODV | 24 |
| 2.5.2.1.b | Table de routage de l'AODV | 26 |
| 2.5.2.1.c | Maintenance des Routes AODV | 26 |
| 2.5.2.1.d | Avantages | 26 |
| 2.5.2.1.e | Inconvénients | 27 |
| 2.5.2.1.f | AODV + QOS | 27 |
| 2.5.3 | Les protocoles hybrides | 27 |
| 2.6 | Les protocoles à base de position | 28 |
| 2.6.1 | Greedy Perimeter Stateless Routing-GPSR | 28 |
| 2.6.1.1 | Greedy Forwarding (transfert gourmand) | 28 |
| 2.6.1.2 | Perimeter Forwarding (transfert de périmètre) | 30 |
| 2.6.1.3 | Planarized Graph (graphe planaire) | 30 |
| 2.6.1.4 | Avantages | 31 |
| 2.6.1.5 | Inconvénients | 31 |
| 2.6.2 | Conclusion | 32 |
| 3 | Applications multimédias : architectures et état de l'art | 33 |
| 3.1 | Introduction | 33 |
| 3.2 | Architectures de communication | 33 |
| 3.2.1 | Client/serveur | 34 |
| 3.2.2 | Pair à pair | 34 |
| 3.2.3 | Hybride | 34 |
| 3.3 | Le streaming | 35 |

| | |
|--|-----------|
| TABLE DES MATIÈRES | 4 |
| 3.4 Broadcasting dans VANET | 38 |
| 3.5 L'auto-configuration des adresses Ip pour les applications temps réel | 39 |
| 3.6 État de l'art | 39 |
| 3.7 Conclusion | 41 |
| 4 Simulation | 42 |
| 4.1 Introduction | 42 |
| 4.2 Le simulateur utilisé | 42 |
| 4.3 Modèle de mobilité | 43 |
| 4.4 Modèle de trafic | 43 |
| 4.5 Les métriques de comparaison | 43 |
| 4.6 Configurations et paramètres de simulation | 44 |
| 4.7 Résultats | 44 |
| 4.8 Discussions | 46 |
| 4.9 Conclusion | 50 |
| Conclusion générale | 51 |
| Bibliographie | 52 |

Table des figures

| | | |
|-----|---|----|
| 1.1 | les exigences des applications temps réel en termes de délai vs débit | 12 |
| 1.2 | la sensibilité des applications multimédias à la perte versus le délai | 14 |
| 2.1 | les types de communications | 19 |
| 2.2 | Les protocoles à base de topologie | 21 |
| 2.3 | principe de noeud MPR | 23 |
| 2.4 | Propagation des paquets RREQ | 25 |
| 2.5 | y est le voisin de x le plus proche de la destination D | 29 |
| 2.6 | X est plus proche de D que ses voisins y, w | 29 |
| 2.7 | Gauche : graphe de 200 noeuds palcés dans une aire de 2000*2000 (m*m), la portée de chaque noeud est 250 m ; Centre : graphe GG ; Droite : graphe RNG | 30 |
| 2.8 | Gabriel graph | 31 |
| 2.9 | Relative Neighborhood Graph | 31 |
| 3.1 | La méthode de l'enchevêtrement | 35 |
| 3.2 | La méthode de "Forward Error Correction" | 35 |
| 3.3 | La méthode de "MDC" | 36 |
| 3.4 | L'architecture P2P dans VANET selon la méthode MDC | 37 |
| 3.5 | Broadcast selon DV-CAST | 38 |
| 3.6 | les zones de VAC avec scope=3 | 39 |
| 4.1 | graphe de délai pour le protocole AODV | 45 |
| 4.2 | graphes de délai pour les protocoles GPSR et DSDV | 46 |
| 4.3 | graphes de débit pour les protocoles GPSR, DSDV, et AODV | 47 |
| 4.4 | graphes de rejet de paquets pour les protocoles GPSR, DSDV, et AODV | 48 |
| 4.5 | graphe de gigue pour le protocole AODV | 49 |
| 4.6 | graphes de gigue pour les protocoles DSDV, et GPSR | 49 |

Liste des tableaux

| | | |
|-----|---|----|
| 1.1 | Caractéristiques des Apps multimédias en termes de paramètres de QOS. | 13 |
| 1.2 | exigences des jeux. | 16 |
| 4.1 | propriétés de simulation. | 44 |
| 4.2 | propriétés de l'environnement urbain. | 44 |
| 4.3 | Taux de livraison. | 45 |

Introduction générale

Le routage présente des grands défis dans les réseaux Ad-Hoc comparé avec le routage dans les réseaux filaires. En fait, une stratégie intelligente de routage est requise afin de supporter les paramètres de réseau (la mobilité, le nombre de noeuds, la densité du trafic, la qualité de service et la superficie du réseau).

Un réseau VANET (Vehicular Ad-Hoc Network) est un réseau Ad-Hoc où les noeuds sont des véhicules, restreint à une topologie précise. Dû à la forte mobilité, les topologies sont dynamiques et la perte de connexion est fréquente, voire la perte de données, ce qui fait de ce type de réseaux un domaine de recherche récent.

La bonne raison pour laquelle un réseau VANET est né, c'est l'amélioration de sécurité des automobilistes, mais avec le passage du temps une question s'est posée : pourquoi ne pas mettre à l'aise les voyageurs et mêmes les conducteurs ?

Afin d'y répondre donc quelques tentatives ont eu lieu pour automatiser quelques tâches comme les péages automatiques, la réservation de parking, et même le déploiement des applications de confort (y compris les applications multimédias). Ces dernières constituaient une gamme de défis qui complexifie en plus la mission des protocoles de routage dans ce type de réseaux.

Les applications multimédias prennent de l'importance dans les réseaux filaires ou sans fil, mais leurs exigences en termes des paramètres de la qualité de service (Débit, Gigue, Latence et Perte) dans un réseau comme VANET, rendent leur déploiement difficile selon un clan, et quasiment impossible selon autre [4].

Dans ce travail, nous étudions l'impact de la mobilité sur la QOS de ces applications, avant de déterminer par la simulation le protocole de routage le plus approprié.

Le reste de ce mémoire est organisé comme suit : le chapitre 1 prend en détail les exigences des applications multimédias en temps réel en termes des paramètres de la QOS. Dans le chapitre 2, les réseaux VANET sont présentés avant qu'on analyse des protocoles de routage MANET afin de conclure ceux qui sont applicables à VANET. des architectures et des protocoles récemment proposées sont abordées dans le chapitre 3.

Et finalement une simulation aura lieu dans le chapitre 4 pour déterminer le protocole de routage le plus approprié au contexte du VANET, et savoir si tel protocole peut répondre aux exigences des applications en question. Nous terminons par une conclusion résumant les différentes étapes de ce mémoire et donnant des perspectives à ce travail.

Chapitre 1

Les exigences des applications multimédias en temps réel

1.1 Introduction

Les applications multimédias prennent de l'importance dans les réseaux filaires, ou sans fil, ces applications peuvent être confrontées à des problèmes liés principalement à la mobilité des noeuds (véhicules).

Une gamme d'applications est classée sous l'intitulé multimédia, qui peut être élastique, ou non élastique. Ce dernier peut être divisé en deux sous catégories : interactive, et adaptative.

Dans ce chapitre, on va essayer d'aborder les exigences des deux sous catégories en termes des paramètres de la QOS, et on va se concentrer très fortement sur les jeux distribués, en fait les jeux sont un très bon exemple sur les applications en temps réel, vu qu'ils sont peu tolérants à tout paramètre de QOS ce qui peut être le motif de généraliser les résultats qu'ils fournissent.

On ne peut pas parler du trafic non élastique, sans parler de la qualité de service et de ses paramètres. Ça ne veut pas du tout dire que le trafic élastique n'a rien à voir avec la QOS, la perte par exemple est un point de faiblesse de ce type de trafic, mais généralement, il y a d'autres techniques qui peuvent pallier à ce problème.

1.2 Définition de la qualité de service

Le RFC 2386 définit la QOS comme un ensemble de besoins à assurer par le réseau pour le transport d'un trafic d'une source à une destination.

Elle est aussi définie par le degré de satisfaction d'un utilisateur des services fournis par un système de communication [19].

1.2.1 Paramètres de la QOS

Délai : le délai est défini comme étant le temps écoulé entre l'envoi d'un paquet par un émetteur et sa réception par un destinataire, il est appelé délai de bout en bout, délai aller simple, ou latence.

Gigue : variation de latence des paquets, c'est donc la différence entre les délais.

Débit : la ressource de transmission qu'occupe un flot, la gestion de la bande passante est un élément important pour la garantie de la qualité de service.

Perte de paquets : elle se produit lorsqu'il y a des erreurs d'intégrité, ou lorsque l'intensité du trafic sur les liens de sortie devient supérieure à leur capacité d'écoulement, et peut aussi être due à la forte mobilité dans un réseau de type VANET.

1.2.2 L'impact de la congestion

D'après [13] tous ces paramètres sont concernés par la congestion, en l'absence de la congestion, chaque flot peut utiliser le niveau de bande passante qu'il souhaite, aucun paquet n'est perdu, la latence est minimale, et la gigue est quasiment nulle.

En effet ces paramètres sont nécessaires pour définir avec exactitude la différence entre un trafic temps réel et un autre trafic.

1.2.3 Les modèles de la QOS

beaucoup de modèles ont été proposés récemment, mais généralement, ces modèles se basent sur les trois modèles les plus connus et répandus : best effort, IntServ, et DiffServ, chacun de ces modèles essaye de répondre à la question comment offrir les services au trafic.

1.2.3.1 Modèle Best Effort

C'est le modèle le plus utilisé, il est cependant utilisé dans l'internet, son principe est très simple " service au mieux ", il consiste à ne pas différencier les services, quel que soit le type de trafic.

1.2.3.2 Modèle IntServ (Integrated Services)

Les services intégrés en français, on distingue deux types de services : les services élastiques, et les services temps réel.

Dans la première catégorie, l'application attend d'avoir reçu la totalité des paquets avant de traiter les données, le délai de transit peut largement varier, le service est de type best effort.

Et puisque les services temps réel sont sensibles au délai et surtout à la gigue : Le modèle distingue les services tolérant une variation du délai de transit pour lesquels une classe de service est définie, et une autre classe pour les autres services.

1.2.3.3 Modèle DiffServ (Differentiated Services)

Le modèle DiffServ a pour but d'agréger les flots en quelques classes offrant des services spécifiques, la spécification des classes se fait par le champ TOS dans IP v4, les paquets marqués sont traités en priorité.

Dans ce modèle, en plus du best effort, on a les services garantis, et les services assurés.

1.2.3.4 Quel modèle choisir pour VANET ?

L'implémentation d'une politique de différenciation de trafic doit se faire au niveau de chaque noeud, et évidemment c'est le dernier modèle (DiffServ) qui convient le mieux aux réseaux VANET, les applications temps réel ne doivent pas être traitées de la même façon, par exemple il faut différencier les services offerts aux applications adaptatives de ceux offerts aux applications interactives.

1.3 Classification des trafics

On distingue deux grandes catégories de flots qu'on les classe selon leur élasticité aux paramètres de QOS.

Suivant le type du trafic les besoins de QOS sont différents, par exemple pour les applications non temps réel (trafic élastique) comme le transfert de fichier ou la messagerie, la fiabilité des communications est le but essentiel, alors que pour les applications temps réel (trafic non élastique) comme les jeux en ligne et la voix sur IP la latence doit être limitée sinon le paquet est inutile, et la gigue elle aussi s'il dépasse une certaine valeur peut impliquer une discontinuité au niveau de la restitution de données.

1.3.1 Applications élastiques

Les applications élastiques ont apparues très tôt sur Internet, elles représentent la majorité du trafic dans tous les réseaux.

Les exemples les plus connues sont les courriers électroniques (SMTP), la navigation sur le web (http), et le téléchargement de fichiers (FTP), toutes ces applications ne nécessitent pas une quantité spécifique de la bande passante pour fonctionner, et ne sont sensibles qu'à la perte et à la déconnexion, pour cela le protocole de la couche transport le plus approprié pour ces applications est le TCP.

1.3.2 Applications non élastiques (temps réel)

Ces applications sont de plus en plus populaires dans le monde d'Internet, il s'agit des jeux en ligne, de la voix IP (VoIP), de la visioconférence, de vidéo streaming, etc.

Peu importe si elles sont utilisées à travers une architecture traditionnelle client-serveur ou en utilisant une connexion P2P, la principale exigence de ces applications est incarnée par leur faible tolérance aux délais (CF. Figure 1.1), et principalement à la gigue.

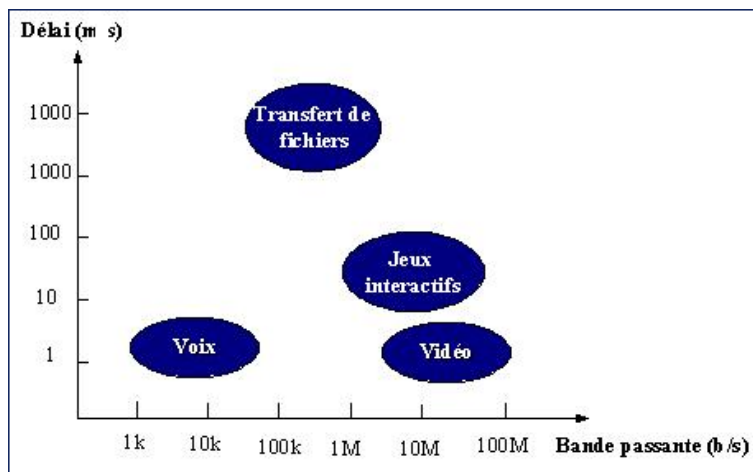


FIGURE 1.1 – les exigences des applications temps réel en termes de délai vs débit

En effet pour réduire l'impact de la gigue les paquets sont bufférisés à la destination avant d'être lus (CF. Table 1.1), pour cela la gigue doit être bornée pour prévoir la taille du tampon, qui dépend principalement de la gigue et du débit de réception de données.

Dans ce travail on s'intéresse donc à ces applications, qu'on les divise en deux catégories :

| Application | Caractéristiques du multimédia | | | |
|--------------------|--------------------------------|------------|-----------|-----------|
| | débit | Sensible à | | |
| | | Délai | Gigue | Perte |
| VoIP | Faible | Important | Important | Moyen |
| Visioconférence | Important | Important | Important | Moyen |
| Vidéo à la demande | Important | Moyen | Moyen | Moyen |
| jeux | Faible | Important | Important | Important |

TABLE 1.1 – Caractéristiques des Apps multimédias en termes de paramètres de QOS.

1.3.2.1 Applications temps réel interactives

Dans ce genre d'applications, il y a une interaction entre l'utilisateur et l'application, comme par exemple : la majorité des jeux, la visioconférence, la téléphonie sur IP etc.

Ce genre d'applications est très dur, si le paquet par exemple n'arrive pas dans les délais, il est automatiquement rejeté, et il a fait une consommation inutile de la bande passante.

1.3.2.2 Applications temps réel adaptatives

La vidéo streaming, le chat, les jeux au tour par tour sont des très bons exemples sur cette catégorie, en fait l'application du principe de buffer suffit même si les données viennent en retard, elles sont beaucoup plus sensibles à la perte que les applications interactives et moins sensibles en termes de la latence.

La performance de ces applications n'est donc pas liée aux délais (ceux-ci étant suffisamment faibles), mais plutôt au partage de la bande passante.

la Figure 1.2 montre la sensibilité des applications multimédias à la perte versus le délai.

1.3.2.3 Exigences des jeux multi-joueurs

Les exigences des jeux multi-joueurs dépendent de type de jeu en question, il s'agit de :

1.3.2.3.a Le jeu de tir à la première personne FPS (First person shooter)

Le FPS est un jeu d'action rapide, où le problème de la latence est prédominant. Les meilleurs exemples de FPS sont Counter Strike, la série des Quake et Unreal Tournament.

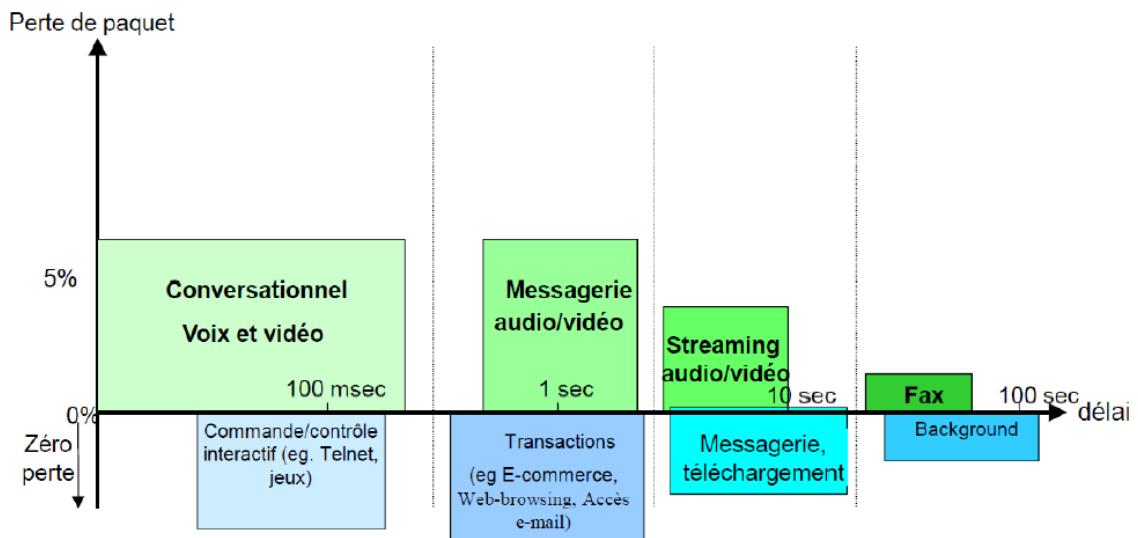


FIGURE 1.2 – la sensibilité des applications multimédias à la perte versus le délai

Le délai maximum tolérable pour ce genre de jeux est 150 ms [4, 1], et en termes de la bande passante ce n'est pas gourmand 40 kbps/utilisateur,

La durée de session est aussi un facteur essentiel puisque dans les réseaux VANET la durée de connexion entre les noeuds n'est pas longue. Pour 60% des jeux FPS, la session ne dépasse pas 15 minutes, et n'excède pas 30 minutes pour 80% [1].

La perte lui aussi ne devra jamais excéder 3% sinon ça commencera à se dégrader. La taille des paquets moyenne est également assez faible, généralement inférieure à 200 octets avec une moyenne proche de 100 octets,

Et pour tout type de jeu la gigue ne doit pas dépasser 20 ms [14].

1.3.2.3.b Jeux de rôle massivement multi-joueurs MMORPG

MMORPG en anglais (Massively Multiplayer Online Role Playing Game). La problématique principale dans ce type de jeux se situe au niveau du serveur : comment gérer des millions de joueurs simultanément ?

Les MMORPG les plus connus sont World of Warcraft (WoW) et Everquest. Ce genre de jeux peut impliquer des millions de joueurs connectés au monde virtuel simultanément. Les paquets sont envoyés en rafales et le besoin de bande passante moyenne est faible [3], aussi faible que 7kbps pour certains jeux.

Les tailles de paquets sont également petites, pour la plupart moins de 200 octets, avec une taille moyenne d'environ 100 octets.

La durée moyenne de connexion d'un utilisateur dans un jeu MMORPG avoi-

sine les 100 minutes pour certains jeux, beaucoup plus que pour un jeu FPS.

Certains MMORPG sont capables de fonctionner sans problème avec 1s de latence [4].

1.3.2.3.c Jeu de stratégie en temps réel RTS (Real Time Strategy Games)

Le joueur contrôle un grand nombre d'unités et doit détruire les bases des adversaires. La problématique des RTS vient du grand nombre d'informations à transmettre en même temps.

Quelques exemples connus sont Age of Empires, Warcraft 3. Ces jeux ont tendance à soutenir un petit nombre de joueurs dans une partie donnée, généralement moins de 8. Ils ne sont pas trop sensibles à la latence, le temps de latence est toléré jusqu'à 500ms [4, 1, 15].

Comme pour les jeux FPS, les besoins en bande passante sont à nouveau maintenus suffisamment bas pour permettre aux jeux d'être joués sur une connexion modem, mais leur durée de session est vraiment variable on peut jouer jusqu'à 8 jours dans la même partie.

1.3.2.3.d Les jeux de sport

Dans ce type de jeu, les joueurs se concentrent avant tout sur leur performance individuelle. Ces jeux ne sont plus jouables si la latence dépasse 500 ms, Les exemples les plus joués sont Need for Speed, Fifa et Pro Evolution Soccer.

Ils sont limités à peu de joueurs, en général 4 [15].

1.3.2.3.e Jeux au tour par tour

Tout jeu multi-joueurs n'étant pas forcément en temps réel, c'est le cas des jeux au tour par tour.

Les jeux au tour par tour regroupent les jeux de Poker en ligne, certains jeux de Stratégie au tour par tour (wargames), les jeux de golf, ou les jeux utilisant des pages web pour effectuer les actions. Leurs caractéristiques sont très proches de celles du trafic élastique.

1.3.2.4 Les jeux en VANET

Comme déjà dit, la durée de session est une contrainte influente dans les réseaux VANET, et surtout dans l'environnement urbain. D'après [1] ce sont les jeux FPS qui semblent les plus appropriés aux réseaux VANET, malgré qu'ils restent limités à la latence.

Finalement, la table 1.2 qui résume les exigences des jeux.

| Type de jeu | Durée moyenne de session | Latence tolérable | Bp requise/Utilisateur |
|-------------------|--------------------------|-------------------|------------------------|
| FPS | 15 mn pour 60% | <150 ms | 40kbps |
| MMORPG | Plus de 90 mn | 1000 ms | 7 kbps |
| RTS | variable | 500 ms | 40 kbps |
| Les jeux de sport | 15 minutes | <500 ms | variable |

TABLE 1.2 – exigences des jeux.

1.4 Le protocole de la couche transport

Le protocole TCP (Transport Control Protocol), est un protocole en mode connecté. C'est-à-dire que pour chaque paquet, on a un acquittement (ACK). Si le paquet n'est pas acquitté, on le renvoie.

Cette façon de faire est avantageuse lorsqu'on a besoin de transférer des données importantes. Mais lorsqu'on parle par exemple et qu'un paquet est perdu, ce n'est pas la peine de le renvoyer, il sera trop tard pour prendre en compte l'information et alors gaspillage de la bande passante.

Contrairement au protocole TCP, le UDP (User Datagram Protocol), offre un léger service : il prend juste les messages de la couche d'application et les envoie vers la destination sans aucune garantie que les messages en fait vont être livrés et dans quel ordre.

En effet, UDP ne fait ni la retransmission, ni le contrôle de flux ni le contrôle de congestion, et sa simplicité d'utilisation assure une réduction des frais généraux ; en outre, le fait qu'il n'existe aucune connexion à établir, le rend plus rapide que TCP.

Toutes ces raisons font de l'UDP l'idéal pour les applications temps réel qui ont besoin d'une quantité constante de bande passante minimale à leur disposition pour fonctionner correctement.

1.5 Conclusion

Dans ce chapitre, on a détaillé les applications multimédias en temps réel, leurs types, leurs exigences en termes des paramètres de la QOS.

Les exigences de ces applications sont très diverses, en fait, parfois on trouve une catégorie qui est peu tolérante à l'un des paramètres de la QOS, et une autre qui est très tolérante au même paramètre.

Généralement ces exigences peuvent être limitées par des seuils extrêmes, ces seuils peuvent à leur tour nous aider à simuler avec exactitude les protocoles de routage, après d'être analysés dans le chapitre suivant.

Chapitre 2

L'analyse des protocoles de routage dans les réseaux VANETs

2.1 Introduction

Les réseaux véhiculaires (VANET) ont un intérêt croissant pour leurs utilisations dans les applications de sécurité, le péage automatique, les applications multimédias. Ils ont des propriétés uniques telles que les noeuds étant capables de se déplacer à grande vitesse, la topologie dynamique, le nombre important de noeuds.

Il est donc important, d'enquêter sur les protocoles de routage pour VANET pour assurer une communication efficace. La plupart des protocoles de routage ad hoc se répartissent en deux catégories : les protocoles à base de topologie et les protocoles à base de position [1].

Les protocoles à base de topologie utilisent les informations des états des liens existants dans le réseau pour prendre des décisions de routage lors de la transmission d'un paquet à partir d'un noeud source à un noeud destination. Ils peuvent être à leur tour sous-divisés en proactifs, réactifs et hybrides.

Un protocole proactif maintient les itinéraires pour d'autres noeuds dans le réseau indépendamment du trafic réseau. Alors qu'un protocole réactif ne maintient que les routes en cours d'utilisation. Au même temps un protocole hybride essaie de combiner les deux approches, par exemple, la combinaison d'un régime local proactif, et mondial réactif.

Alors que les protocoles à base de position exigent que les noeuds qui participent, soient en mesure de déterminer leurs positions géographiques. Cela est généralement réalisé par l'utilisation de GPS.

La décision de routage est alors basée sur la position du noeud destination, et celle des voisins du noeud source. Dans ce chapitre, on va essayer de répondre

d'une manière indirecte à la question : pourquoi les protocoles basés sur la position sont plus adaptés à VANETs que ceux basés sur la topologie ?

2.2 Types de communications

Les communications véhiculaires peuvent être classées en 3 types de communications : (i) véhicule à véhicule (V2V), (ii) infrastructure à véhicule (I2V), (iii) véhicule à infrastructure (V2I). CF. 2.1, les communications inter-infrastructures ne font pas partie de VANET.

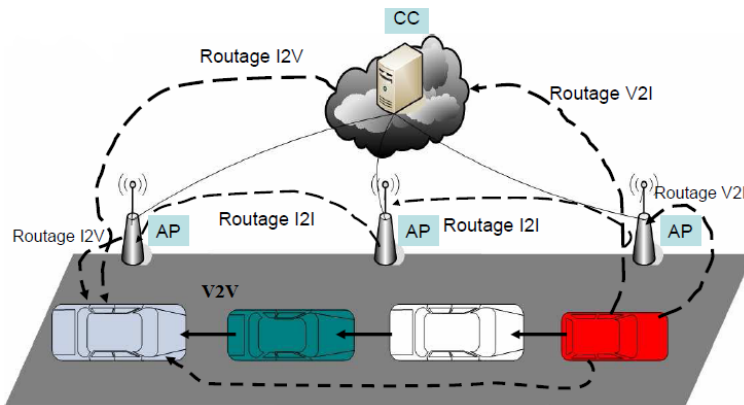


FIGURE 2.1 – les types de communications

2.2.1 véhicule à véhicule (V2V)

Dans cette architecture trois types de communications sont possibles, (i) coopérative (Broadcast) : freinage d'urgence, Signalisation des accidents, l'intervalle de communication est de 300 m à 1 km, (ii) échange (unicast) : Permet un échange de communication entre de 2 véhicules, l'intervalle de communications de 0 à 5 km, (iii) transport de données (forwarding).

2.2.2 infrastructure à véhicule (I2V) et véhicule à infrastructure (V2I)

Cette architecture, comprend la signalisation (prévention, la sécurité), et les loisirs, elle fait partie de VANET, mais il s'agit d'une architecture hybride.

2.3 Caractéristiques

Le VANET se caractérise par la forte mobilité des noeuds rendant la topologie du réseau fortement dynamique, du nombre important de noeuds, de la topologie qui est restreinte à celle des routes, et du problème de sécurité qui est vivement posé, ce sont ces facteurs qui différencient le VANET des autres réseaux mobiles MANETs.

2.3.1 L'impact de mobilité

Le fait que les noeuds soient fortement mobiles, impliquent une variation brusque de la topologie, ce qui fait que les messages de contrôle périodiques comme le message hello devraient être diffusés au bout d'un intervalle de temps très court. Ce qui augmente la congestion dans le réseau, et dégrade les performances des paramètres de QOS.

2.4 Classification des applications (ITS)

Une classification en quatre catégories des applications (ITS) est faite dans [18], contenant :

- (i) Les applications orientées infrastructures : concerne la gestion des autoroutes, transport de marchandises etc.
- (ii) Les applications orientées véhicules : concerne la conduite assistée, alertes d'accidents.
- (iii) Les applications orientées conducteurs : concerne le choix du meilleur trajet.
- (iv) Les applications orientées passagers : concerne les jeux distribués, l'accès à Internet etc...

D'autres [1, 2] classifient ces applications en deux catégories : (i) applications de confort, et (ii) applications de sécurité. Quelle que soit la classification de ces applications, leur intérêt reste une chose dont tout le monde est d'accord.

Dans ce travail nous adoptons la deuxième classification.

2.4.1 Applications de sécurité

Les applications de sécurité comme l'alerte en cas d'accident, l'évitement de congestion, prévention routière etc. peuvent être exposées à des problèmes majeurs, et nécessitent une vraie attention envers leur utilisation.

Ces applications ne sont pas l'objectif de ce mémoire, malgré leur priorité comparées aux applications de confort.

2.4.2 Applications de confort

Comprennent toute application à usage secondaire comme les applications multimédias, l'accès à Internet, Les applications en temps réel qui constituent un défi majeur, leur déploiement dans un réseau de type VANET, est une question que tous les chercheurs évitent d'y répondre de façon directe. Ces applications peuvent être confrontées à des problèmes graves liés à la mobilité des véhicules.

2.5 Les protocoles à base de topologie

Beaucoup des protocoles conçus pour MANET sont à base de topologie comme l'indique la figure 2.2. Comme déjà dit, ils utilisent l'état de lien pour envoyer l'information de source à destination, et sont divisés en trois catégories : proactive, réactive, et hybride.

Comme exemple de ces protocoles on prend OLSR qui est un protocole proactif, et le protocole réactif AODV.

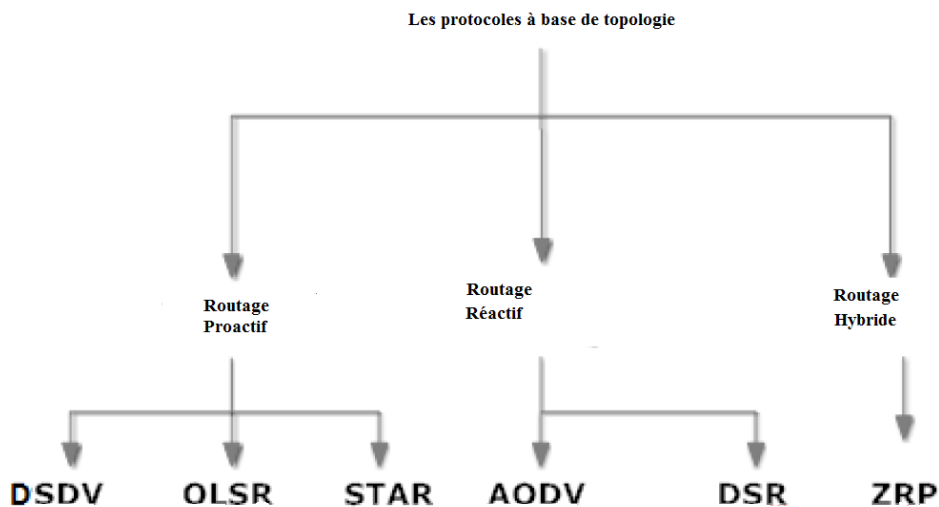


FIGURE 2.2 – Les protocoles à base de topologie

2.5.1 Les protocoles proactifs

Le principe de base des protocoles proactifs est de maintenir à jour les tables de routage, de sorte que lorsqu'une application désire envoyer un paquet à un autre mobile, une route soit immédiatement disponible. Dans le contexte des réseaux

ad hoc, les noeuds peuvent apparaître ou disparaître de manière aléatoire et la topologie même du réseau peut changer ; cela signifie qu'il va falloir un échange continu d'informations pour que chaque noeud ait une image à jour du réseau [17].

La question qui se pose c'est : est-ce que cette maintenance à jour est positive avec la variation brusque de topologie dans des réseaux à forte mobilité comme VANET, qui nous assure que la route maintenue ne serait pas changée au bout de certain temps qui est moins que le temps de mise à jour !

L'avantage de ce type de protocole c'est évidemment l'obtention immédiate de route voulue, mais ça aura vraiment d'effet sur la bande passante. Ces protocoles ne sont pas du tout applicables en VANET, à cause de la forte mobilité, en fait ils sont gourmands en termes de bande passante et nécessitent un espace de stockage très important, pour sauvegarder les tables de routage, ce qui les rend des protocoles inappropriés au contexte du VANET [2].

Le protocole OLSR est un bon exemple sur ce type de protocole. Dans ce qui suit on va essayer d'expliquer ses principes, et de mettre en évidence ses avantages et ses inconvénients.

2.5.1.1 Optimized Link State Routing- OLSR

OLSR introduit l'idée de relais multipoint pour relier les noeuds afin de réduire la surcharge de l'inondation des informations. Bien que tous les noeuds voisins reçoivent des informations envoyées par un autre noeud, seuls les relais multipoints sont autorisés à retransmettre cette information. Les relais multipoints d'un noeud A sont ses voisins directs de telle sorte que tous les voisins à deux sauts de A sont un voisin direct d'au moins un relais multipoint de A. Chaque noeud diffuse périodiquement sa liste de voisins à un saut pour pouvoir sélectionner ses noeuds relais multipoints. La figure 2.3 illustre ce principe, où les noeuds marqués comme MPR sont les relais multipoint du noeud A. Une stratégie possible pour la sélection des relais multipoint consiste à sélectionner les voisins directs qui ont le plus haut degré de voisins à deux sauts.

2.5.1.1.a Avantages

- L'utilisation du principe des noeuds MPRs réduit l'utilisation de la bande passante comparativement aux autres protocoles proactifs.
- La couverture totale du réseau est assurée tout en limitant sensiblement le nombre de réémissions.
- Puisqu'il n'y a pas une opération spécifique à appliquer pour trouver un chemin de source à destination, la latence est considérablement faible ce qui est avantageux pour les applications multimédias [1].

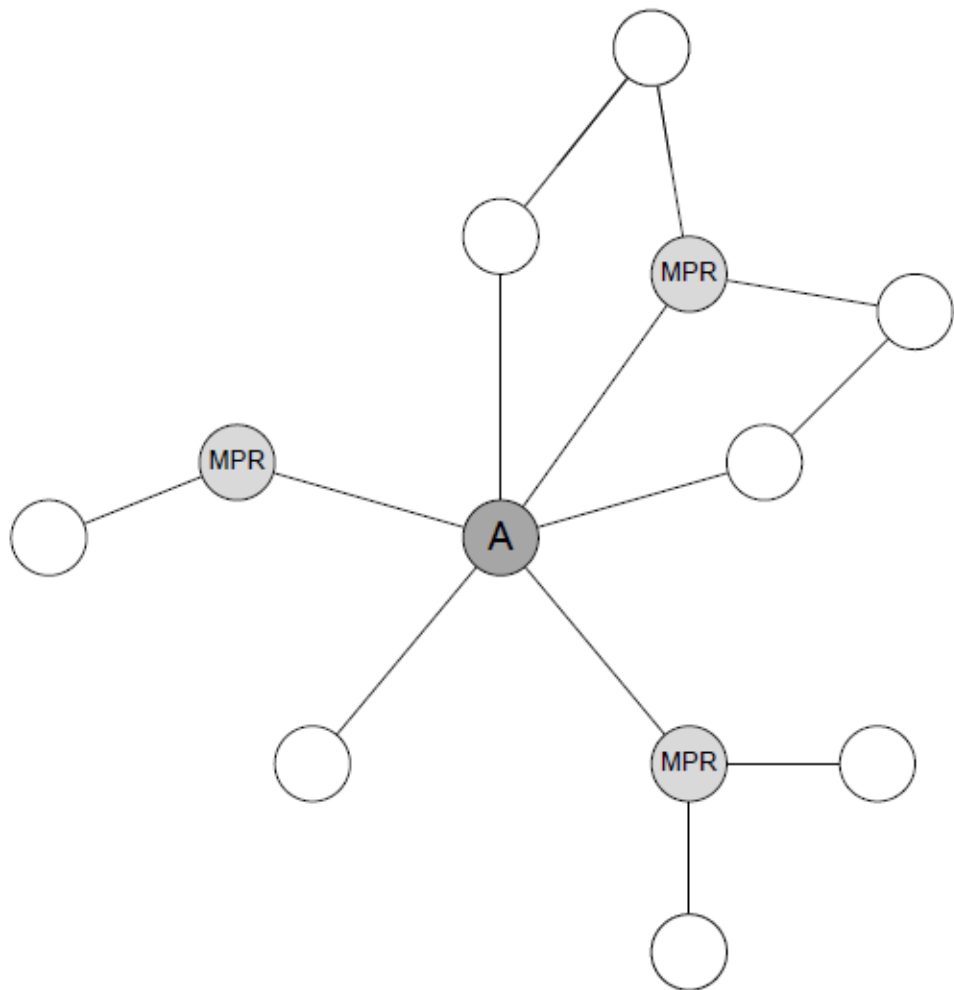


FIGURE 2.3 – principe de noeud MPR

2.5.1.1.b Inconvénients

- La bande passante et l'espace de stockage sont utilisés pour maintenir les routes à tous les noeuds qui peuvent inclure des liaisons non utilisées ou inactives [17].
- Si le réseau est peu dense, et tous les noeuds sont des relais multipoints, alors le protocole se réduit à un protocole à état de lien standard [1].

2.5.2 Les protocoles réactifs

Le principe des protocoles réactifs est de ne rien faire tant qu'une application ne demande pas explicitement d'envoyer un paquet vers un noeud distant. Cela permet d'économiser de la bande passante et de l'énergie. Lorsqu'un paquet doit être envoyé, le protocole de routage va rechercher un chemin jusqu'à la destination. Une fois ce chemin trouvé, il l'inscrit d'abord dans la table de routage avant de l'utiliser.

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). L'avantage majeur de cette méthode est qu'elle ne génère du trafic de contrôle que lorsqu'il est nécessaire. Les principales contreparties sont que l'inondation est un mécanisme coûteux qui va faire intervenir tous les noeuds du réseau en très peu de temps et qu'il va y avoir un délai à l'établissement des routes [17].

Les protocoles réactifs se répartissent en des protocoles de routage de source (source routing), et des protocoles de routage noeud-par-noeud.

Dans le routage de source, l'émetteur précise dans l'en-tête de chaque paquet la liste des noeuds qu'il devra traverser pour atteindre sa destination [17, 2], le protocole DSR est un bon exemple de cette catégorie.

Le principal inconvénient de routage de source, c'est qu'il peut ne pas convenir aux réseaux à grande échelle, là où le nombre de noeuds est très élevé, et leur comportement est très dynamique, comme VANET [2].

Dans le routage noeud-par-noeud, chaque paquet ne contient que le noeud suivant, ce qui peut pallier au problème du changement brusque de topologie. Ce type de routage est meilleur que le routage de source [2], un des protocoles qui utilisent ce type de routage est le protocole AODV [13].

2.5.2.1 Ad Hoc On Demand Distance Vector Routing- AODV

AODV représente essentiellement une amélioration de l'algorithme proactif DSDV. Le protocole AODV, réduit le nombre de diffusions de messages, et cela en créant les routes lors du besoin, contrairement au DSDV, qui maintient la totalité des routes. L'AODV est basé sur l'utilisation des deux mécanismes " Découverte de route " et " Maintenance de route ", en plus du routage noeud-par-noeud, le principe des numéros de séquence et l'échange périodique du DSDV.

2.5.2.1.a Découverte de route de l'AODV

La Découverte de route est l'une des caractéristiques les plus importantes de tout protocole de communication sans fil. La nécessité d'une découverte de route de base se pose quand un noeud source veut communiquer avec un noeud destination particulier. AODV découvre une route en diffusant RREQ à tous ses noeuds

voisins (CF. figure 2.4). Le RREQ diffusé contient l'adresse du noeud source et celle du noeud destination, afin d'identifier les noeuds en question.

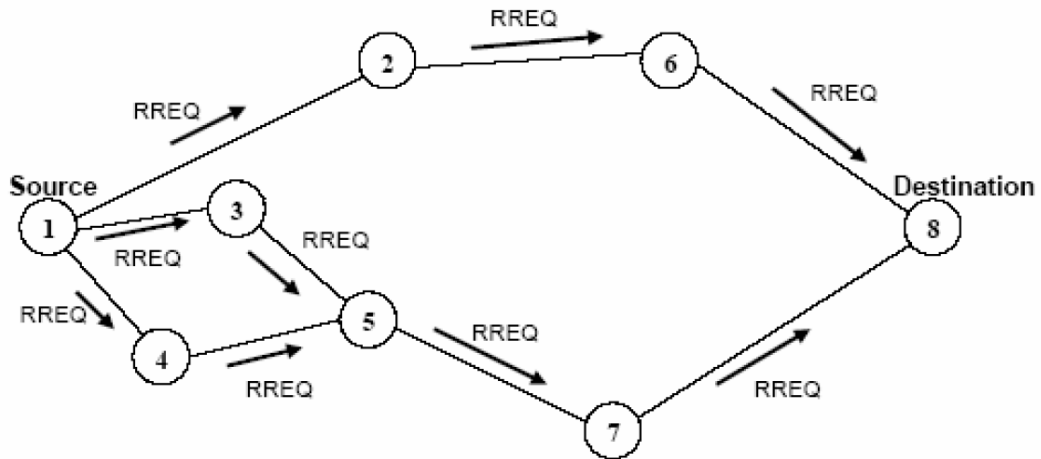


FIGURE 2.4 – Propagation des paquets RREQ

L'AODV utilise aussi les principes des numéros de séquence afin de maintenir la consistance des informations de routage. A cause de la mobilité des noeuds dans les réseaux VANET, les routes changent fréquemment ce qui fait que les routes maintenues par certains noeuds, deviennent invalides. Les numéros de séquence permettent d'utiliser les routes les plus nouvelles ou autrement dit les plus fraîches (fresh routes) [2].

RREQ contient également un ID et un compteur qui compte combien de fois le RREQ a été généré à partir d'un noeud spécifique. Quand un noeud source diffuse un RREQ à ses voisins, qui retransmettent à leur tour le RREQ à leurs voisins en incrémentant le compteur, Si un tel noeud reçoit des demandes de route multiples de la même source ID, les demandes répétées seront alors rejetées pour ne pas faire une boucle.

Afin de limiter le coût dans le réseau, AODV propose d'étendre la recherche progressivement. Initialement, la requête est diffusée à un nombre de sauts limité. Si la source ne reçoit aucune réponse (RREP) après un délai d'attente déterminé, elle retransmet un autre message de recherche en augmentant le nombre de sauts maximum. En cas de non réponse, après un certain temps (RREP_WAIT_TIME) cette procédure est répétée un nombre maximum de fois avant de déclarer injoignable la destination [17].

Tous les noeuds intermédiaires stockent les informations des noeuds destinations en particulier le numéro de séquence, et dès qu'ils reçoivent un RREQ, ils comparent le numéro de séquence stocké avec le numéro de séquence inclus dans

le message RREQ. Si le numéro de séquence RREQ est supérieur ou égal au numéro de séquence conservé alors un RREP est envoyé au noeud source [2] et la table de routage est mise à jour [13].

Si la requête RREQ est rediffusée un certain nombre de fois (RREQ_RETRIES) sans la réception de réponse, un message d'erreur est délivré à l'application [13].

Quand un noeud de transit (intermédiaire) envoie le paquet de la requête à un voisin, il sauvegarde aussi l'identificateur du noeud à partir duquel la première copie de la requête est reçue. Cette information est utilisée pour construire le chemin inverse, qui sera traversé par le paquet RREP.

Un voisin est considéré actif, pour une destination donnée, s'il délivre au moins un paquet de données sans dépasser une certaine période (active timeout period) [17].

2.5.2.1.b Table de routage de l'AODV

La table de routage du protocole AODV contient les informations suivantes [2][17].

1. l'adresse IP de destination (Destination IP Adress),
2. le numéro de séquence pour la destination (Destination Sequence Number),
3. le nombre de sauts pour atteindre la destination (Hop Number),
4. le prochain voisin désigné pour relayer le paquet vers la destination (Next Hop),
5. la durée de vie, i.e le temps pendant lequel, la route est considérée valide (Lifetime),
6. la liste des voisins actifs (Active neighbor list), et
7. le buffer de la requête (Request buffer).

2.5.2.1.c Maintenance des Routes AODV

La détection des routes invalides se fait en envoyant périodiquement des messages HELLO (RREP avec TTL=1) aux voisins.

Lorsqu'un noeud détecte qu'une route n'est plus valide, il supprime cette route de la table de routage, et envoie cette information à ses voisins actifs.

2.5.2.1.d Avantages

- Seules les routes actives sont maintenues. Cela réduit la taille globale des tables de routage [1].
- Applicable aux réseaux à grande échelle par rapport aux protocoles de routage hybrides ou proactifs [2].

- AODV réduit l'inondation de messages dans le réseau par rapport aux protocoles de routage proactifs [1].

2.5.2.1.e Inconvénients

- Contrairement aux protocoles comme DSR qui peuvent mettre en cache plusieurs routes vers une destination unique [1], AODV ne maintient qu'un seul prochain saut par destination.
- Des routes sont annulées, si elles ne sont pas utilisées, même si la topologie ne change pas. Un noeud peut avoir à effectuer une diffusion de RREQ pour une route valide qui vient d'expirer.
- Un paquet RREQ pourrait atteindre tous les hôtes dans un réseau, et alors une consommation considérable de la bande passante.
- Le protocole de routage AODV, n'assure pas l'utilisation du meilleur chemin existant entre la source et la destination [13].

2.5.2.1.f AODV + QOS

Le protocole de routage AODV avec QOS a pour objectif de :

- Ajouter une métrique plus appropriée que la distance (nombre de sauts) : délai, bande passante.
- Faire face aux changements fréquents de la topologie due à la forte mobilité des noeuds (véhicules).

Cela consiste à ajouter les deux paramètres (délai maximum, bande passante minimale) aux paquets de contrôle : RREQ et RREP, et aussi à la table de routage + la liste des sources qui ont demandé des garanties de délai ou de bande passante [13].

Chaque mobile vérifie s'il est en mesure d'honorer le service demandé, avant de retransmettre le message. Si un noeud détecte que la QoS demandée n'est pas satisfaite, alors il envoie un message à la source ayant initié cette demande de QoS, pour l'informer.

2.5.3 Les protocoles hybrides

Le routage hybride combine les caractéristiques de routage des deux types de protocoles de routage : réactif et proactif pour faire un routage plus évolutif et efficace. Généralement les protocoles hybrides sont basés sur la zone ; cela signifie que le nombre de noeuds est divisé en différentes zones pour faire la découverte de route, chaque zone est de 3 à 4 sauts [17].

Selon [2] les protocoles de routage hybrides ne sont pas adaptés ni aux réseaux à grande échelle, ni aux réseaux où le comportement des noeuds est très

dynamique, voire les deux (cas VANET).

2.6 Les protocoles à base de position

Ces protocoles sont exclusivement créés pour les réseaux à forte mobilité, dans ce type de protocoles aucune route n'est maintenue [1], et chaque noeud doit être en mesure de déterminer sa position géographique, que ce soit par GPS, ou autre système de localisation.

Ce type de routage se base sur deux composantes : un service de localisation, et une stratégie de transfert [1, 2]. Le service de localisation est la méthode de découverte de l'emplacement physique du noeud destination pour un paquet donné, ce service est sollicité lorsqu'on ne sait pas la position du noeud destinataire.

Alors que la stratégie de transfert, est l'ensemble des algorithmes utilisés pour le transfert de données, après la détection de la position du noeud destination.

2.6.1 Greedy Perimeter Stateless Routing-GPSR

C'est un protocole de routage géographique, qui garantit un passage à l'échelle car seules les informations locales sont stockées et utilisées.

Il a été conçu pour les réseaux de type MANET et spécialement les réseaux de capteurs, son principe est très simple, il suppose que tous les noeuds se trouvent au niveau d'un même plan [12]. Du fait de la mobilité des noeuds, certains algorithmes de routage qui se basent sur la topologie du réseau, ou lance une phase de découverte de routes pour acheminer des données ne sont pas adaptés à GPSR. De ce fait, il utilise la position géographique des noeuds pour l'acheminement des paquets de données ou de contrôle.

Chaque noeud dispose d'une table de position, et si après un temps T, aucun message " beacon " n'est reçu de la part de l'un des noeuds voisins inscrits dans la table, il enlève ce dernier et suppose qu'il n'est plus dans sa portée.

L'acheminement des paquets par GPSR, se fait selon deux modes suivant la densité du réseau : le " Greedy Forwarding " et le " Perimeter Forwarding " (appelés respectivement GF et PF dans la suite)

2.6.1.1 Greedy Forwarding (transfert gourmand)

Le GF crée une route du noeud source à destination en réalisant à chaque fois un saut vers le noeud le plus proche du noeud destination (CF. figure 2.5). Cette façon de faire est avantageuse lorsque la mobilité des noeuds n'est pas assez forte [12], faute de quoi elle offre un taux de réussite près de celui des réseaux filaires.

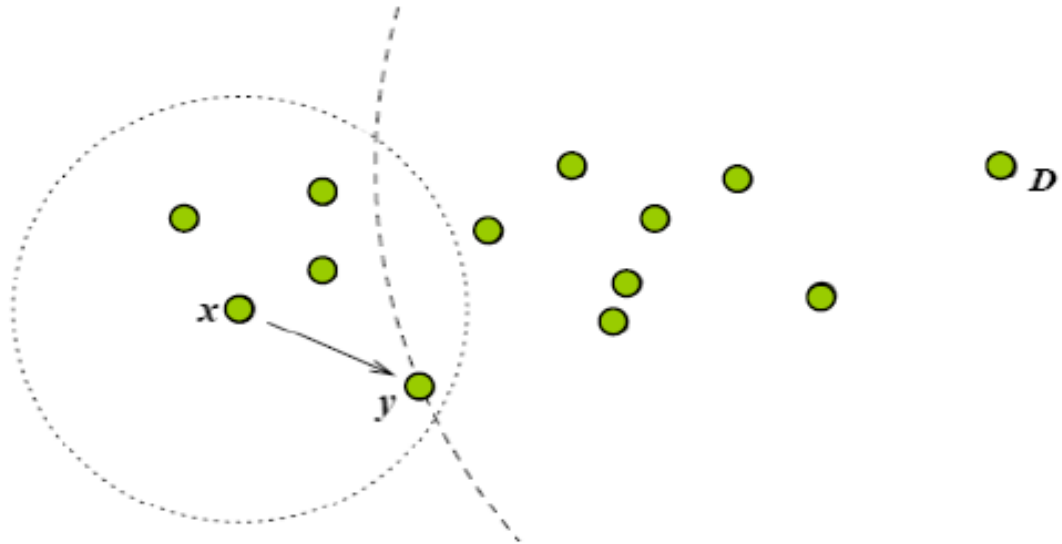


FIGURE 2.5 – y est le voisin de x le plus proche de la destination D

Ce mode est caractérisé par le fait qu'il est avantageux en termes de délai [2], s'il arrive réellement à délivrer les paquets en question, mais ceci peut ne pas être fait dans le cas où le noeud actuel est le noeud le plus proche du noeud destination mais pas forcément dans sa portée (CF. figure 2.6), dans ce cas on utilise le mode PF, et on enregistre la localisation actuelle, soit LA cette localisation.

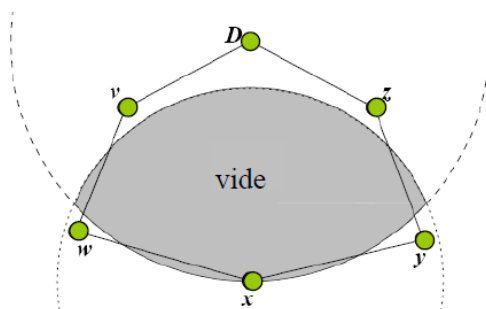


FIGURE 2.6 – X est plus proche de D que ses voisins y, w

2.6.1.2 Perimeter Forwarding (transfert de périmètre)

Ce mode utilise la règle de la main droite, c'est-à-dire que lorsqu'un nœud S envoie un paquet au nœud M, celui dernier va transmettre le paquet selon le sens inverse des aiguilles d'une montre en partant de M par rapport au segment [SM], et en évitant évidemment les nœuds déjà sollicités, afin de constituer un graphe planaire.

Un nœud recevant un paquet en mode « Perimeter » peut le transformer en mode « Greedy » et poursuivre le routage avec la méthode « greedy forwarding » si la distance entre ce nœud et la destination est inférieure à celle entre LA et la destination.

Grosso modo, l'idée c'est que je peux être le plus proche du nœud destination, lorsqu'il s'agit de mes voisins à un saut et pas forcément de ceux qui sont à deux sauts par exemple.

2.6.1.3 Planarized Graph (graphe planaire)

Ces graphes sont utilisés pour définir les liens à suivre, un nœud A peut être dans la portée d'un nœud B, mais le lien entre les deux est inutilisé si la communication est assurée par autre nœud. Ceci est fait pour éviter la congestion.

Plusieurs algorithmes existent pour rendre un graphe planaire, GPSR utilise les deux types de graphes : GG, et RNG. Chaque nœud construit un graphe du réseau lui permettant de diminuer les possibilités de routage.

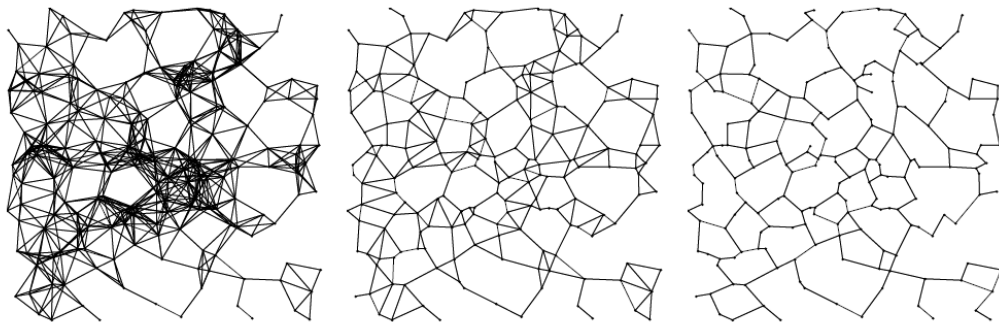


FIGURE 2.7 – Gauche : graphe de 200 nœuds placés dans une aire de 2000*2000 (m*m), la portée de chaque nœud est 250 m ; Centre : graphe GG ; Droite : graphe RNG

- Gabriel graph (GG) voir la figure 2.8, il se définit de la manière suivante : Deux nœuds u et v sont reliés par une arête si et seulement si aucun autre nœud ne se trouve dans le cercle de diamètre (u, v) [16].

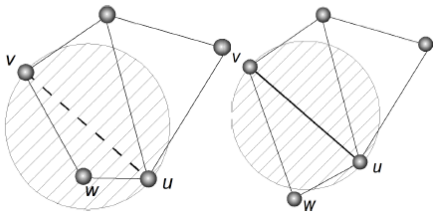


FIGURE 2.8 – Gabriel graph

- Relative Neighborhood Graph (RNG) voir la figure 2.9, c'est un graphe dont la plus longue arête entre deux noeuds voisins a été enlevée si un troisième noeud permet de conserver la connexité, On remarquera que le graphe RNG est un sous-graphe du graphe de Gabriel voir la figure 2.7.

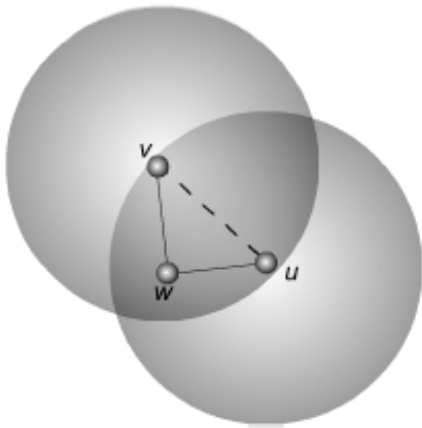


FIGURE 2.9 – Relative Neighborhood Graph

2.6.1.4 Avantages

- Adapté pour la haute mobilité.
- scalable.
- Diminue la perte de paquets grâce à la prise en compte du mouvement des noeuds intermédiaires par rapport à la destination.

2.6.1.5 Inconvénients

- besoin de GPS.
- ne fonctionne pas en cas de l'inexistence des signaux GPS.

2.6.2 Conclusion

La plupart des protocoles de routages MANET n'est pas applicable directement aux réseaux VANETs, et surtout les protocoles proactifs, tout simplement parce que ce type de protocoles trouve la route avant qu'elle soit désirée. Cette façon de faire n'est pas du tout avantageuse dans VANET à cause de la forte mobilité, une route entre deux noeuds peut être mise à jour autant de fois avant d'être demandée par telle application, ce qui est gourmand en termes de la bande passante et de l'espace de stockage.

Malgré que les protocoles réactifs semblent meilleurs que ceux proactifs en termes du débit, et de l'espace demandé, ils resteront limités. Le fait que la route ne soit pas prête à la demande peut augmenter le temps de latence, ce qui n'est pas désiré par les applications en temps réel.

Alors généralement les protocoles à base de topologie théoriquement semblent inapplicables aux réseaux VANETs, surtout lorsqu'il s'agit des applications en temps réel.

Une autre gamme de protocole a été ensuite proposée, il s'agit des protocoles à base de position, comme exemple sur cette catégorie nous avons vu le GPSR. C'est un protocole performant, mais comme tout autre protocole se basant sur position, peut être exposé à des problèmes comme par exemple lorsqu'il s'agit d'un tunnel ou d'une forêt, là où les signaux GPS ne peuvent pas pénétrer.

Mais l'affirmation de ce qu'on a dit sur ces protocoles reste sous réserve, jusqu'à ce qu'une simulation soit faite dans le dernier chapitre.

Chapitre 3

Applications multimédias : architectures et état de l'art

3.1 Introduction

Dans ce chapitre, nous allons voir, les architectures et les approches proposées pour les applications en temps réel dans VANET. L'architecture de communication est l'une des questions les plus posées par les chercheurs, est-il mieux d'utiliser une architecture centralisée comme client/serveur, ou bien puisque le réseau VANET est un réseau décentralisé, c'est le P2P qui paraît le plus approprié ?

On trouve aussi dans la littérature, pour les applications adaptatives (le cas streaming), plusieurs méthodes qui ont été proposées pour la correction d'erreur et pour le recouvrement des paquets perdus.

Dans ce contexte et vu les caractéristiques des réseaux VANET, plusieurs problèmes doivent être résolus comme le broadcasting qui nécessite une amélioration pour éviter la congestion, l'adressage qui doit minimiser le nombre de handovers dans le réseau.

Pour nous mettre là où se sont arrêtés les autres, un passage léger par l'état de l'art fait partie des éléments essentiels que prend en charge ce chapitre.

3.2 Architectures de communication

A travers tous les documents traitant le problème des applications multimédias dans les réseaux, il y a trois architectures principales : client/serveur, pair à pair (P2P), et l'architecture hybride.

3.2.1 Client/serveur

Dans une architecture client/serveur, une machine centrale joue le rôle du serveur, communiquant avec des clients. Le rôle de ce serveur est d'héberger une instance du jeu et de gérer les différents états de celui-ci. Les clients ne communiquent pas entre eux. Le serveur est le centre de toutes les communications échangées.

L'architecture client/serveur n'est pas une solution pour VANET s'il n'y a pas d'infrastructure fixe qui joue le rôle du serveur [3].

3.2.2 Pair à pair

Comme l'indique le nom, dans ce type d'architecture, les noeuds (véhicules) communiquent en direct les uns aux autres. Dans le cas de jeux cette architecture ne peut supporter qu'un nombre très restreint de joueurs, il s'agit le plus souvent des jeux de sport et des jeux de stratégie temps réel [15], mais aussi en cas des réseaux où le nombre d'utilisateur n'est pas important l'architecture P2P peut être mise en oeuvre quel que soit le type du jeu.

De plus, il y a une ressemblance entre le VANET, et l'architecture P2P : tous les deux sont décentralisés, autonomes, et fortement dynamiques. Plusieurs facteurs font de l'architecture P2P la plus appropriée pour VANET à cause de :

L'inexistence d'un noeud central : c'est-à-dire que lorsqu'un noeud se déconnecte, ce qui est probable dans un réseau où la mobilité est forte comme VANET, les autres noeuds considèrent qu'il a quitté et continuent normalement, contrairement à l'architecture client/serveur tout s'arrête brusquement.

L'effectif : généralement l'effectif dans un réseau VANET est limité par rapport à réseau Internet, selon [3] le nombre de joueurs maximum dans un réseau de type VANET est de 25 joueurs avec un débit de 6Mb/s (100kb/s par joueur), ce qui peut pallier au problème de la scalabilité de l'architecture P2P.

3.2.3 Hybride

Elle regroupe les caractéristiques des deux architectures précédentes, par exemple lorsqu'il s'agit des commandes de jeu, on utilise une architecture client/serveur et pour la voix par exemple la communication se fait directement.

Ce genre d'architecture n'est pas recommandé pour VANET pour les mêmes raisons précédemment citées dans le cas de client/serveur.

3.3 Le streaming

Ce genre d'applications fait partie de la catégorie adaptative. Suite à la perte qui peut être produite, quelques méthodes ont été proposées, comme la méthode d'enchevêtrement, la méthode FEC-Forward Error Correction, et MDC-Multiple Description Coding [11].

La méthode de l'enchevêtrement consiste à partitionner les paquets en mélangeant les unités de donnée, comme le montre la figure 3.1. Cependant cette méthode n'est pas applicable au VANET à cause de l'augmentation de la latence. La technique FEC consiste à combiner des encodages pour constituer une infor-

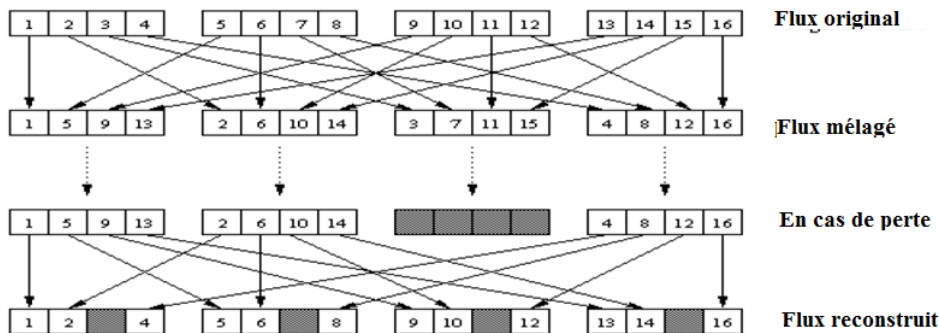


FIGURE 3.1 – La méthode de l'enchevêtrement

mation redondante de chaque paquet. La méthode de FEC est inapplicable au VANET à cause du besoin de la bande passante, et des pertes consécutives en VANET [5], (CF. Figure 3.2).

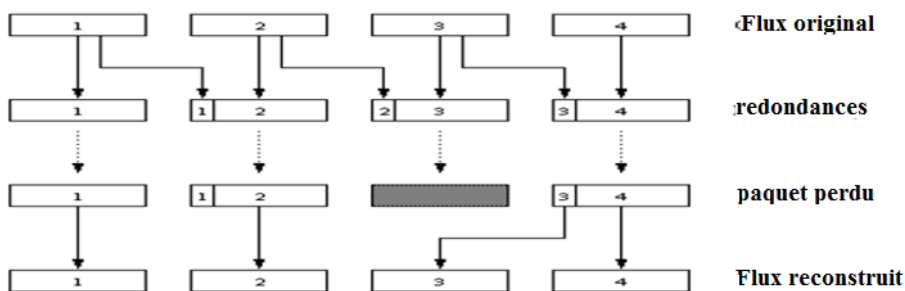


FIGURE 3.2 – La méthode de "Forward Error Correction"

Nadia et al dans [9] ont proposé une méthode se basant sur MDC [11], qui repose sur le P2P dans VANET.

Son principe est très simple, les trames I (initiales) sont suivies d'une succession de trames P (prévues), les trames B (bidirectionnelles) n'existent pas, chaque flux est décomposé en descriptions MDC1, MDC2, MDCn, selon le nombre de pairs. Chaque suite de trame est divisée en deux tranches (slice) 0 et 1. Supposons que les trames P4S1 et P6S1 de la description MDC 1 et P2S1, P4S1, P6S0 et P6S1 de la description MDC2 sont perdues, le recouvrement de ces trames se fait comme l'indique la figure 3.3.

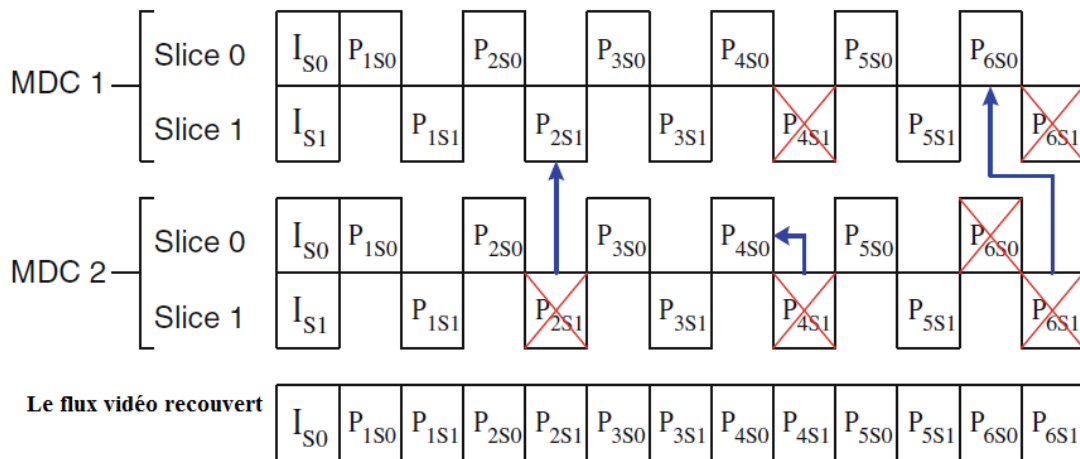


FIGURE 3.3 – La méthode de "MDC"

La figure 3.4 montre comment les noeuds procèdent en cas de perte de paquets. Ils peuvent émettre des paquets dès qu'ils les reçoivent, même si le flux est en cours.

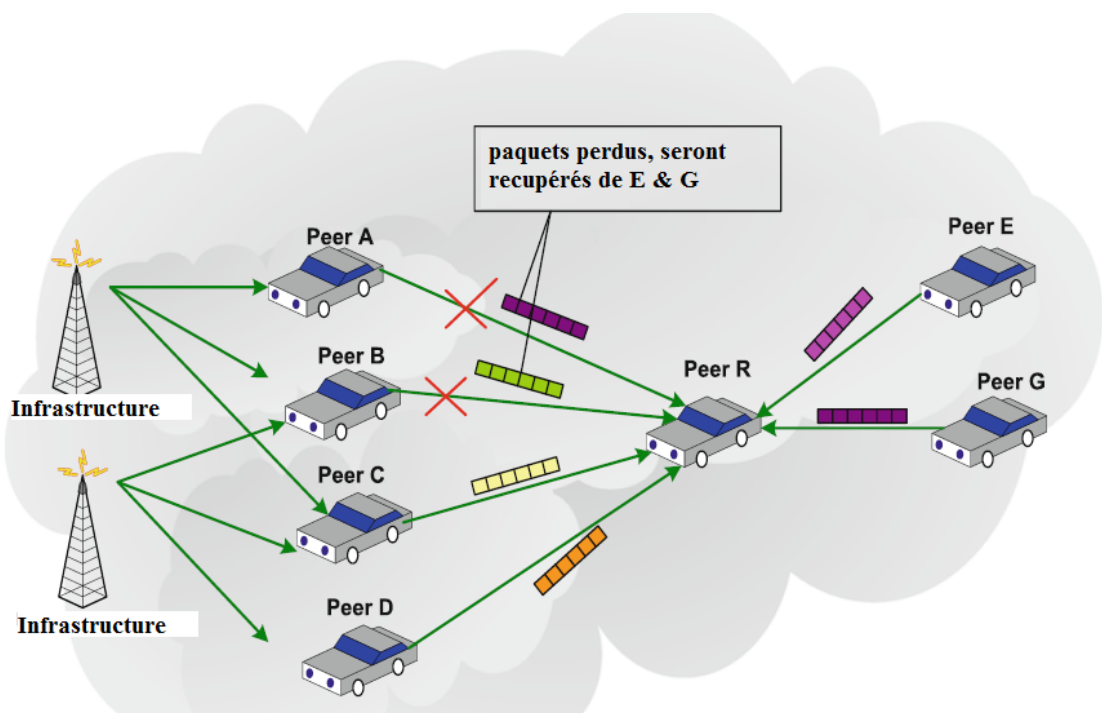


FIGURE 3.4 – L'architecture P2P dans VANET selon la méthode MDC

3.4 Broadcasting dans VANET

Un nouveau protocole distribué de diffusion a été proposé dans [6] par O. Tonguz et al. il s'agit de DV-CAST, qui est conçu spécialement pour l'amélioration du processus de diffusion (Broadcast) dans VANET.

Selon ce protocole chaque véhicule doit être en mesure de déterminer trois paramètres (i) Dflg : est-ce que le véhicule a la même direction que le véhicule source. (ii) MDC s'il est le dernier dans son groupe (cluster). (iii) ODC s'il est au moins connecté à un autre véhicule (ODN) dans le sens opposé, ces paramètres prennent les valeurs (0 ou 1).

l'organigramme de la figure 3.5 illustre le principe de fonctionnement de ce protocole.

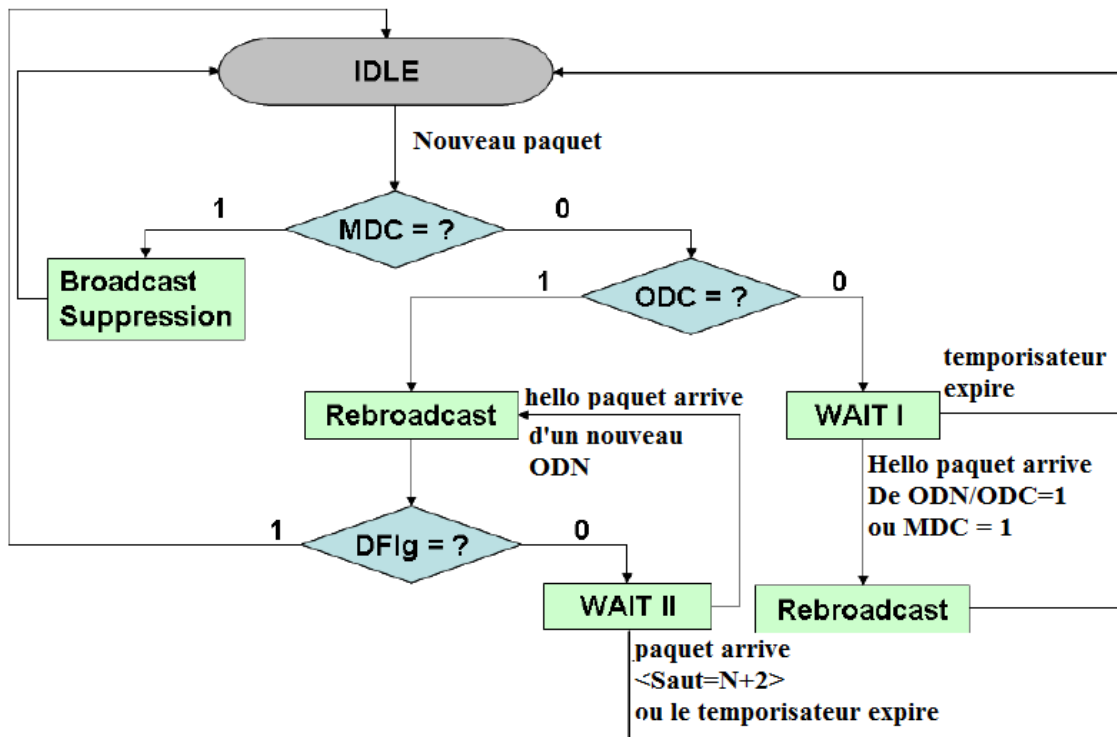


FIGURE 3.5 – Broadcast selon DV-CAST

3.5 L'auto-configuration des adresses Ip pour les applications temps réel

Pour pallier au problème de l'auto-configuration des adresses ip, M. Fazio et al. [7] ont proposé un nouveau protocole VAC (Vehicular Address Configuration protocol). L'idée c'est que le réseau des véhicules est organisé hiérarchiquement, et à chaque zone on attribue un chef de zone (leader) fixe infrastructure par exemple, qui est chargé de gérer les adresses ip.

La zone gérée par chaque chef est mesurée par un nombre de sauts appelé scope, dans la figure 3.6 scope =3. Si un véhicule sort d'une zone dont le chef est A, elle doit demander l'attribution d'une autre adresse, au chef B de la nouvelle zone.

Ce protocole est avantageux lorsque la mobilité n'est pas forte, faute de quoi le processus de handover sera fréquent. Les auteurs ont dit que ce protocole convient beaucoup plus aux jeux centralisés.

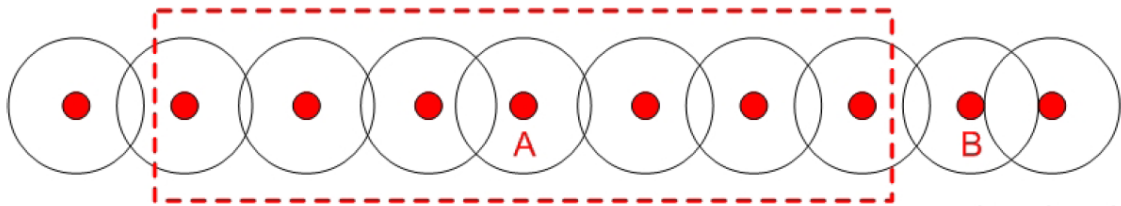


FIGURE 3.6 – les zones de VAC avec scope=3

3.6 État de l'art

Dans une étude qui a été faite sur le protocole AODV [1], afin d'en déduire s'il peut répondre aux exigences des jeux fps dans les réseaux VANET, Les auteurs ont fini par le dire que le protocole AODV ne peut répondre à ces exigences que si la stabilité des liens choisis par ce protocole est assurée. Si la route est cassée fréquemment, le protocole ne peut pas répondre à ces exigences en termes de délai et de perte. Mais, comme travail future ils voient que les protocoles à base de position peuvent répondre à ces exigences.

Dans [2]l'objectif essentiel était de mettre en évidence les problèmes auxquels peuvent être confrontés les protocoles de routage dans VANET. Selon l'auteur, les protocoles proactifs et hybrides consomment inutilement la bande passante et ne sont pas qualifiés. Tandis que les protocoles réactifs peuvent concurrencer les

protocoles à base de position, sa simulation a démontré que le GPSR est beaucoup meilleur que l'AODV dans les deux environnements : autoroute et urbain.

Démarrant de l'idée que le déploiement d'applications multimédias est difficile dans un réseau de véhicules multi-saut, les auteurs de [18] ont proposé d'utiliser de gros paquets sur des autoroutes, et de faire une grande fragmentation avec de petits paquets dans des zones urbaines (cités).

Dans [13] une modification a eu lieu sur AODV pour en faire un autre, supportant la notion de qualité de service en ajoutant des nouveaux champs dans les messages de contrôle : RREQ et RREP (modification), en comparant AODV et AODV modifié, ce dernier a prouvé ses performances.

Malgré que la plupart des articles, présumant que l'AODV est meilleur que l'OLSR, dans VANET, les auteurs dans [8] interviennent pour prouver le contraire.

L'auteur dans [20] a fait une modification au niveau de la couche MAC (802.11) avec un algorithme distribué dont le but essentiel est de minimiser la latence. Il a enlevé le mécanisme RTS/CTS, et le mode infrastructure, comparé avec les protocoles AODV, DSDV, et DSR, son protocole était performant dans des scénarios simples, mais ses performances se dégradent dans des scénarios complexes comme Manhattan.

F. Kaiser et al. [21] font une étude structurelle sur les réseaux VANETs, qui démontre que le protocole DSR est beaucoup meilleur que le GPSR en termes de taux de livraison (perte), et que les applications multimédias nécessitent des infrastructures installées au bord de la route pour qu'elles soient efficacement utilisables.

Dans [4], une étude étendue sur les jeux multi-joueurs dans les réseaux AD HOC, et le VANET où l'auteur a conclu que l'existence des infrastructures est indispensable, pour gérer le problème de l'adressage automatique.

3.7 Conclusion

Toutes ces architectures, et toutes ces approches restent limitées, et montrent leurs avantages dans des cas spécifiques et sous certaines conditions.

L'architecture P2P malgré qu'elle est qualifiée par la majorité, peut être exposée à des problèmes dans le cas d'existence des infrastructures sur la route (architecture hybride).

il s'avère aussi très claire l'inconvénient majeur du protocole VAC, qui est incarné au nombre de handovers qui peut être atteint à cause de la mobilité, ainsi que l'algorithme de diffusion qui reste améliorable en termes de ses paramètres.

Dans le chapitre suivant, et après avoir analysé les protocoles de routage et aborder leur état de l'art, on est donc prêt à entamer la phase de simulation.

Chapitre 4

Simulation

4.1 Introduction

Tout ce qu'est dit sur les protocoles de routage reste théorique, et nécessite une affirmation pratique. Souvent on trouve des contradictions dans les travaux scientifiques, un groupe qui qualifie un protocole, un autre clan le disqualifie, ce qui est dû à la diversité des conditions de simulation. C'est pour cette raison qu'il faut être rationnel dans le choix des propriétés de simulation.

Les résultats obtenus sont restreints à un environnement précis, pour un modèle de trafic précis, et offert par un simulateur précis, il est probable qu'en changeant le simulateur, d'autres résultats vont être obtenus. Toutes ces contraintes font de la simulation un outil d'approximation loin d'être stricte. Qui n'est pas alors une preuve dogmatique, mais au moins peut donner des résultats approximatifs.

Dans ce chapitre, on va mettre en évidence notre environnement logiciel, ainsi que ses propriétés, et nos paramètres de simulation, avant de discuter les résultats obtenus.

4.2 Le simulateur utilisé

NS (Network Simulator) est un outil logiciel de simulation de réseaux informatiques. Il est essentiellement élaboré avec les idées de la conception par objets, de la réutilisation du code et de modularité. Il est aujourd'hui un standard de référence en ce domaine, plusieurs laboratoires de recherche recommandent son utilisation pour tester les nouveaux protocoles.

NS2 implémente la version de 1997 de la norme 802.11. A ce titre, le débit maximum possible est de 2 Mbit/s [17]. La version qu'on a utilisé est 2.30, qui date de septembre 2006, sous Ubuntu 9.04.

4.3 Modèle de mobilité

Puisqu'il s'agit d'un réseau de véhicules, on a choisi le scénario urbain (Manhattan) téléchargé à partir de [23], les propriétés de ce scénario seront abordées un peu plus loin.

4.4 Modèle de trafic

Il est possible de générer des paquets par un générateur de trafic en spécifiant le type, la durée, la vitesse de flux du trafic dans le réseau.

Un générateur est valable sous le répertoire ns2.30/indep-utils/cmu-scen-gen.

Pour générer du trafic on procède comme suit :

```
ns cbrgen.tcl [-type cbrtcp] [-nn nodes] [-seed seed] [-mc connections] [-rate rate]
>[fichier de sortie]
```

- Type de trafic : CBR(Constant Bit Rate) ou TCP
- Seed : initialisation de variable de générateur des nombres aléatoires
- nn : nombre des noeuds
- mc : le nombre maximal des connexions
- Rate : nombre des paquets par seconde (bit rate)

4.5 Les métriques de comparaison

Puisqu'il s'agit des applications multimédias, les métriques sont évidemment les paramètres de la qualité de service.

Une autre métrique classique a été ajoutée, il s'agit du taux de paquets délivrés.

Taux de paquets délivrés : Le nombre de paquets reçus sur le nombre de paquets envoyés.

Le délai de bout en bout : Le délai que dure un paquet avant d'être reçu.

Le débit : Le nombre de paquets reçus par seconde multiplié par la taille du paquet.

Le nombre de paquets rejetés : Le nombre de paquets rejetés par le protocole, cette métrique peut refléter le niveau de congestion que produit un tel protocole.

La gigue : La variation de délais, cette métrique est très importante, surtout pour les applications interactives.

| | |
|---------------------------|-----------------------|
| Type de trafic | CBR |
| Protocole de transmission | UDP |
| Taille du paquet | 200 octets |
| La portée | 250 m |
| Fréquence d'envoi | 1 paquet/seconde |
| Nombre de connexions | 3 durant 100 secondes |
| Durée de simulation | 200 secondes |

TABLE 4.1 – propriétés de simulation.

| | |
|------------------------|-----------|
| Topologie (m*m) | 2400*2400 |
| Nombre de noeuds | 150 |
| Vitesse maximale | 64 km/h |
| Nombre de routes | 1188 |
| Nombre d'intersections | 383 |

TABLE 4.2 – propriétés de l'environnement urbain.

4.6 Configurations et paramètres de simulation

Dans ce travail nous avons comparé les protocoles : AODV, DSDV et GPSR, quand le flux généré correspond à un trafic à débit constant (CBR, Constat Bit Rate), transporté par le protocole UDP, la taille du paquet est de 200 octets, la portée de chaque noeud est égale à 250 mètres, la fréquence d'envoi est de 1 paquet par seconde, trois connexions simultanées pendant 100 secondes, la simulation a duré 200 secondes voir le tableau 4.1. Les propriétés de l'environnement urbain sont illustrées dans le tableau 4.2.

4.7 Résultats

Pour les deux métriques le délai et la gigue, les résultats de l'AODV étaient en seconde, alors que pour les autres en milliseconde.

Le tableau 4.3 montre le taux de livraison, il s'avère claire que le protocole DSDV n'est pas recommandé à cause de perte, les deux autres protocoles offrent quasiment les mêmes résultats, cette perte peut être expliquée par le fait que les liaisons entre les noeuds ne sont pas toujours garanties, aussi puisque la durée de simulation ne dépasse pas 200 s, il est probable que d'autres paquets pourraient être reçus.

En termes de délai le protocole AODV n'était pas performant comparé avec les autres comme le montre la figure 4.1, malgré que le DSDV à cause de pertes

| Le protocole | Nb paquets émis | Nb paquets reçus | pourcentage |
|--------------|-----------------|------------------|-------------|
| AODV | 197 | 162 | 82.23% |
| GPSR | 195 | 149 | 76.41% |
| DSDV | 198 | 66 | 33.33% |

TABLE 4.3 – Taux de livraison.

consécutives ne couvre pas tout le temps, ce qui fait que le GPSR soit le meilleur (CF. Figure 4.2).

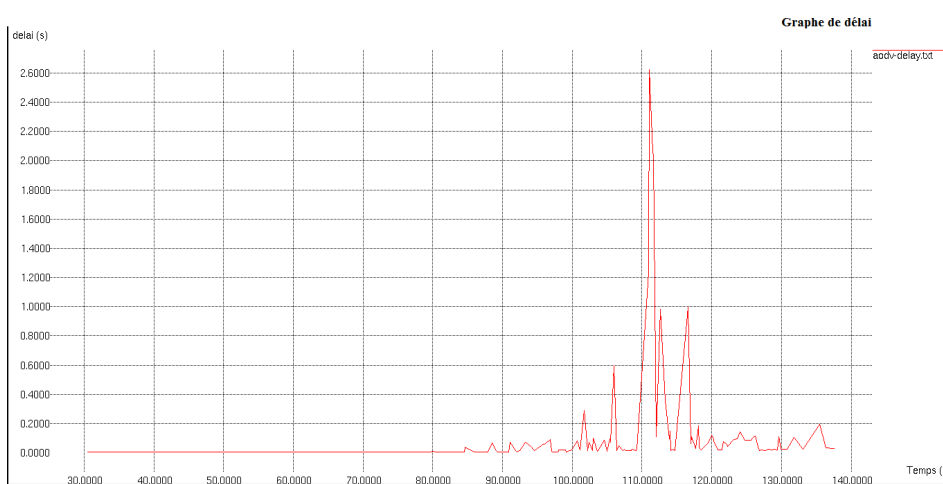


FIGURE 4.1 – graphe de délai pour le protocole AODV

Pour ce qui concerne le débit figure 4.3, une tendance claire à zéro, par la faible bande passante totale disponible 2 MB/S, les trois protocoles sont semblables.

concernant le rejet de paquets, aucun paquet n'a été perdu pour le GPSR (CF. Figure 4.4), ce qui est dû au fait que les paquets soient en train d'appliquer la règle de la main droite jusqu'à ce qu'un chemin à la destination soit disponible. Beaucoup de paquets ont été rejetés par l'AODV malgré que c'est le meilleur en termes de livraison. C'est généralement dû au fait qu'un paquet peut être supprimé plusieurs fois c'est-à-dire lorsqu'un chemin est modifié durant l'envoi, un message d'erreur va être acheminé vers la source pour qu'elle trouve un autre chemin, mais cette façon de faire a influencé le délai comme déjà vu.

Les résultats en termes de la gigue sont semblables à ceux de délai, le protocole AODV donne des résultats en seconde (2s maximum) figure 4.5, le protocole DSDV est meilleur que l'AODV concernant cette métrique, dû au fait que les chemins sont déjà mis dans la table de routage, avant qu'ils soient demandés, tandis que le protocole GPSR comme prévu donne de bons résultats (CF. Figure 4.6).

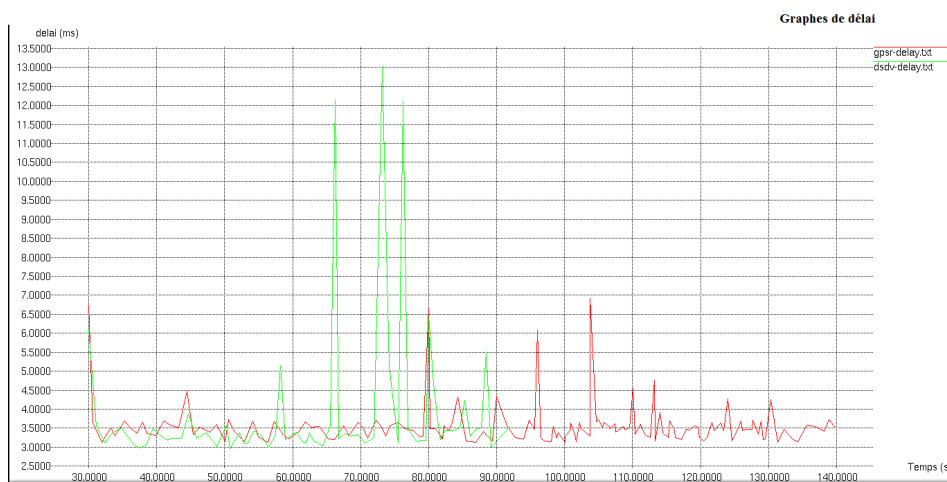


FIGURE 4.2 – graphes de délai pour les protocoles GPSR et DSDV

4.8 Discussions

Le délai, s'agissant des applications interactives, ne doit pas dépasser 1s, l'AODV n'arrive pas à satisfaire cette métrique, celui-ci a franchi la barrière de 2.5 s, le motif de ce retard à notre avis est que la route est à la demande donc le processus de détection de la route occupe une partie importante du délai. Les deux autres protocoles GPSR (max 6.5 ms) et DSDV (max 13 ms), ont pu satisfaire cette métrique mais celui dernier n'a pas pu couvrir tout le temps dû à la perte consécutive.

Les résultats obtenus en termes de la gigue sont comme ceux en termes de délai : AODV (max 1.9s), GPSR (max 3.5ms), DSDV (max 9.5 ms), on peut dire que la gigue ne pose pas un problème pour les deux derniers, ce qui est requis c'est (max 20 ms), concernant l'AODV une technique de bufférisation peut pallier à ce problème.

S'agissant de rejet de paquets, aucun paquets n'a été rejeté dans le cas de GPSR, vu que la règle de la main droite s'applique jusqu'à ce qu'un chemin soit disponible (aucun encombrement au niveau de la file d'attente). 70% des paquets routés par le DSDV n'ont pas été reçus, et seulement 17% et 23% des paquets routés par l'AODV, et le GPSR successivement, métrique n'ont satisfaites alors (3% max dans le cas de jeux).

Dû à la faible bande passante disponible 2MB/s, une tendance dès le début à zéro a été notée, un débit moyen de (2kO/s) métrique à peine satisfaites pour certains type de jeux (1KO/S), et près d'être satisfaites pour d'autres (5KO/s).

Quelques exigences ont été satisfaites donc, et d'autres ne l'ont pas été, tout simplement parce que les liaisons entre les véhicules ne sont pas toujours garanties, c'est-à-dire qu'il peut y avoir des cas où un tel véhicule ne possède aucun

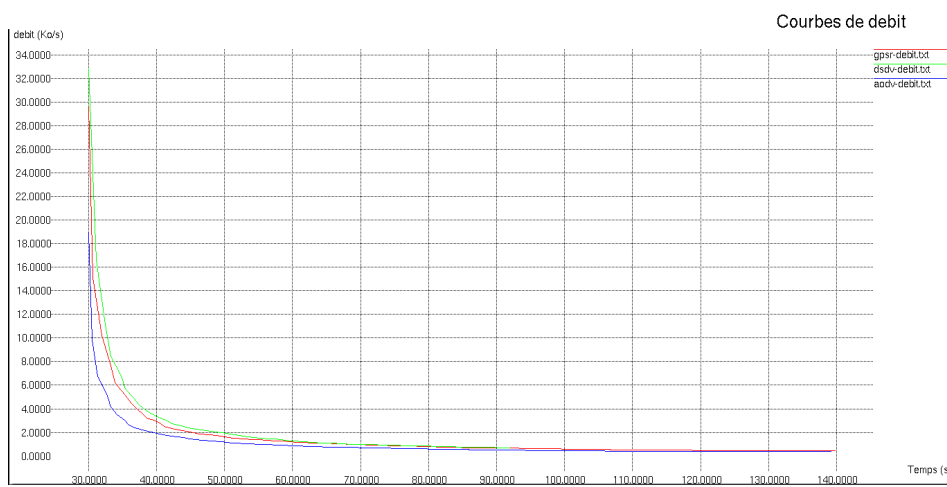


FIGURE 4.3 – graphes de débit pour les protocoles GPSR, DSDV, et AODV

voisin, ce qui peut avoir d'effet sur les paramètres de la QOS.

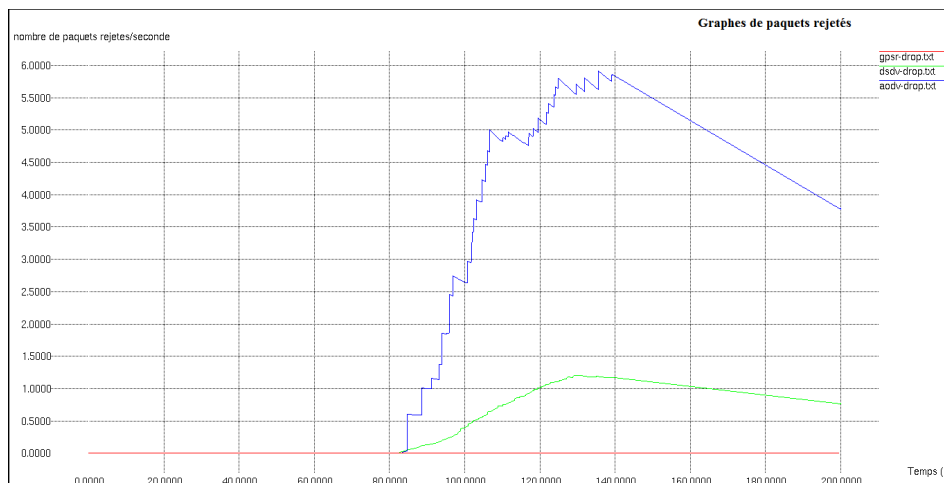


FIGURE 4.4 – graphes de rejet de paquets pour les protocoles GPSR, DSDV, et AODV

La métrique la plus difficile à satisfaire c'est la perte, sachant que tout paquet viens hors délai est supposé perdu, dans le cas des applications multimédias interactives y compris les jeux, ce genre d'applications est alors difficile à déployer sans la présence d'une suite d'infrastructures située sur la route, assurant la fiabilité, et garantissant la stabilité des liens. Dans ce cas les conditions de communication vont être améliorées, et alors une bonne réflexion sur les résultats.

Alors que les applications multimédias adaptatives peuvent l'être, mais il leur faut une technique de correction d'erreur comme celles déjà vues, et d'autres de recouvrement des paquets perdus (TCP par exemple).

D'autre point de vue, les protocoles proactifs à base de topologie représentés par le DSDV ne sont pas conseillés pour VANET, à cause de la congestion qu'ils produisent pour la mise à jour instantanée de leurs tables de routage. Alors que les protocoles réactifs, et à partir des résultats offerts par l'AODV peuvent concurrencer les protocoles géographiques comme le GPSR qui donne de bons résultats et est sans doute le meilleur.

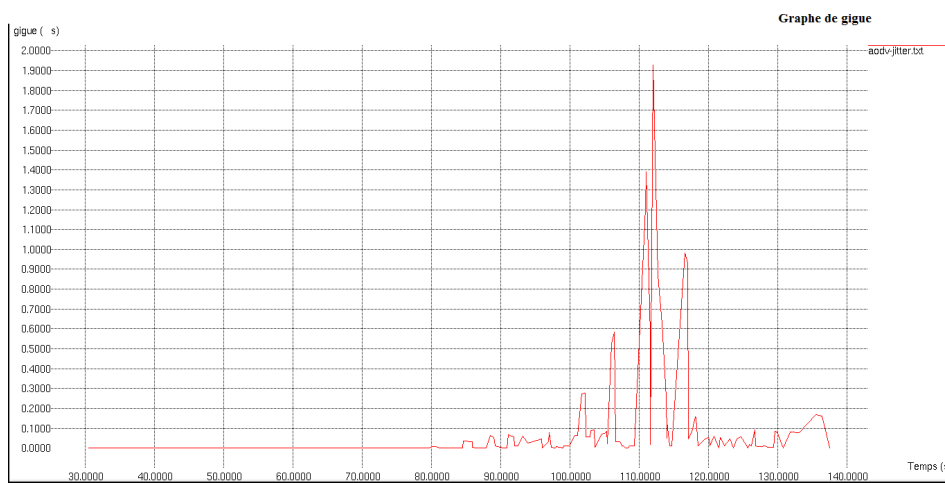


FIGURE 4.5 – graphe de gigue pour le protocole AODV

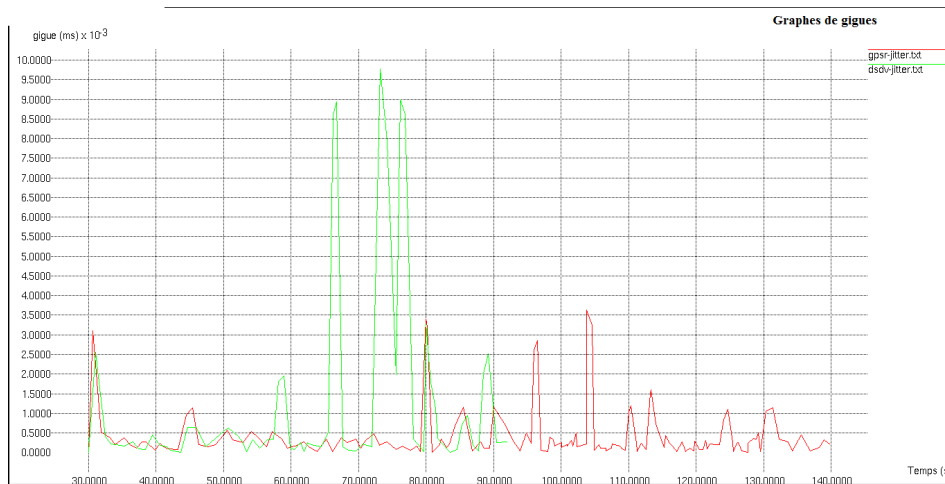


FIGURE 4.6 – graphes de gigue pour les protocoles DSDV, et GPSR

4.9 Conclusion

Dans cette partie, nous avons pris un protocole de routage de chaque catégorie. Pour tous les protocoles, la forte mobilité était un facteur influant qui dégrade les performances surtout en termes de débit et de taux de livraison.

Le DSDV, a fourni les meilleures performances en termes de délai, et de la gigue, mais le fait d'obtenir un taux de perte près de 70 % le rend inapplicable au réseau VANET, ce qui assure ce qu'on a déjà dit sur l'application des protocoles proactifs dans les réseaux à forte mobilité.

Le protocole AODV, qui est un réactif a été bon en termes de livraison, mais il lui fallait quelques secondes pour que ses paquets soient reçus, ce qui n'est pas désiré par les applications multimédias, Les résultats obtenus ont montré que le protocole GPSR est le plus adapté au contexte du vanet afin de fournir les services en temps réels liés au confort.

Conclusion générale

Le bilan général de ce travail peut être incarné par le rassemblement de plusieurs points de vue. Ce travail n'est pas seulement une évaluation des applications multimédias dans le VANET, mais aussi une comparaison entre les protocoles les plus utilisés dans ce domaine.

Les performances des applications multimédias souffrent de plusieurs problèmes dont la latence, la gigue,... etc dans le cas filaire, voire sans fil. Ceci est dû à la nature des exigences que nous avons détaillées dans le chapitre 1, ces exigences classifient les applications multimédias en plusieurs catégories. Chaque catégorie possède alors des propriétés spécifiques, et à peine des points communs avec les autres.

Dans le chapitre 2, on a présenté les réseaux VANETs juste avant d'aborder les caractéristiques des protocoles de routage MANET, afin d'en conclure ceux qui y sont applicables, en mettant en évidence leurs principes, leurs avantages et leurs désavantages. On a trouvé donc que les protocoles proactifs à base de topologie, ne conviennent pas à ce type de réseaux, tandis que les protocoles réactifs et les protocoles géographiques peuvent offrir de bons résultats.

Le chapitre 3 comprend en plus de l'état de l'art, quelques architectures récemment proposées pour les applications multimédias dans le VANET. Ces architectures sont limitées d'une part et améliorables d'autre part.

Les résultats des simulations sont présentés et analysés dans le chapitre 4. Comme perspectives à ce travail, nous proposons

- d'utiliser d'autres protocoles de routage,
- de simuler avec d'autres modèles de mobilités,
- d'étudier l'impact de la couche MAC sur les applications temps réels, et
- d'utiliser une architecture hybride (infrastructures).

Bibliographie

- [1] Alan Byrne, "Suitability Analysis of Routing Protocols in Vehicular Ad Hoc Networks for Distributed Multimedia Applications", Master dissertation, Trinity College Dublin, Ireland, sep 2008.
- [2] Bilal Mustafa , Umar Waqas Raja, "Issues of Routing in VANET", Master thesis, School of Computing, sweden, june 2010.
- [3] Ozan K. Tonguz, Mate Boban, "Multiplayer games over Vehicular Ad Hoc Networks : A new application", Journal of Carnegie Mellon University, United States, sep 2010.
- [4] Claudio Enrico Palazzi, "Fast Online Gaming over Wireless Networks", PhD thesis, University of California, United States, 2007.
- [5] N. Qadri, M. Altaf, M. Fleury, M. Ghanbari, "Robust Video Streaming over an Urban VANET, IEEE International Conference on Wireless and Mobile Computing", University of Essex (United Kingdom), University of Aleppo (Syrian Arab Republic), 2009.
- [6] Ozan Tonguz, Nawaporn Wisitpongphan, Fan Bai, Priyantha Mudalige, and Varsha Sadekar, "Broadcasting in VANET", in the paper of IEEE Communications Society subject matter experts for publication, Carnegie Mellon University, United States, 2008.
- [7] Maria Fazio, Claudio E. Palazzi, Shirshanka Das, Mario Gerla, "Facilitating Real-time Applications in VANETs through Fast Address Auto-configuration", Journal, university of california (United States), University of bologna (Italy), University of Messina (Italy), 2007.
- [8] Jerome Haerri , Fethi Filali , Christian Bonnet , "Performance Comparison of AODV and OLSR in VANETs Urban Environments under Realistic Mobility Patterns", Article, Institut Eurécom, France, 2006.
- [9] Nadia N. Qadri · M. Fleury · B. R. Rofoee · M. Altaf · M. Ghanbari, "Robust P2P Multimedia Exchange within a VANET", in Journal of Wireless Pers Commun, 2010

- [10] Josiane Nzouonta, Neeraj Rajgure, "Guiling Wang, VANET Routing on City Roads using Real-Time Vehicular Traffic Information", In TRANSACTIONS of IEEE VEHICULAR TECHNOLOGY, Sep 2009.
- [11] Vivek K Goyal, "Multiple Description Coding : Compression Meets the Network", in proc of IEEE signal processing magazine, 2001.
- [12] Hassen DKHIL , "Greedy perimeter stateless routing sur omnet++", mémoire d'ingénieur, Ecole nationale supérieur d'informatique, Tunisie, 2009.
- [13] Mariam Dawoud, "analyse du protocole AODV dans les réseaux VANET", DEA, Université Paul Sabatier, France, 2006.
- [14] Z. MAMMERI, "Réseaux et QoS" (support de cours).
- [15] Cédric LÉVY-BENCHETON, "Jeux Vidéo Multijoueurs sur Internet". 12 Décembre 2006
- [16] Nouha B.S, "Conception d'une nouvelle stratégie de routage dynamique pour les réseaux mobiles AD HOC", Mémoire de Mastère, École Nationale d'Ingénieurs de Sfax, Tunisie, 2006.
- [17] B. Ahmed, "Étude par Simulation des Performances des protocoles de Routage dans les Réseaux Ad-hoc sans Fil", thèse de Magister, Université Amar Telidji Laghouat, Algerie, 2007.
- [18] Sofiane Khalfallah, Bertrand Ducourthial, "Expérimentations des communications inter-véhicules", article de Colloque Francophone sur l'Ingénierie des Protocoles (CFIP), Université de Technologie de Compiègne, France, 11 Feb 2008
- [19] Presentation <http://wapiti.telecom-lille1.eu/commun/ens/peda/options/ST/RIO/pub/exposes/exposesrio2007/Bamouh-Lalaoui/presentation.html>, 1 Mai 2011.
- [20] M. Samuel Lucier, "développement d'un protocole de routage basé sur le potentiel et adapté aux réseaux de type VANET", rapport de stage d'ingénieur, Wuhan University, China, 4 septembre 2007.
- [21] Florent Kaisser, Anthony Busson, Véronique Vèque, "Propriétés structurales des réseaux ad hoc de véhicules", poster de Colloque Francophone sur l'Ingénierie des Protocoles (CFIP), University Paris Sud, France, 8 Mars 2008
- [22] A. Rachid, SU-OLSR une nouvelle solution pour la sécurité du protocole OLSR, mémoire de Maitrise, Université du Québec, Canada, 2009.
- [23] Modèles de mobilité <http://www.lst.inf.ethz.ch/research/ad-hoc/car-traces/>, 3 juin 2011.