

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE  
SCIENTIFIQUE  
UNIVERSITE MOHAMMED SADDIK BEN YAHYA - JIJEL



*Faculté des Sciences Exactes et Informatique  
Département d'Informatique  
Spécialité: Réseaux et Sécurité*



# Mémoire

*En vue de l'obtention du diplôme de Master en informatique*

## THEME

*ETUDE DE STREAMING VIDEO DANS LES RESEAUX DE  
MOBILES AD-HOC A TRAVERS LE SIMULATEUR OMNET++*

*Encadré par :*

*Mr: Rouibah said*

*Présenté par :*

*Bourabia Afifa*

*Bourebia Soumia*

2015/2016



M. inf - RS 03/16

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE  
SCIENTIFIQUE  
UNIVERSITE MOHAMMED SADDIK BEN YAHYA - JIJEL



Faculté des Sciences Exactes et Informatique  
Département d'Informatique  
Spécialité: Réseaux et Sécurité



# Mémoire

En vue de l'obtention du diplôme de Master en informatique

## THEME

ETUDE DE STREAMING VIDEO DANS LES RESEAUX DE  
MOBILES AD-HOC A TRAVERS LE SIMULATEUR OMNET++

Encadré par :

Mr: Rouibah said

Présenté par :

Bourabia Afifa

Bourebria Soumia

2015/2016

## **Résumé**

*Le présent mémoire a été élaboré dans le cadre du projet de fin d'études pour l'obtention du diplôme de master en informatique, option : réseaux et sécurité. Ce travail consiste à étudier le streaming vidéo dans les réseaux de mobiles ad-hoc à travers le simulateur OMNET++. Notre première préoccupation est la présentation des différentes notions de base que nous avons étudié pour le besoin de ce mémoire: le streaming vidéo, les réseaux de mobiles ad-hoc et la synergie entres ces deux axes. Enfin, nous présentons l'environnement de travail et les résultats de nos simulations réalisées au cours de ce projet.*

**Mot clés** : streaming vidéo, réseaux de mobiles ad-hoc, OMNET++

## **Abstract**

*This work was developed as part of the graduation project for obtaining master degree diploma in Specialty of Networks and Security. It is about studying video streaming in mobile ad hoc networks through OMNET++ simulator. Our first concern is the presentation of the various basics that we studied for the need of this note: video streaming, ad-hoc mobile networks and synergy between these two axes. Finally, we present our working space and the results of our simulations achieved during this project.*

**Keywords:** *streaming video, mobile ad hoc networks, OMNET++.*

## *Remerciements*

*Avant toute chose, nous remercions Dieu le tout puissant qui nous a donné la force et la patience.*

*Au terme de ce travail, la première personne que nous tenons à remercier est notre encadrant Mr ROUIBAH SAID pour avoir bien accepté de diriger notre travail, pour l'aide qu'il nous a apporté, ses précieux conseils, sa patience et son encouragement.*

*Nos vifs remerciements vont également aux membres de jury pour l'intérêt qu'ils ont porté à notre travail.*

*Un grand merci à tous les enseignants qui ont contribué à notre formation, pour enrichir nos connaissances et nous guider durant toutes les années d'études.*

*Un grand merci à nos parents non seulement pour nous avoir encouragés et supportés tout au long de nos études mais dans toutes les sphères de notre vie.*

*Un grand merci à nos frères, nos sœurs, pour leurs encouragements et supports.*

*Nous remercions également nos amis qui nous permettent de garder un équilibre de vie et pour leur soutien durant la période de rédaction de ce mémoire.*

*Nous remercions tous ceux qui ont contribué de près ou de loin à l'aboutissement de ce travail.*

*A tous ceux que nous aimons, et à tous ceux qui nous aiment.*

*Merci infiniment...*



## **Dédicace**

*Je dédie ce modeste travail à :*

*Ma famille:*

*Mon père, gentil, attentionné, tendre, patient, cultivé, sage, drôle, rassurant, le bonheur de ma vie, mon premier encadrant depuis ma naissance, qui m'a appris de respecter les autres, d'apprendre de mes erreurs, de viser toujours plus haut, et poursuivre mes rêves.*

*Ma mère la fleur de ma vie, qui a œuvré pour ma réussite, de par son amour, son soutien, tous les sacrifices consentis et ses précieux conseils, pour toute son assistance et sa présence dans ma vie, reçois à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude.*

*Mes frères et ma sœur :*

*Mon grand frère Mourad qui m'a toujours soutenu et poussé en avant, pour son encouragement, sa tendresse et générosité.*

*Hichame et Mouhamed , mes deux princes , la joie de ma vie.*

*Ma très chère sœur Nour el Houda , Mon âme sœur et ma fidele amie , le cadeau du ciel à moi, pour sa présence à coté de moi , son encouragement , son support et son amour.*

*Mon Promoteur :Mr.Rouibah Said.*

*Mon binôme AFIFA, ma très chère amie et la moitié de mon âme, pour sa présence à coté du moi durant les difficiles moments, pour sa patience, et sa générosité.*

*A toute sa famille.*

*Toute la famille BOUREBIA et BOUKHALOUT.*

*Mes chères amies FATHIA, BOUCHRA et a tous mes amis.*

*Tous ceux que je les aime, et qui m'aiment ...*

**SOUMIA**

## ***Dédicace***

*Je dédie ce travail à :*

*Mes chers parents, source de tendresse, de patience et de générosité.  
Je prie le bon dieu de les bénir, de vieillir sur eux en espérant qu'ils seront  
toujours fières de moi.*

*Mes chères sœurs AMIRA et AMEL qui m'ont placé la barre haute et qui  
m'ont toujours poussé en avant.*

*Mes chères frères ISLAM et MOHAMMED EL-ADLANI qui m'ont toujours  
soutenu.*

*Ma toute première nièce MERIEM.*

*Mon encadrant Mr.ROUIBAH SAID.*

*Mon chère binôme SOUMIA, qui a eu la patience de me supporter durant  
ce mémoire, et qui m'a soutenu et encouragé pendant tous les moments  
difficiles vécus et à toute sa famille*

*Toute la famille BOURABIA et BELGUESSAB.*

*Mes chères amies SABRINA, SOUMIA et à tous mes amis.*

*Tous ceux qui, par un mot, m'ont donné la force de continuer ...*

**AFIFA**

# I

## TABLE DES MATIERES

Table des matières .....	I
Liste des figures .....	II
Liste des tableaux .....	III
Liste des acronymes .....	IV
Introduction générale .....	1

### *Chapitre 01 : Généralités sur le streaming*

1. Introduction .....	3
2. Définition du streaming vidéo .....	3
3. Techniques de streaming .....	3
3.1. Streaming progressif .....	4
3.1.1. Les avantages de streaming progressif .....	4
3.1.2. Les inconvénients de streaming progressif .....	4
3.2. Streaming continu .....	4
3.2.1. Les avantages de streaming continu .....	5
3.2.2. Les inconvénients de streaming continu .....	5
4. Comparaison entre le streaming progressif et le streaming continu.....	5
5. L'architecture de streaming .....	5
6. La compression .....	6
6.1. Formats de compression.....	6
6.1.1. Compression avec perte .....	6
6.1.2. Compression sans perte.....	6
6.1.3. Compression des fichiers audio .....	7
6.1.4. Compression des fichiers vidéo .....	7
7. Exemple d'architectures de streaming .....	7
8. Contraintes et besoins pour le streaming .....	8
9. Méthodes de distribution .....	8
9.1. Diffusion unicast .....	8
9.2. Diffusion multicast .....	9
9.3. Broadcast .....	10
10. Les modes de diffusions .....	10
10.1. Mode PUSH .....	10



10.2. Mode PULL .....	10
10.3. Mode HYBRIDE .....	11
11. Les protocoles dédiés aux streaming .....	12
11.1. TCP .....	12
11.2. UDP .....	12
11.3. Comparaison entre le streaming TCP et le streaming UDP.....	12
11.4. RTP .....	13
11.5. RTCP .....	13
11.6. RTSP .....	13
11.7. SRTP .....	14
11.8. HTTP .....	14
12. Solutions pour faire le streaming .....	14
12.1. Codage redondant .....	14
12.2. Les réseaux CDN .....	14
12.3. Streaming vidéo en pair à pair .....	14
13. Les champs d'application du streaming .....	15
13.1. La vidéo à la demande .....	15
13.2. La Vidéoconférence .....	15
13.3. Les jeux en réseau .....	15
14. Statut juridique du streaming .....	15
15. Conclusion .....	16

## ***Chapitre 02 : Les réseaux de mobiles ad-hoc***

1. Introduction .....	17
2. Les environnements mobiles .....	17
2.1 Architecture avec infrastructure .....	17
2.2 Architecture sans infrastructure .....	18
3. Définition des réseaux mobiles ad hoc .....	19
4. Caractéristiques des réseaux mobiles ad hoc .....	20
5. Modélisation d'un réseau mobile ad hoc .....	21
6. La communication dans les réseaux de mobiles ad-hoc .....	22
7. Problèmes liés aux réseaux de mobiles ad-hoc .....	22
8. Domaines d'application des réseaux mobiles ad-hoc .....	23
9. Les protocoles de routage des réseaux mobiles ad hoc .....	23

9.1. Les protocoles proactifs .....	24
9.1.1. OLSR .....	24
9.1.2. DSDV .....	24
9.1.3. GSR .....	24
9.1.4. FSR .....	25
9.2. Les protocoles réactifs .....	25
9.2.1. AODV .....	25
9.2.2. DSR .....	26
9.2.3. CEDAR .....	26
9.2.4. ABR .....	26
9.3. Les protocoles hybrides .....	27
9.3.1. ZRP .....	27
9.3.2. ZHLS .....	27
9.3.3. CBRP .....	27
9.4. Les avantages et les inconvénients des protocoles de routages des réseaux de mobiles ad hoc .....	28
10. Le streaming vidéo sur les réseaux de mobiles ad-hoc .....	28
11. Les problèmes de streaming dans les réseaux de mobiles ad-hoc .....	29
12. Les techniques de streaming dans les réseaux de mobiles ad-hoc .....	29
12.1. Codage par description multiple .....	29
12.2. Routage multi chemins .....	30
12.3. Inter-couches .....	31
13. La qualité de service .....	32
14. Conclusion .....	32

### ***Chapitre 03 : La simulation avec OMNET++***

1. Introduction .....	33
2. Généralités .....	33
2.1. La simulation .....	33
3. Types de simulation .....	33
3.1. Systèmes de simulation continus .....	33
3.2. Systèmes de simulation discrets .....	33
4. Les outils de simulation réseau .....	34
4.1. NS2 .....	34

4.2. OPNET++ .....	34
4.3. GIOMOSIM .....	34
4.4. OMNET++ .....	35
5. Comparaison entre les simulateurs .....	35
6. Le simulateur OMNET++ .....	36
6.1. Présentation d'OMNET++ .....	36
6.2. Description architecturale d'OMNET++.....	36
6.3. Les principaux fichiers d'OMNET++.....	37
6.3.1. Fichier .ned .....	37
6.3.2. Fichier.ini .....	38
6.3.3. Fichier.msg .....	39
7. Les plates formes d'OMNET++ .....	39
7.1. INET .....	39
7.1.1. Définition .....	39
7.1.2. Architecture d'INET .....	39
7.2. MIXIM .....	40
7.2.1. Définition .....	40
7.3. Intégration d'INET et MIXIM .....	40
8. Conclusion.....	41

### ***Chapitre 04 : Analyse et conception***

1. Introduction .....	42
2. Analyse .....	42
2.1. Le but de projet .....	42
2.2. Problématique .....	42
2.3. Partie d'émulation temps réel .....	42
2.4. Partie d'émulation virtuelle .....	43
3. Conception .....	44
3.1. Partie d'émulation temps réel .....	44
3.1.1 Module Listener .....	45
3.1.2 ModuleServeur .....	45
3.1.3 ModuleClient .....	45
3.2 Partie d'émulation virtuelle .....	46
3.2.1 ModuleUdpvideostreamCli .....	46

3.2.2 ModuleUDPvideostreamSrv .....	46
4. Conclusion .....	47

## ***Chapitre 05 : Réalisation***

1. Introduction .....	48
2. L'environnement de travail .....	48
2.1. Configuration matériel .....	48
2.2. Logiciels .....	48
2.2.1. OMNET++ .....	48
2.2.2. Le Framework INET2.99.1 .....	48
2.2.3. VLC .....	49
3. Description de réseau sous OMNET++ .....	52
4. Paramètres et variables de simulation .....	54
4.1. Paramètres de simulation .....	54
4.1.1. Le débit .....	54
4.1.2. Les paquets de contrôle .....	55
4.1.3. Les paquets utiles .....	55
4.1.4. Les paquets perdus .....	55
4.1.5. Le trafic émis .....	55
4.1.6. Le trafic routé .....	55
4.1.7. La qualité de la vidéo reçue .....	55
4.2. Modèle de topologie .....	56
4.3. Modèle d'énergie .....	56
4.4. Les protocoles de routage utilisés .....	56
4.5. Nombre de nœuds .....	57
4.6. Modèle de Mobilité .....	57
4.7. Modèle de trafic .....	57
4.8. Analyse des résultats .....	58
4.8.1. Scénario 01 : Taux de réception (Trecep/vitesse de nœuds) .....	58
4.8.2. Scénario 02 : Nombre de paquets perdus (nbrpqt/vitesse de nœuds).....	59
4.8.3. Scénario 03 : Nombre de paquets de contrôle (nbreC /vitesse de nœuds) .....	60
4.8.4. Scénario 04 : délai de bout en bout en fonction de nombre de nœuds .....	61
4.8.5. Scénario 05 : énergie restante en fonction de modèle de mobile.....	62
4.8.6. Scénario 06 : énergie restante restante en fonction de temps de simulation....	62

4.8.7. Impact du facteur Paquet perdu sur la qualité de réception de la vidéo .....	63
4.8.8. Impact du facteur de variation de délai D (Gigue) sur la qualité de la réception de la vidéo.....	64
5. Conclusion .....	65
Conclusion générale .....	66
Bibliographie .....	67
Webographie .....	68

## Liste Des Figures

### **Chapitre 01 : Généralités sur le streaming**

Figure 1- 1 : Techniques de streaming .....	4
Figure 1- 2 : Architecture de streaming.....	5
Figure 1- 3 : l'unicast.....	9
Figure 1- 4 : le multicast.....	9
Figure 1- 5 : mode PUSH.....	10
Figure 1- 6 : mode PULL.....	11
Figure 1- 7 : mode hybride.....	11

### **Chapitre 2 : Les réseaux de mobile ad hoc**

Figure 2- 1 : Le modèle de réseau mobile avec infrastructure .....	18
Figure 2- 2 : Le modèle de réseau mobile sans infrastructure.....	18
Figure 2- 3 : Exemple d'un MANET .....	19
Figure 2- 4 : La modélisation d'un réseau de mobiles ad-hoc .....	21
Figure 2- 5 : Changement des topologies .....	22
Figure 2- 6 : classification des protocoles.....	23
Figure 2- 7 : exemple MDC .....	30

### **Chapitre 3 : La simulation avec OMNET++**

Figure 3- 1 : module composé et simple .....	36
Figure 3- 2 : fichier .ned en mode graphique .....	37
Figure 3- 3 : fichier .ned en mode texte .....	38
Figure 3- 4 : exemple d'un fichier.ini .....	38
Figure 3- 5 : exemple d'un fichier.msg .....	39
Figure 3- 6 : Intégration de MIXIM et INET dans un projet OMNET.....	40

### **Chapitre 4 : Analyse et Conception**

Figure 4-1 : Analyse et conception .....	43
Figure 4- 2 : émulation virtuelle .....	44

### **Chapitre 5 : Réalisation**

Figure 5- 1 : L'option flux.....	49
Figure 5- 2 : La boîte de dialogue ouvrir média.....	50

Figure 5- 3 : Choix de protocole de transport.....	50
Figure 5- 4 : Configuration de la destination.....	51
Figure 5- 5 : Choix de transcodage.....	51
Figure 5- 6 : Zone réseau. ....	52
Figure 5- 7 : structure de réseau .....	53
Figure 5- 8 : modèle de topologie.....	56
Figure 5- 9 : MassMobility.....	58
Figure 5- 10 : RandomMobility.....	58
Figure 5- 11 : LinéaireMobility.....	58
Figure 5- 12 : MassMobility.....	59
Figure 5- 13 : RandomMobility.....	59
Figure 5- 14 : LinéaireMobility.....	59
Figure 5- 15 : MassMobility .....	60
Figure 5- 16 : RandomMobility.....	60
Figure 5- 17 : LinéaireMobility.....	60
Figure 5- 18 : délai de bout en bout Vs nombre de nœuds.....	61
Figure 5- 19 : énergie restante Vs le modèle de mobilité.....	62
Figure 5- 20 : énergie restante Vs temps de simulation pour le protocole AODV.....	63
Figure 5- 21 : énergie restante Vs temps de simulation pour le protocole DSR.....	63
Figure 5- 22 : Impact du facteur Paquet perdu sur la qualité de réception de la vidéo.....	64
Figure 5- 23 : Impact du facteur de la gigue sur la qualité de la réception de la vidéo.....	65



**Liste Des Tableaux*****Chapitre 01 : Généralités sur le streaming***

Tableau 1-1 : Comparaison entre le streaming progressif et le streaming continu .....5

Tableau 1- 2 : Comparaison entre le streaming TCP et le streaming UDP ..... 13

***Chapitre 03 : La simulation avec OMNET++***

Tableau 3- 1 : Comparaison entre les simulateurs .....36

***Chapitre 05 : Réalisation***

Tableau 5- 1 : Equipements de manipulation .....48

Tableau 5-2 : Paramètres de simulation .....58



## Liste des acronymes

<b>AAC</b>	Advanced Audio Coding
<b>ABR</b>	Associativity Based Routing
<b>AODV</b>	Ad-hoc On Demand Distance Vector
<b>CBRP</b>	Cluster Based Routing Protocol
<b>CDN</b>	Content Delivery Network
<b>CEDAR</b>	Core Extraction Distributed Ad hoc Routing
<b>CPU</b>	central processing unit
<b>DBF</b>	distance victor belman ford
<b>DIFFSERV</b>	Differential Services
<b>DSDV</b>	Distance Source Distance Vector
<b>DSR</b>	Dynamic Source Routing
<b>FSR</b>	Fisheye State Routing
<b>GSM</b>	Global System for Mobile Communications
<b>GSR</b>	Global State Routing
<b>HTTP</b>	Hyper Text Transfer Protocol
<b>IEEE</b>	Institute of Electrical and Electronics Engineers
<b>IPV4</b>	Internet Protocol Version 4
<b>IPV6</b>	Internet Protocol Version 6
<b>JMPEG</b>	Joint Photographic Experts Group
<b>MPEG4</b>	Moving Picture Experts Group 4
<b>LAN</b>	Local Area Network
<b>LTE</b>	Liquid Tension Experiment (band)
<b>MAC</b>	Media Access Control
<b>MANET</b>	Mobile Ad Hoc Network
<b>MDC</b>	Codage par description multiple
<b>MJPEG</b>	Motion joint Picture Experts Group
<b>MPEG1</b>	Motion Picture Experts Group 1
<b>MPEG2</b>	Motion Picture Experts Group 2
<b>MP3</b>	Moving Picture Expert Group 3
<b>MP4</b>	Moving Picture Expert Group 4

<b>OLSR</b>	Optimized Link State Routing
<b>OTCL</b>	Object Tool Command Language
<b>OSI</b>	Open Systems Interconnection
<b>PAN</b>	Personnel Area Network
<b>PARSEC</b>	Parallel Simulation Environment for Complex Systems
<b>PPP</b>	Point-to-Point Protocol
<b>RFC</b>	Request For Comments
<b>RTCP</b>	Real Time Control Protocol
<b>RTP</b>	Real Time Transport Protocol
<b>RTSP</b>	Real Time Streaming Protocol
<b>SRTP</b>	Sequenced Routing Update Protocol
<b>TCL</b>	Tool Command Language
<b>TCP</b>	Transmission Control Protocol
<b>UCLA</b>	University of California, Los Angeles
<b>UDP</b>	User Datagram Protocol
<b>UMTS</b>	Universal Mobile Telecommunications System
<b>VIP</b>	Video Over IP
<b>VLC</b>	Vidéo LanClient
<b>VOD</b>	Video On Demand
<b>WAV</b>	Windows Wave(audio format/file extension)
<b>WWW</b>	World Wide Web
<b>ZHLS</b>	Zone-Based Hierarchical Link State
<b>ZRP</b>	Zone Routing Protocol

## *Introduction générale*

Les progrès récents dans la technologie informatique, la technologie de compression, les dispositifs de stockage, Et les réseaux à grande vitesse ont rendu possible de fournir des services multimédias en temps réel sur l'internet. Actuellement la technologie de streaming est devenue très répondu, de nombreux sites et applications l'utilisent pour proposer des contenus multimédia à leurs utilisateurs, en offrant une expérience de visualisation plus satisfiable et plus amusante. D'une autre part le développement récent des technologies des réseaux sans fil et des appareils informatiques mobiles fournissent la plate forme technique pour étendre ces services de streaming au nombre croissant des utilisateurs finaux.

Le développement important des réseaux sans fil, aussi bien à l'intérieur des entreprises, pour remplacer des réseaux filaires traditionnels, qu'au niveau des lieux publics, ainsi que le grand succès connu par la téléphonie mobile, ont montré l'intérêt commercial des moyens de transmissions sans fil et laisse à penser que l'utilisateur va devenir de plus en plus mobile. Dans ce contexte, des travaux de recherches ont débuté afin d'obtenir des solutions de communication sans fil de plus en plus performantes. Et s'affranchir de toute contrainte, en formant ainsi des réseaux mobiles totalement dynamiques et spontanés. Il s'agit des réseaux de mobiles Ad Hoc.

Un réseau de mobile ad hoc est une collection d'entités mobiles, interconnectées par une technologie sans fil formant un réseau temporaire sans l'aide d'aucune infrastructure préexistante ou administration centralisée. Dans de tels environnements, les hôtes mobiles sont obligés de se comporter comme des routeurs afin de maintenir les informations de routage du réseau. Cette nouvelle technologie constitue un domaine de recherche actif et ouvre de nombreuses perspectives, cela est du essentiellement à la diversité des applications potentielles, ainsi qu'aux contraintes introduites par ces réseaux, entre autre : la mobilité des nœuds, le changement dynamique de la topologie, la bande passante limitée, des sources d'énergie autonomes ...etc. Ces contraintes font que les solutions retenues pour les réseaux classiques ne sont pas applicables directement aux réseaux de mobiles ad hoc.

Comme pour les réseaux filaires, le besoin de transmettre de différents types de média sur les réseaux de mobiles Ad hoc est important et même nécessaire. Les application de streaming telles que la téléphonie, la vidéo à la demande ou la vidéo conférence , exigent un transfert de données complexe et l'on voit apparaître un réel besoin

de garantie sur la qualité de service offerte , cependant, au regard des spécificités des réseaux de mobiles Ad hoc, la garantie d'une qualité de service pour certains types d'applications est une tâche très complexe, puisque ce type de réseaux introduit des défis supplémentaires qui doivent être traités pour fournir un bon service de visualisation de contenu multimédia ,ces défis sont causés par le moins de ressources disponibles(CPU, la bande passante ,la capacité de stockage, l'énergie...),la nature dynamique de l'environnement(la disponibilité des ressources)et le taux d'erreur élevé(due de changement des route).

Ce mémoire est structuré en cinq chapitres :

Dans le premier chapitre, nous avons présenté les différents concepts et les contraintes liés au domaine de streaming. Le deuxième chapitre est consacré aux réseaux de mobiles Ad hoc, en mettant la lumière sur ses caractéristiques, ses spécificités, les protocoles de routages existants et le streaming dans ce type de réseaux. Dans le troisième chapitre, nous présentons plusieurs simulateurs de réseaux et les outils qu'on a utilisé pour le développement de notre application. Nous consacrons le quatrième chapitre, à la présentation de l'étape d'analyse et de conception de notre application. Dans Le cinquième chapitre nous décrivons l'environnement de travail, les détails de la réalisation de certaines fonctionnalités ainsi que les résultats de simulation obtenus. Nous terminerons ce mémoire par une conclusion générale.



*Généralités sur le  
streaming*

Chapter 1

## **1. Introduction**

Après la grande révolution dans le domaine informatique et l'apparition des multiprocesseurs qui sont capables d'exécuter des milliers d'instructions dans un temps non remarquable, il est devenu possible de réaliser des systèmes de streaming pour la transmission des flux continus en temps réel, sans restriction sur la taille des données.

Par conséquent les données multimédia ont connu des grands changements dans les modes d'acquisition, de distribution et de consommation. L'utilisateur final peut jouir d'une représentation sur la volée de contenu multimédia sans avoir à télécharger le fichier entier. Ainsi, l'accès en temps réel interactive et progressive au contenu multimédia devient une réalité avec le streaming. Actuellement cette technique de délivrance de la vidéo et d'audio sur internet est très importante pour plusieurs entreprises, non seulement pour le lancement des produits sur le webcast mais aussi pour la diffusion des contenus plus complexe, comme la vidéo on demande (VOD) et les événements en direct. Elle est aussi utilisée par des milliers de sites web qui offrent aux internautes la possibilité de visualiser les données sans téléchargement.

Dans ce chapitre nous mettons l'accent sur la technique de streaming en décrivant son principe de fonctionnement, les protocoles utilisées, les modes de distribution, les domaines d'applications et son statut juridique dans le monde.

## **2. Définition du streaming vidéo**

Le streaming vient du mot anglais Stream qui signifie flux. La technologie du streaming permet la génération, la diffusion et la lecture de Stream (vidéo, audio ou les deux) au fur et à mesure de sa réception en « flux continu ». D'autres types de médias (photos, livres électroniques, applications ...etc.) n'ont pas besoin de cette méthode de transfert, car n'ayant aucune contrainte de séquentialité, ils sont téléchargés entièrement avant d'être traités, exécutés ou affichés.

## **3. Techniques de streaming**

Il existe plusieurs techniques du streaming, la figure suivante représente quelques exemples.

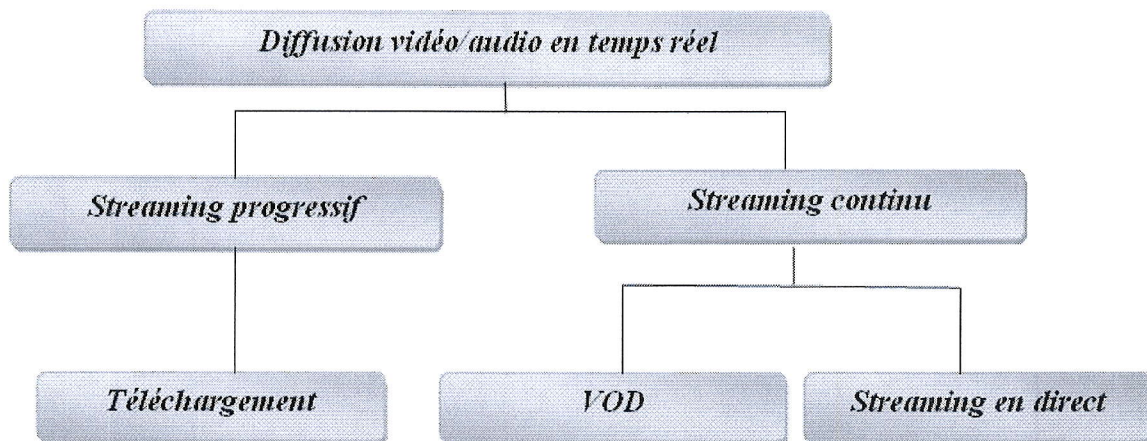


Figure 1-1 : Techniques de streaming.

### 3.1. Streaming progressif

Aussi appelé pseudo streaming ou streaming statique, repose sur le protocole HTTP, il permet à l'utilisateur de lire un média alors que le fichier est en cours de téléchargement, c'est le navigateur qui se charge d'effectuer la lecture de la vidéo, avec le streaming progressif une copie du fichier est conservé sur le disque dur de l'utilisateur.

#### 3.1.1. Les avantages de streaming progressif

- Ne nécessite pas un serveur spécialisé, un serveur HTTP étant suffisant.
- Une fois que la vidéo est stockée (ou cachée) sur un disque dur, il est très facile à la copier.

#### 3.1.2. Les inconvénients de streaming progressif

- Ne peut pas transmettre de flux en temps réel.
- Ne peut pas sauter un passage de la vidéo sans télécharger tout le début.
- Ne s'adapte pas à la qualité de la connexion de l'utilisateur.
- Induit une attente pour avoir les premières images si le fichier est de taille importante.

### 3.2. Streaming continu

Aussi appelé le vrai streaming ou streaming dynamique, il utilise un serveur spécial dédié au streaming (ex : xiph Icecast, real Hilex streaming server) pour diffuser l'information à différents niveaux de qualité, il permet de parcourir, avancer et de revenir à un point donné du contenu, contrairement au streaming statique qui requiert de visionner le contenu de début à la fin.

**3.2.1. Les avantages de streaming continu**

- Ne laisse pas de copie média sur le disque dur du client.
- Pas besoin de télécharger le fichier.
- Transmission en temps réel.
- Adaptation avec la qualité de connexion de l'utilisateur.

**3.2.2. Les inconvénients de streaming continu**

- Problème de perte de donnée lors de transmission.
- Utilise un serveur spécialisé généralement couteux.

**4. Comparaison entre le streaming progressif et le streaming continu**

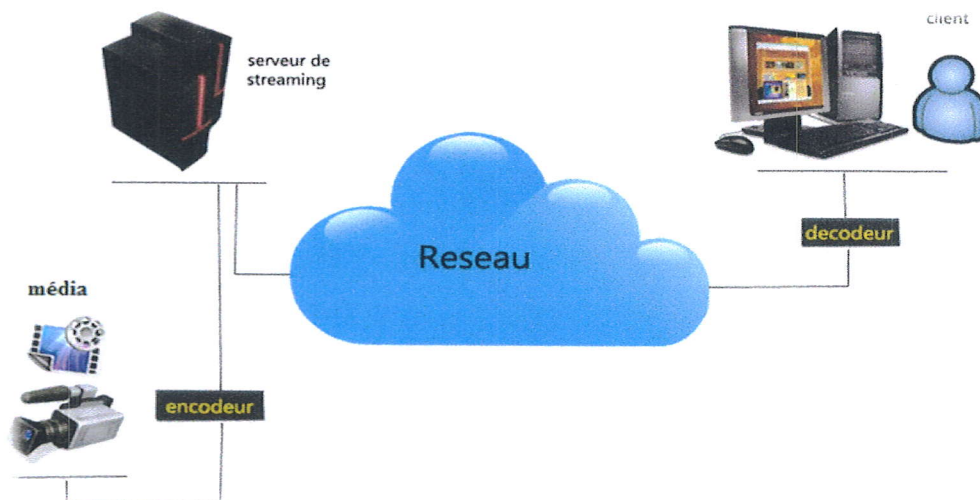
Streaming continu	Streaming progressif
<ul style="list-style-type: none"> <li>- Utilise un serveur spécial de streaming.</li> <li>- Utilisation des protocoles RTP/RTCP sur UDP.</li> <li>- Pas de copie de média sur le disque.</li> </ul>	<ul style="list-style-type: none"> <li>- Utilise le serveur HTTP.</li> <li>- Utilisation du protocole HTTP sur TCP.</li> <li>- Téléchargement progressif avec copie de média sur le disque de l'utilisateur.</li> </ul>

**Tableau1-1 :** Comparaison entre le streaming progressif et le streaming continu.

**5. L'architecture de streaming**

La plupart des systèmes de streaming fonctionnent selon un modèle client/serveur, Le client demande des données à partir d'un serveur sur un réseau, le serveur se contente de lui délivrer ces données comme un flux audio et vidéo continu codé ou compressé (voir section6) dans un format spécial qui rétrécit les données à une taille gérable.

La figure suivante représente l'architecture d'un système basé sur le streaming :



**Figure 1-2 :** Architecture de streaming.



La génération d'un flux vidéo/audio est suivit par une opération d'encodage qui sert à faire compresser le contenu du média à fin de diminuer la taille de fichier, l'opération inverse (décompression) est réalisé lors de la lecture du fichier par le client.

Les médias sont ensuite envoyés à un serveur de streaming qui analyse la bande passante des clients et qui adapte la qualité de diffusion à cette dernière.

A l'arrivée d'une demande de visualisation par un client, le serveur place la réponse dans une mémoire tampon. Quand il ya assez de données dans cette mémoire pour permettre de lire le début du fichier audio ou vidéo, la lecture démarre et le téléchargement du flux continu pour alimenter la mémoire tampon avec la suite du fichier.

## **6. La compression**

De nos jours, la puissance des processeurs augmente plus vite que les capacités de stockage et de la bande passante des réseaux car cela demande d'énormes changements dans les infrastructures de télécommunication, ainsi pour palier à ce manque, il est courant de réduire la taille des données en exploitant la puissance des processeurs plutôt qu'en augmentant les capacités de stockage et de transmission de données.

La compression est un processus qui permet de diminuer la taille des données par encodage en utilisant un codec<sup>1</sup>, ce qui permet le stockage et la transmission efficace de ces données. Le processus inverse est appelé la décompression.

### **6.1. Formats de compression**

#### **6.1.1. Compression avec perte**

Une compression avec perte détruit la possibilité d'avoir les données originales depuis les données compressées, il s'agit d'une représentation en éliminant les données inutiles en se focalisant sur les données significatives seulement. Il existe plusieurs algorithmes de compression avec perte comme : MP3, AAC, JMPEG...etc.

#### **6.1.2. Compression sans perte**

Une compression qu'elle soit vidéo, audio ou autre est dite sans perte si on peut obtenir les données originales d'après les données compressées. Il existe plusieurs algorithmes de compression sans perte comme : MPEG4, WAV...etc.

---

<sup>1</sup> Codec : Vient de compression/décompression, permet de traiter une donnée en l'encodant et la décodant. Un codec peut être un logiciel, matériel ou une combinaison des deux.

### 6.1.3. Compression des fichiers audio

Les fichiers audio sont volumineux ce qui pose des problèmes lors du stockage et d'acheminement dans le réseau, il est nécessaire donc de faire une compression des données audio afin de répondre aux besoins de bande passante, de transmission et de stockage.

Pour compresser un fichier audio, il est utile d'utiliser certaines caractéristiques du son comme le seuil d'audibilité (les sons qui sont inaudibles peuvent être retirés du flux audio) et le masquage fréquentiel (un son peut relèver le seuil d'audibilité au voisinage de sa fréquence, il masque les sons d'intensité les plus faibles).

Les principales normes utilisées pour la compression audio sont : MPEG1, MPEG2 ...etc.

### 6.1.4. Compression des fichiers vidéo

Les fichiers vidéo ont aussi de grande taille ce qui nécessite de les compresser afin de les adapter avec les ressources existantes, tel que la bande passante et l'espace de stockage.

La compression de la vidéo repose sur la réduction de la quantité des données à stocker avec la minimisation de l'impact sur la qualité visuelle. Il existe de nombreuses méthodes de compression de la vidéo comme la diminution de fréquence d'images, diminution de la résolution et élimination des redondances.

Les principales normes utilisées sont : MJPEG, MP4 ...etc.

## 7. Exemple d'architectures de streaming

Il existe plusieurs architectures de streaming media, les plus utilisées sont :

- Advanced Streaming Format, conçu pour le logiciel Windows Media Player.
- Adobe Flash, conçu pour Adobe Flash Player.
- RealMedia conçu pour le logiciel RealPlayer.
- QuickTime, framework de transport de données conçu pour QuickTime Player et lu par plusieurs logiciels.
- SHOUTcast, protocole créé initialement pour Winamp.

Les nouveaux formats utilisant la technologie de streaming adaptatif sont :

- Microsoft Smooth Streaming, conçu pour le greffon Silverlight.
- Apple HTTP Streaming, conçu pour le lecteur QuickTime (à partir de la future version X) et le iPhone OS (à partir de la version 3, déjà disponible).
- Adobe Dynamic Streaming, conçu pour Adobe Flash Player (à partir de la version 10).

## 8. Contraintes et besoins pour le streaming

Un service de streaming doit respecter les contraintes suivantes :

– **Contraintes de débit** : Chaque application de streaming nécessite un débit min pour son bon fonctionnement, on augmentant ce débit la qualité de flux s'améliore.

– **Contraintes d'adaptation** : Pour adapter le flux dans sa totalité, aux conditions du réseau actuel, il faut voir comment pourrions nous jouer sur la vitesse d'encodage de la vidéo pour réduire la taille des paquets afin d'en « buffériser » plus en moins de temps.

– **Contraintes de délai** : Le délai signifie le retard total d'un bout à l'autre (du codeur au décodeur) du signal vidéo transmis, en streaming il faut minimiser le délai le plus possible pour éviter que les applications de streaming soit conversationnel.

– **Contraintes de volume des données** : Les flux audio et vidéo sont en général assez volumineux, ce qui impose un délai de transmission non négligeable. même après compression, les données vidéo demeurent volumineuses.

– **Contraintes liées aux pertes de paquets** : Au cours d'une session de streaming vidéo, l'impact visuel engendré par la perte d'un paquet doit être limité pour fournir une qualité de service acceptable. En pratique, les règles de mise en paquets doivent permettre à l'encodeur de faire face aux pertes de paquets. En considérant que le protocole de transport est assez robuste pour maintenir la synchronisation au décodeur. Toutefois la notion de tolérance aux pertes est très dépendante de l'application et des utilisateurs de l'application. Ainsi, le streaming d'un morceau de musique sera très peu tolérant alors qu'une tolérance supérieure est acceptée pour des applications de visiophonie. [1]

## 9. Méthodes de distribution

Il existe trois méthodes de distribution de flux continu : l'unicast, le multicast et le broadcast.

### 9.1. Diffusion unicast

Cette technique consiste à associer un flux à chaque utilisateur, Elle représente une connexion point à point entre le serveur de streaming et le client. Les éléments actifs du réseau (répéteurs, routeurs, commutateurs,...) dirigent les paquets pour arriver au bon endroit. Elle exige beaucoup de bande passante côté serveur mais elle est souple côté client, cette méthode de distribution est très utilisée pour envoyer un paquet vers un seul destinataire (le cas VIP).

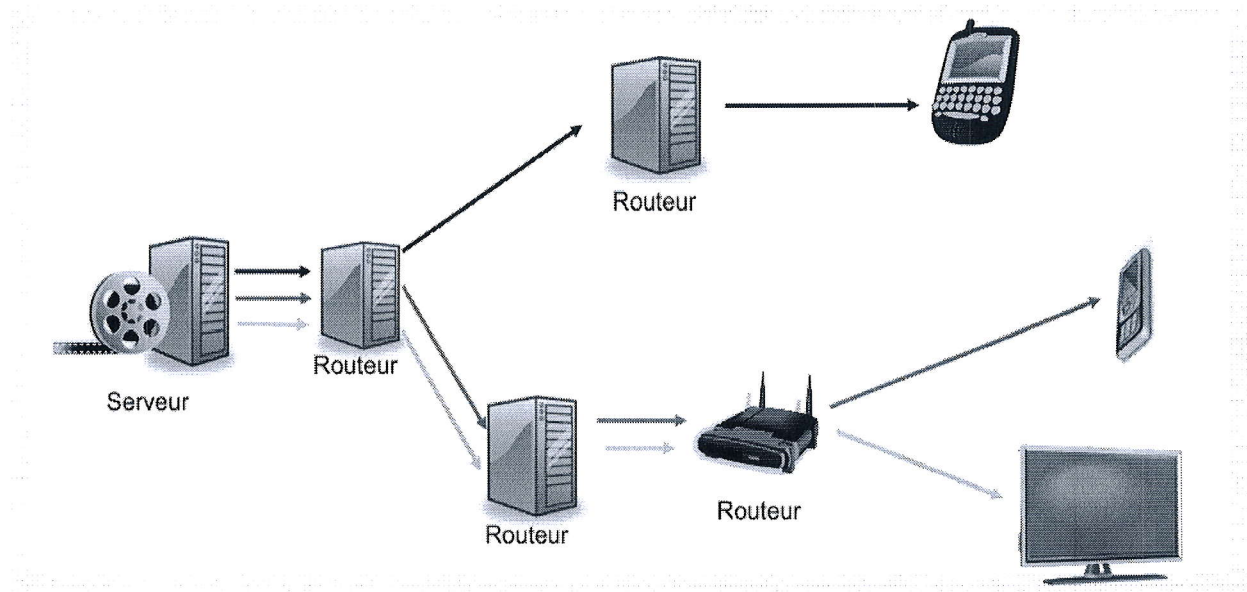


Figure 1-3 : l'unicast.

### 9.2. Diffusion multicast

Cette technique consiste à associer un flux à un groupe d'utilisateurs, un seul flux est émis par le serveur qui est ensuite distribué via des routeurs multicast, vers les membres de groupe, cette technique est plus contraignante pour l'utilisateur car il ne peut pas contrôler le flux qu'il a reçu. Le multicast est souvent utilisé pour la diffusion en direct ou la multi-conférence.

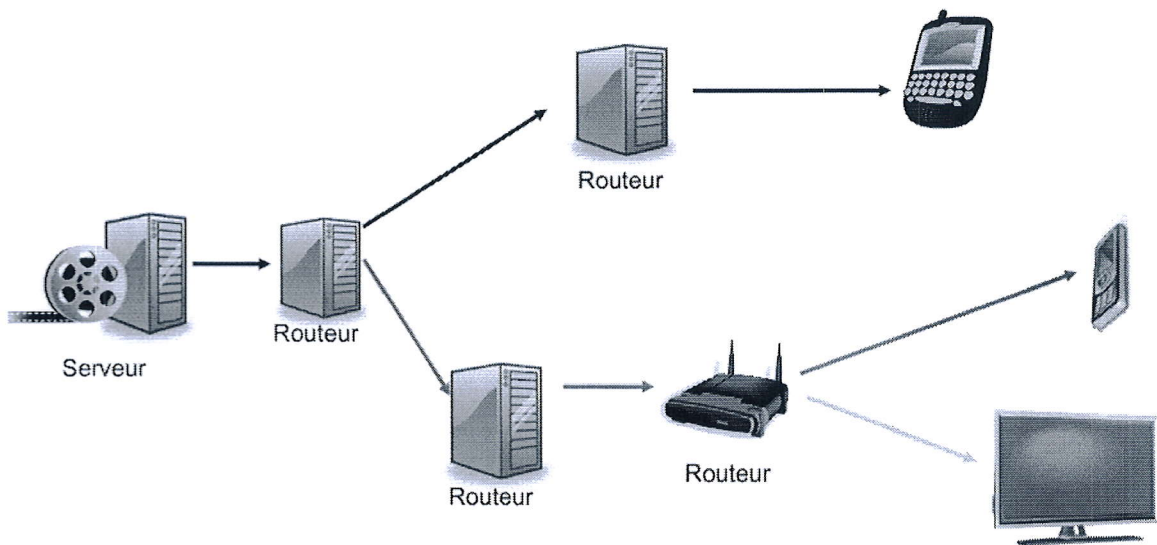


Figure 1-4 : le multicast.

### 9.3. Broadcast

Les flux de broadcast ont une caractéristique particulière : c'est de ne pas pouvoir passer les routeurs puisqu'ils s'adressent uniquement à tous les ordinateurs d'un même sous-réseau.

## 10. Les modes de diffusions

Il existe trois modes d'interaction pour les applications vidéo, et qui définissent les manières d'accéder à l'information dit « Mode PUSH », « Mode PULL » et « Mode HYBRIDE ».

### 10.1. Mode PUSH

Dans ce mode, c'est le serveur qui gère le transfert des données par l'envoi de tous les flux disponibles aux clients.

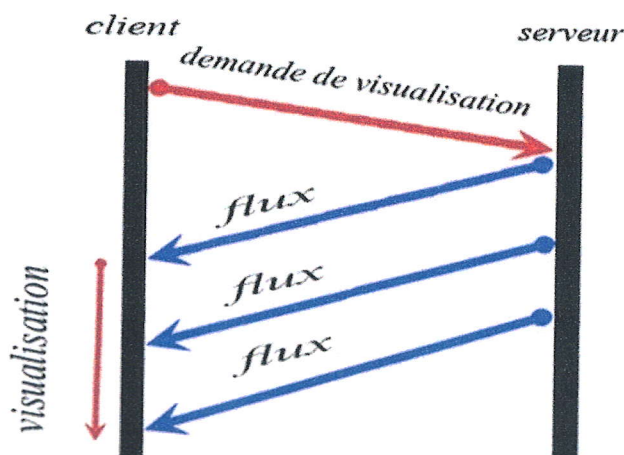


Figure 1-5 : mode PUSH.

Dans ce mode le client ne gère pas le transfert de flux, les segments des vidéos sont poussés vers le client au bon moment. Le mode PUSH est simple à mettre en œuvre et ne nécessite pas un serveur spécifique, mais il consomme beaucoup de bande passante. Il est très utilisé surtout dans des applications de la vidéoconférence, et dans le streaming progressif.

### 10.2. Mode PULL

Dans le mode PULL c'est le client qui gère le transfert de flux (pause, stop, avance et retour rapide, etc.), en demandant les segments qu'il souhaite visualiser.

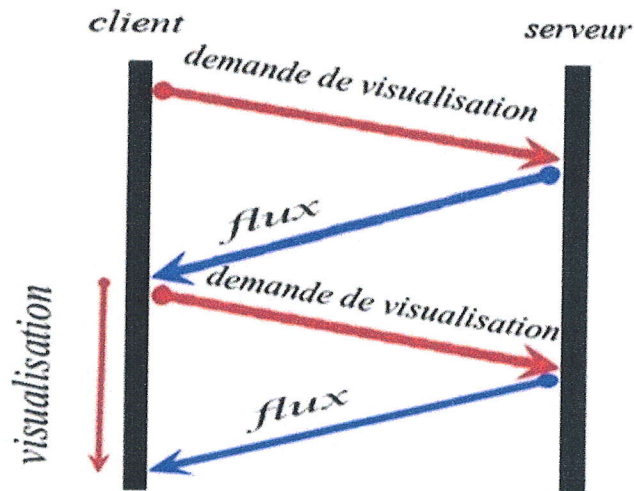


Figure 1-6 : mode PULL.

Ce mode est simple à mettre en œuvre et moins gourmand en ressource (bande passante), il offre aux clients la possibilité de charger immédiatement un bloc de la vidéo souhaitée (ex : youtube , dailymotion... etc.)

### 10.3. Mode HYBRIDE

C'est une combinaison entre les deux modes précédant (PUSH et PULL), dont le serveur et le client partagent la responsabilité de gestion de transfert des flux.

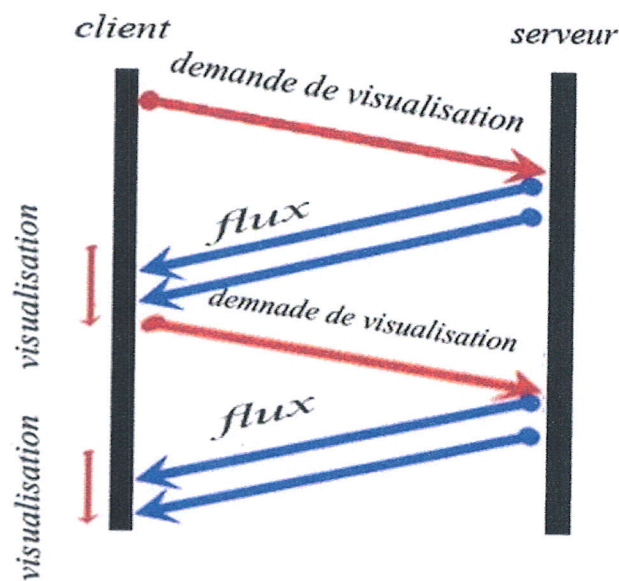


Figure 1-7 : mode hybride.

## 11. Les protocoles dédiés aux streaming

### 11.1. TCP

TCP est un protocole de transport fiable, en mode connecté .Il est orienté connexion, il délivre le flux de données dans le même ordre dans lequel il a été envoyé, après l'établissement d'une session de communication, il permet d'assurer le transfert des données de façon fiable grâce au mécanisme d'acquittement utilisé, et assure que les paquets transmis arrivent dans le même ordre d'émission grâce au numéro de séquence.

TCP contient un mécanisme pour le contrôle de congestion de réseau, aussi il contrôle les erreurs.

Le streaming sur TCP utilise le protocole HTTP, cette technique est largement utilisée sur internet surtout par les systèmes de streaming commerciaux (VOD, live...etc.). Quand un paquet envoyé n'arrive pas dans un temps donné, une solution utilisée est l'attend de l'arrivée de ce paquet (le client stoppe la vidéo en attendant le paquet en suite il continue la lecture), cette stratégie de stopper et attendre est facile à mettre en place. Cependant, elle rend l'expérience de visionnement insatisfaisante.

Malgré les difficultés de cette technique, l'utilisation du TCP pour le streaming fourni une transmission fiable qui élimine la nécessité pour la récupération de la perte à des niveaux supérieurs.

### 11.2. UDP

UDP est un protocole de transport de datagramme non fiable, en mode non connecté documenté dans la RFC 768 de l'IETF.

UDP est orienté datagramme (sans connexion), ce qui évite les problèmes de connexion/déconnexion et ne séquence pas les données.

UDP est plus rapide et simple que TCP. Il ne garantit pas la délivrance du message au destinataire (non fiable), ni l'ordre d'arrivée des paquets. Il ne présente aucun mécanisme de contrôle de congestion ou de contrôle d'erreurs.

Le Streaming sur UDP utilise le protocole RTP (voir section 12) pour faire la diffusion du contenu en temps réel.

### 11.3. Comparaison entre le streaming TCP et le streaming UDP

	TCP	UDP
<b>Utilités</b>	HTTP, VOD	Téléphonie, temps réel
<b>Fiabilité</b>	Fiable	Non fiable
<b>QOS</b>	Médiocre	Bonne

<b>Rapidité</b>	Moins rapide que UDP	Très rapide
<b>Le multicast</b>	Non compatible	Compatible
<b>Temps réel</b>	Impossible pour la téléphonie. Pas d'horodatage	Utilisé avec RTP

**Tableau 1-2 :** Comparaison entre le streaming TCP et le streaming UDP.

#### 11.4. RTP

Le protocole RTP standardisé en 1996 permet la transmission des paquets en temps réel sur le réseau, les paquets sont marqués temporellement de manière à être réordonnés par le client afin d'afficher la vidéo de manière cohérente.

RTP peut être utilisé pour faire l'unicast, le multicast, pour le transport dans une seule direction et pour les applications interactives, il s'appuie sur le protocole UDP plutôt que sur le protocole TCP. RTP fournit très peu de correction d'erreurs (les paquets perdus, en retard ou endommagés sont ignorés).

RTP est conçu pour travailler en conjonction avec le protocole RTCP.

#### 11.5. RTCP

RTCP est basé sur la transmission périodique des paquets de contrôle à tous les participants d'une session.

Il permet de :

- Transmettre des informations de contrôle sur la session.
- Fournir des informations sur la qualité de la session.
- Garder une trace de tous les participants à une session.
- Contrôler le débit auquel les participants à une session RTP transmettent leur paquet RTCP.

Le fonctionnement du RTCP avec RTP permet l'adaptation d'une diffusion en temps réel aux liens et débits différents. Par exemple, si jamais le débit d'un lien baisse ou augmente pendant une transmission, c'est RTCP qui va dire à RTP qu'il peut prendre un autre flux de données avec une qualité adaptée.

#### 11.6. RTSP

RTSP est un protocole de communication de niveau applicatif, permet de contrôler le flux depuis un client (lecture, pause, avance rapide, arrêt...), RTSP ne transporte pas les données elles-mêmes et doit être associé à un protocole de transport.



### 11.7. SRTP

SRTP est un profile de RTP, il est conçu pour sécuriser la multiplication à venir des échanges multimédias sur les réseaux sans fils et filaires.

Les principaux services offerts par SRTP sont :

- Pouvoir rendre confidentielles les données RTP.
- Authentifier et vérifier l'intégrité des paquets RTP.
- La protection contre le rejeu.

### 11.8. HTTP

HTTP est un protocole de la couche 7 du modèle OSI, développé pour le WWW, il est utilisé dans le streaming pour la transmission de flux continu sur TCP.

## 12. Solutions pour faire le streaming

### 12.1. Codage redondant

Cette solution est basée sur la sauvegarde de plusieurs copies d'un même fichier multimédia sur le serveur, chaque copie est encodée avec un format différent. Lorsque une demande de visualisation arrive, le serveur choisit la copie à envoyer selon les caractéristiques de la connexion de client. Cette technique est facile à maitre en œuvre mais gourmande en ressources.

### 12.2. Les réseaux CDN

Les réseaux de diffusion de contenu (CDN) sont les plus utilisés, nécessitent la présence de plusieurs serveurs de streaming au tour du monde, chaque serveur se charge de délivrer les flux à un ensemble des clients d'une région. Cette technique permet de diminuer la charge sur le serveur principale de streaming par la répartition des clients sur un ensemble de serveur mais reste très couteuse.

### 12.3. Streaming vidéo en pair à pair

Le concept principal du calcul pair-à-pair est que chaque pair est un client et serveur en même temps. Dans ce contexte, le contenu multimédia joué par l'utilisateur est mis en commun entre des pairs. Le partage pair-à-pair emploie le mode ouvrir après le téléchargement, mais dans le cas du streaming pair-à-pair vidéo, on lit les données pendant le téléchargement. Un des avantages de streaming pair-à-pair de vidéo est que les pairs ont la liaison directe à d'autres pairs évitant la communication par l'intermédiaire des serveurs de médiation.

### **13. Les champs d'application du streaming**

De nos jours, le streaming est omniprésent sur les différents sites web, il est utilisé dans tout les sites qui proposent des vidéos tel que youtube et dailymotion, il est aussi utilisé dans les jeux en réseau, dans des applications de type vidéo à la demande...etc.

#### **13.1. La vidéo à la demande**

Aussi appelé vidéo sur la demande ou encore vidéo à la séance, est une technique qui permet de commander et de regarder la vidéo en temps réel sur internet. Elle permet aux utilisateurs de commander des films, des émissions de télévision à la demande à l'heure souhaitée. Il existe plusieurs applications de VOD comme Microsoft Netshow, RealVidéo...etc.

#### **13.2. La Vidéoconférence**

La vidéoconférence ou visioconférence est un moyen de communication de plus en plus utilisé par les entreprises qui souhaitent communiquer à distance avec d'autres entreprises ou personnes en évitant de se déplacer. Elle est basée sur le streaming qui permet d'organiser des réunions de travail, des conférences ,formations ou d'autre réunions à distances tous en donnant l'impression d'être tous présent dans la même salle, en transmettant le flux audio et vidéo à temps réel.

#### **13.3. Les jeux en réseau**

Il est possible de nos jours de jouer sans installer les jeux sur les machines, mais en téléchargeant le flux vidéo/audio et en contrôlant le jeu à distant.

Il existe plusieurs jeux en réseaux qui utilisent la technologie de streaming comme : SECONDLIFE, HALFLIFE, DOOM...etc.

### **14. Statut juridique du streaming**

Il convient de distinguer la situation des diffuseurs des contenus multimédia, un internaute qui met en ligne des vidéos, musiques...etc. protégées par des droits d'auteurs sans autorisation peut être suivit juridiquement par les propriétaires de ces œuvres. La diffusion du contenu multimédia n'est possible qu'après autorisation de son auteur, ainsi la mise à disposition en ligne sans autorisation constitue une contrefaçon au titre du code de la propriété intellectuelle susceptible de poursuite pénale et de condamnation à verse des dommages et intérêts au plan civil.

Certain nombres de sites internet diffusant de la musiques (ou vidéo) ont été fermé tels que le site Blogmusik en avril 2007.

## **15. Conclusion**

Dans ce chapitre nous avons étudié la technique de streaming, son architecture, son principe de fonctionnement, les différents modes de distribution et les champs d'applications, ainsi quelques avantages et inconvénients.



*Les réseaux  
de mobiles ad-hoc*

## 1. Introduction

Dans les réseaux mobiles traditionnels, la communication doit être pilotée à un moment ou un autre par des infrastructures fixes telles que les stations de base, les commutateurs...etc. Or il y a des situations dans lesquelles l'installation de tels équipements n'est pas possible pour des considérations de cout et de performance, une solution à ces problèmes est l'utilisation des réseaux de mobiles ad-hoc.

En effet, un réseau de mobiles ad-hoc est un ensemble de terminaux mobiles formant un réseau temporaire et opérant sans l'installation d'infrastructures, par conséquent on peut l'installer rapidement, il ne nécessite aucune administration central pour la gestion et la configuration des stations. Pour communiquer entre eux, les terminaux ont seulement besoin de disposer de logiciels adéquats .Ce type de système convient aux applications militaires pour des missions temporaires, aux rencontres ponctuelles entre professionnels dans le cadre d'un groupe de travail, à la couverture de zone inaccessibles par les réseaux cellulaires traditionnels ...etc.

Dans ce chapitre nous allons définir les réseaux de mobiles ad-hoc, présenter leurs caractéristiques, les domaines d'utilisations, les protocoles de routages et nous parlerons du streaming dans ce type de réseau.

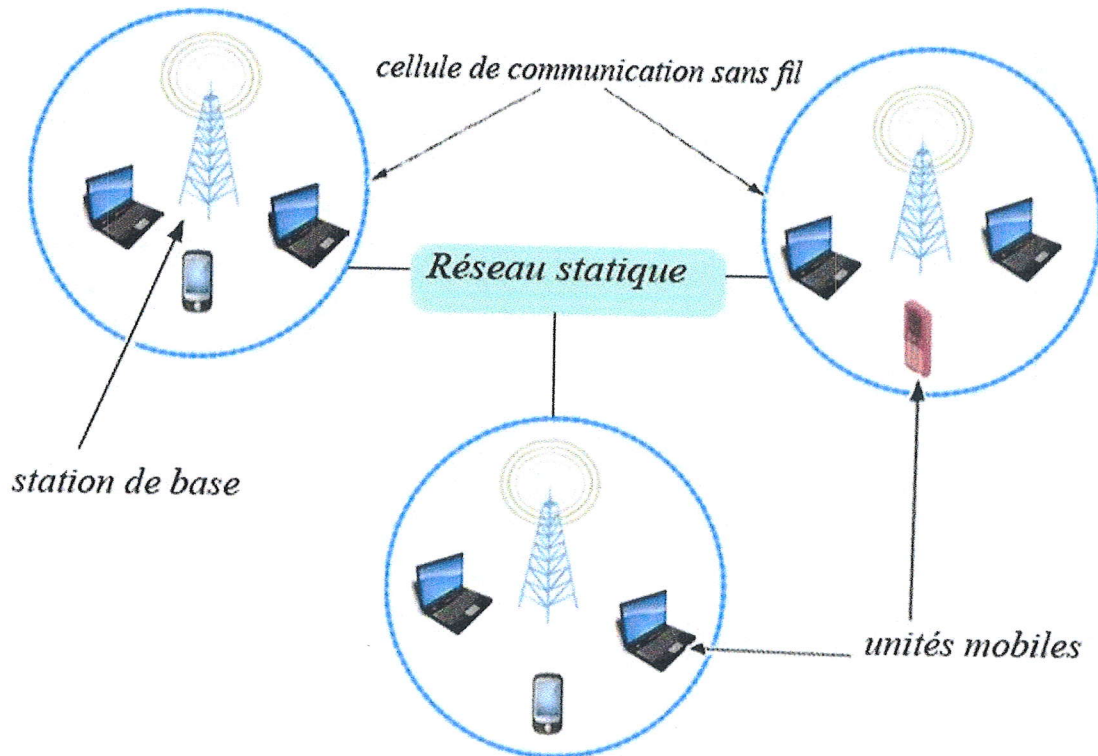
## 2. Les environnements mobiles

Un environnement mobile est un système composé de sites mobiles et qui permet à ses utilisateurs d'accéder à l'information indépendamment de leurs positions géographiques. Il existe deux classes de réseaux mobiles sans fil : réseau mobiles avec infrastructure et sans infrastructure.

### 2.1. Architecture avec infrastructure

Les réseaux mobiles avec infrastructure sont composés de deux ensembles d'entités distinctes : les sites fixes (stations de base) d'un réseau de communication filaire classique (Wired network), et les sites mobiles (Wireless network).

L'unité mobile, dans un instant donné n'est connectée directement qu'à un seul site fixe à travers un réseau sans fil, cette station de base est interconnectée avec les autres stations à travers un réseau filaire, généralement fiable et de débit élevé .Les unités mobiles peuvent communiquer entre eux à travers la station de base à laquelle elles sont directement rattachées. Les unités mobiles sont localisées dans une zone géographique limitée appelée cellule.

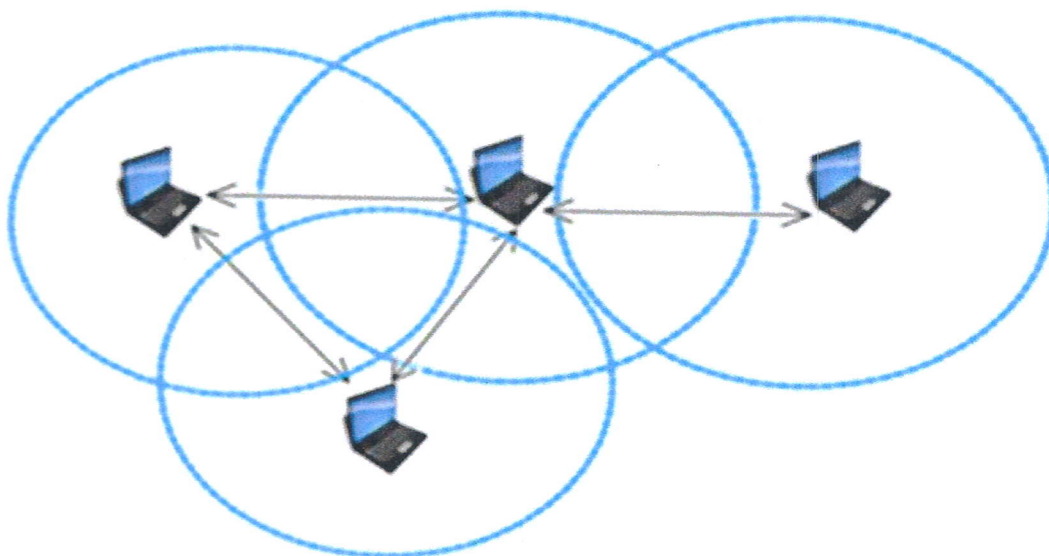


**Figure 2-1 :** Le modèle de réseau mobile avec infrastructure.

Plusieurs systèmes utilisent déjà le modèle de réseau avec infrastructure appelé aussi modèle cellulaire comme les réseaux GSM, UMTS... etc.

## 2.2. Architecture sans infrastructure

Dans les réseaux mobiles sans infrastructure l'entité fixe n'existe pas, tous les sites du réseau sont mobiles et communiquent de manière directe à travers un réseau sans fil.



**Figure 2-2:** Le modèle de réseau mobile sans infrastructure.

Tous les sites du réseau de ce type sont mobiles, et se communiquent d'une manière directe en utilisant leurs interfaces de communication sans fil, donc l'absence de l'infrastructure impose aux nœuds de se comporter comme des routeurs pour acheminer l'information. Ces réseaux, on l'appelant aussi réseaux de mobiles ad-hoc ou MANET.

On s'intéresse dans ce chapitre à ce type de ces réseaux de mobiles, on présentera leurs caractéristiques, domaines d'utilisation, protocole de routage et le streaming dans ces réseaux.

### 3. Définition des réseaux de mobiles ad-hoc

Les réseaux de mobiles ad-hoc (MANET) sont des réseaux d'objets mobiles. Ces objets, nous les appelons aussi équipements ou matériels, peuvent être des assistants personnels, des ordinateurs portables, des téléphones, mais aussi tout type de matériel capable de communiquer sur un réseau sans fil.

Un réseau de mobiles ad-hoc est un réseau créé par une réunion spontanée de mobiles capables de communiquer grâce à leurs interfaces sans fils, sans aucune infrastructure ni configuration préalable. Les nœuds d'un tel réseau coopèrent pour la gestion du réseau.

Un réseau de mobiles ad-hoc se caractérise par sa méthode de découverte des nœuds, son algorithme de routage, et le modèle de mobilité régissant le mouvement des mobiles.

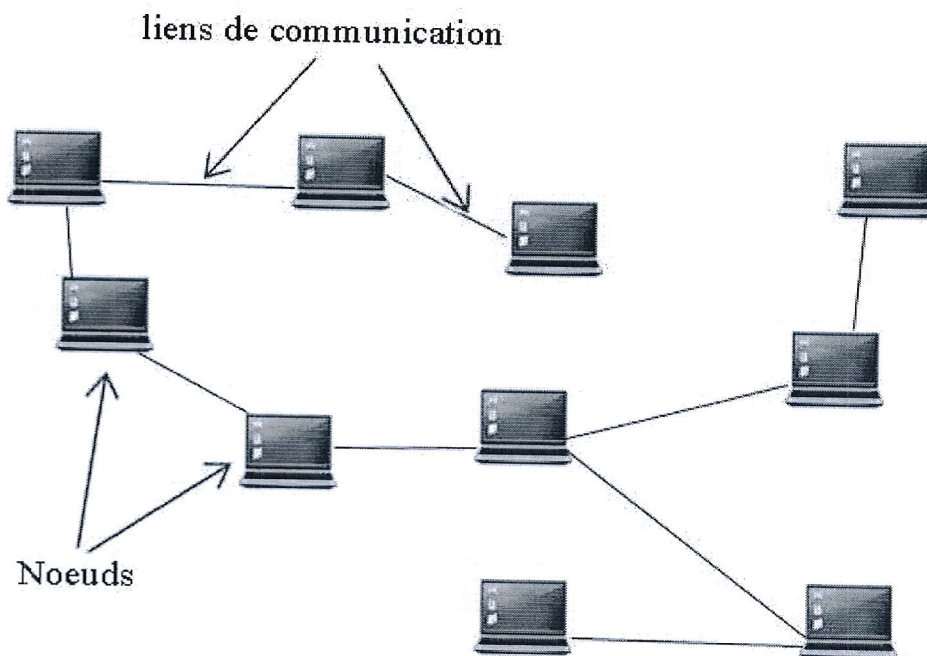


Figure 2-3 : Exemple d'un MANET.

#### 4. Caractéristiques des réseaux de mobiles ad-hoc

Comme tout système de communication, le but d'un réseau de mobiles ad-hoc est de transmettre une information d'une origine donnée vers une destination donnée, le fait que ses composants sont des nœuds autonomes et mobiles (se déplacent librement et arbitrairement) engendre que la topologie du réseau peut changer aléatoirement de façon rapide. Voici quelques autres caractéristiques:

– **L'absence d'infrastructure centralisée** : Les réseaux de mobiles ad-hoc sont des réseaux spontanés, c'est-à-dire que les équipements sont capables de s'organiser en réseau sans aucune configuration initiale. Les équipements constituant le réseau sont mobiles.

– **Une topologie dynamique** : La topologie du réseau n'est pas connue par les participants a priori, elle doit être découverte dynamiquement. Les changements de topologie sont fréquents lors de l'existence du réseau.

– **la contrainte d'énergie** : Les équipements mobiles disposent de batteries limitées ce qui impose une durée de traitement réduite, cela limite les services et les applications supportées par chaque nœud.

– **Une bande passante limitée** : L'utilisation des mediums de communication partagée, ce partage fait que la bande passante réservée pour un hôte soit modeste.

– **L'hétérogénéité des nœuds** : les nœuds peuvent avoir des différences en terme de capacité de traitement (CPU, mémoire) de logiciel et de mobilité (lent, rapide). Dans ce cas, une adaptation dynamique des protocoles s'avère nécessaire pour supporter de telles situations.

– **Sécurité et Vulnérabilité** : Dans les réseaux de mobiles ad-hoc, le principal problème ne se situe pas tant au niveau du support physique mais principalement dans le fait que tous les nœuds sont équivalents et potentiellement nécessaires au fonctionnement du réseau. Les possibilités de s'insérer dans le réseau sont plus grandes, la détection d'une intrusion ou d'un déni de service plus délicate et l'absence de centralisation pose un problème de remontée de l'information de détection d'intrusions [2]

– **Multihops** : Les communications sont multi sauts. Lorsqu'un équipement veut rejoindre un autre qui est hors portée, les messages sont transmis par des équipements intermédiaires. Les équipements jouent un rôle de routeur dans le réseau.



– *Auto-organisé et distribué* : Puisque les réseaux de mobiles ad hoc ne disposent pas de site fixe pour coordonner ou centraliser les échanges, les participants doivent s'auto-organiser à fin de garantir le bon fonctionnement.

– *Temporaire et rapidement déployable* : Ce type de réseau est intrinsèquement temporaire et rapidement déployable. Il n'a pas pour but de remplacer un réseau à infrastructure mais de le compléter ou de le remplacer lorsque nécessaire.

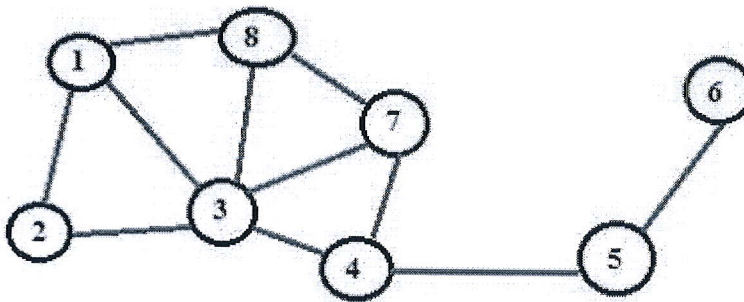
## 5. Modélisation d'un réseau de mobiles ad-hoc

Les réseaux de mobiles ad-hoc sont modélisés par des graphes dynamiques  $G_t = (V_t, E_t)$

$V_t$  : est l'ensemble des objets constituant le réseau.

$E_t$  : modélise les connexions entre ces objets.

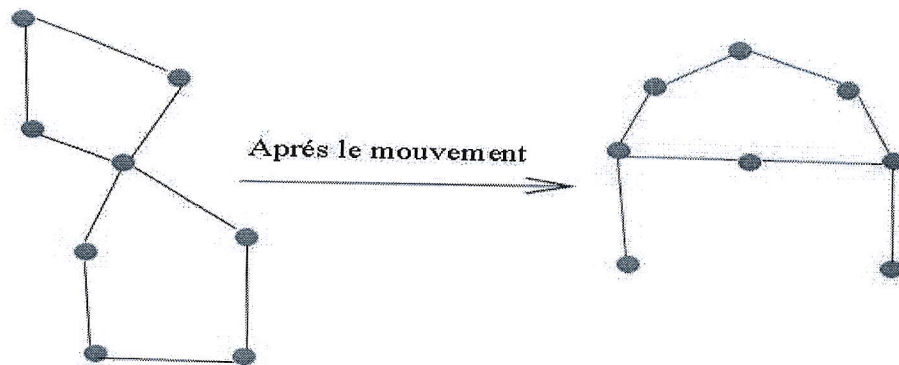
La figure suivante représente un réseau ad-hoc de 8 unités mobiles sous forme d'un graphe :



**Figure 2-4** : La modélisation d'un réseau de mobiles ad-hoc.

La topologie du réseau peut changer à tout moment, elle est donc dynamique et imprévisible ce qui fait que la déconnexion des unités soit très fréquente.

Un exemple de changement de topologie est présenté dans la figure suivante :



**Figure 2-5:** Changement des topologies.

## 6. La communication dans les réseaux de mobiles ad-hoc

Dans les réseaux de mobiles ad-hoc la communication peut être effectuée selon les trois modes (vu dans le premier chapitre : généralités sur le streaming) unicast qui consiste à associer un flux à chaque utilisateur, multicast qui représente une communication de un à plusieurs (mais pas tout le monde) et broadcast ou la diffusion de flux à tous les utilisateurs.

## 7. Problèmes liés aux réseaux de mobiles ad-hoc

L'apparition des réseaux de mobiles ad-hoc a certes facilité la mise en œuvre d'applications mobiles qui ne supportent pas d'infrastructure préexistante mais en revanche, elle a laissé émerger un bon nombre de problèmes dus à la nature de connexion des réseaux sans fil (utilisation des ondes radio lors d'une transmission de données) et la bande passante limitée, qui introduisent des pertes de paquets et des erreurs de transmission donc la communication est moins fiable.

La mobilité arbitraire et la déconnexion fréquente des participants dans le réseau rendent la topologie du réseau non stable (dynamique) en se changeant à des instants imprévisibles d'une manière rapide et aléatoire, en d'autre part les hôtes mobiles sont alimentés par des sources d'énergie autonomes donc restreintes, comme les batteries, par conséquent la durée de traitement est réduite. Donc le paramètre d'énergie doit être pris en considération dans tout contrôle fait par le système. Un autre problème très important est la sécurité de ce type de réseaux cela se justifie entre autres par les vulnérabilités des liens radio aux attaques, ainsi que les contraintes et limitations physiques qui font que le contrôle des données transférées doit être minimisé.

## 8. Domaines d'application des réseaux de mobiles ad-hoc [3]

En plus de leurs utilisations dans les applications tactiques militaires, les réseaux ad-hoc sont utilisés également dans les domaines civils. On peut citer :

- *Les services d'urgence* : opération de recherche et de secours des personnes, tremblement de terre, feux, dans le but de remplacer l'infrastructure filaire.

- *Le travail collaboratif et les communications dans des entreprises ou bâtiments* : dans le cadre d'une réunion ou d'une conférence par exemple.

- *Applications commerciales* : pour un paiement électronique distant (taxi) ou pour l'accès mobile à l'Internet, où service de guide en fonction de la position de l'utilisateur.

- *Réseaux de senseurs* : Les capteurs, chargés de mesurer les propriétés physiques des environnements (comme la température, la pression...), sont dispersés (le plus souvent lâchés d'un avion ou d'un hélicoptère) par centaines, voire par milliers sur le site, effectuent leurs mesures et envoient les résultats à une station par l'intermédiaire d'un routage ad hoc à travers le réseau.

- *Le cadre informatique* : Dans le cadre de l'informatique, les réseaux ad hoc peuvent servir à établir des liens entre ses différents composants. Dans ce cas, on parle non plus de LAN mais de PAN.

## 9. Les protocoles de routage des réseaux de mobiles ad-hoc

Un réseau de mobiles ad-hoc consiste en un ensemble des terminaux sans fil qui se déplacent d'une façon aléatoire, ce déplacement libre produit des changements dans la topologie du réseau ce qui rend difficile de trouver une route entre un nœud d'origine et un nœud de destination. Plusieurs protocoles de routage ont été proposés pour résoudre ce problème, ils sont classés en trois groupes : les protocoles proactifs, les protocoles réactifs et hybride.

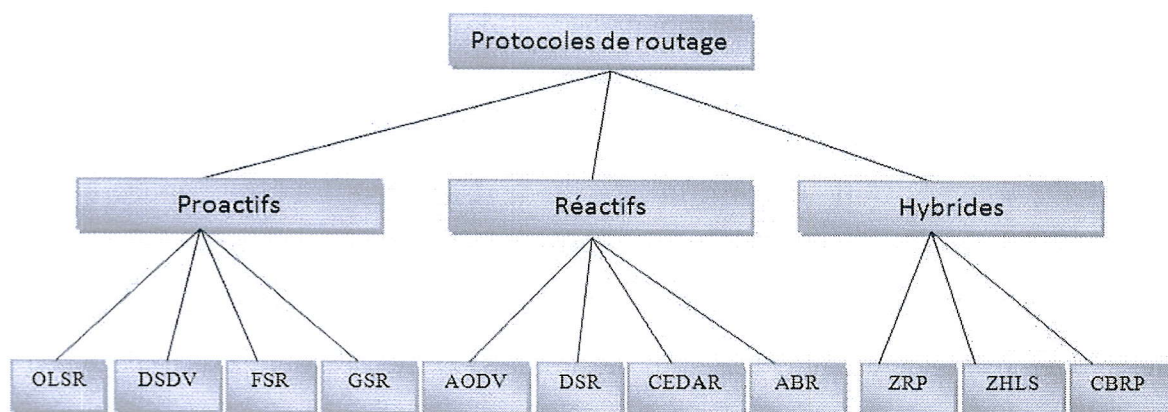


Figure 2-6 : classification des protocoles.

### 9.1. Les protocoles proactifs

Un protocole proactif est un protocole qui construit les tables de routage avant que la demande en soit effectuée. Il identifie en fait à chaque instant la topologie du réseau. [w1]

Les principales méthodes utilisées sont la méthode d'état de lien (Link state) et la méthode de vecteur à distance (distance vector), ces deux méthodes exigent une mise à jour périodique des données de routage qui doivent être diffusées par les différents nœuds de routage du réseau.

Nous présenterons ici les protocoles OLSR, DSDV, FSR et GSR:

#### 9.1.1. OLSR

Le protocole de routage à état de lien optimisé, propose des routes optimales en termes de nombre de sauts dans le réseau en utilisant les relais multipoints (chaque nœud choisit dans ses voisins direct un sous ensemble minimale de nœud qui lui permettent d'atteindre tous ses voisins à deux sauts), Les relais multipoints sont utilisés dans le but de minimiser le trafic dû à la diffusion des messages de contrôle dans les réseaux.

#### 9.1.2. DSDV

Le protocole de routage de vecteur de distance ordonnancé par destination est dérivé d'un algorithme classique de vecteur de distance DBF.

À la demande d'un routage, une table est implantée dans chaque nœud, cette table contient toutes les destinations possibles, le nombre de nœuds nécessaire pour atteindre la destination, le numéro de séquence qui correspond à un nœud destination (pour faire la distinction entre les anciennes et les nouvelles routes, ce qui évite la formation des boucles de routage).

Le DSDV élimine les deux problèmes de boucle de routage "routing loop", et celui du "counting to infinity". Le DSDV utilise une mise à jour périodique et basée sur les événements, ce qui cause un contrôle excessif dans la communication, Le DSDV est lent (une unité mobile doit attendre jusqu'à ce qu'elle reçoive la prochaine mise à jour initiée par la destination, afin de mettre à jour l'entrée associée à cette destination).

#### 9.1.3. GSR

Le protocole de routage GSR est basé sur l'algorithme classique à état de liens, mais il évite le mécanisme inefficace d'inondation des messages de mise à jour. GSR adopte une politique de dissémination des informations de routage identique à celle des protocoles à vecteur de distances. Dans GSR, chaque nœud maintient une table de topologie, une liste de voisins, une table de routage et une table de distances. Pour le calcul des chemins, un algorithme de plus court chemin comme celui de Dijkstra est utilisé mais modifié afin de

construire la table de routage et la table de distance en parallèle avec le calcul des plus courts chemins.[6]

#### 9.1.4. FSR

Le protocole de routage FSR est en réalité une optimisation du protocole GSR. FSR est basé sur la technique nommée « *œil de poisson* » proposée pour réduire la taille des informations dans la représentation des données graphiques. Pour réaliser cette technique, FSR introduit la notion de portée. La portée d'un nœud  $i$  est l'ensemble des nœuds qui peuvent être atteints en passant par  $h$  sauts à partir de  $i$ . Un nœud dans FSR effectue des mises à jour pour les nœuds qui résident dans sa portée plus fréquemment que pour les nœuds qui n'appartiennent pas à sa portée. Cette technique permet de réduire considérablement la charge de routage mais au prix d'utilisation de chemins non optimaux [6].

### 9.2. Les protocoles réactifs

Un protocole réactif est un protocole qui construit une table de routage lorsqu'un nœud en effectue la demande. Il ne connaît pas la topologie du réseau, il détermine le chemin à prendre pour accéder à un nœud du réseau lorsqu'on lui demande [w1].

Donc contrairement aux protocoles proactifs qui, pour un nœud donné, calculent automatiquement les meilleurs chemins pour toutes les destinations possibles, les protocoles actifs ou à la demande, calculent les routes seulement lorsque le besoin se présente. La plupart des protocoles de routage dans les réseaux de mobiles ad-hoc sont réactifs, l'avantage major présenté est la réduction considérable de flot de paquets traversant le réseau.

D'autre part ces protocoles sont considérés comme des protocoles lents a cause de la recherche des chemins qui peut nuire à la performance de réseau dans certains cas (ex: des applications interactives).

Parmi les protocoles de routage réactif nous présentons AODV, DSR, CEDAR et ABR :

#### 9.2.1. AODV

Le protocole AODV est une amélioration de DSDV, les routes sont créées lorsque la demande est effectuée ce qui permet de réduire le nombre de messages diffusés.

AODV utilise également la numérotation de séquence qui permet aux nœuds d'utiliser les routes les plus récentes. Le protocole AODV ne présente pas de boucle de routage, en outre, il évite le problème de comptage à l'infini de l'algorithme de Bellman-Ford, ce qui offre une convergence rapide quand la topologie du réseau ad-hoc se change.

### 9.2.2. DSR

Le protocole DSR est fondé sur le principe du routage par la source, les nœuds n'ont pas besoin de tables de routage. Dans chaque paquet transmis, il y a un champ qui contient la séquence de nœud à suivre pour atteindre la destination.

Selon cette technique, le nœud origine doit connaître la route qu'il va suivre, c'est à dire la séquence de nœuds intermédiaires par lesquels l'information qu'il souhaite transmettre va transiter, avant de l'acheminer. Ainsi lorsqu'un nœud veut envoyer un message, il diffuse dans le réseau un paquet appelé « *route request* » qui contient un champ permettant d'enregistrer tous les nœuds qu'il va visiter, le processus de diffusion se poursuit jusqu'à ce que le paquet en question atteigne le nœud destinataire.

### 9.2.3. CEDAR

CEDAR est un protocole de routage réactif avec QoS. CEDAR repose dans son fonctionnement sur un ensemble de nœuds dominants minimaux appelés « cœur du réseau ». La communication entre les nœuds appartenant au cœur du réseau s'effectue via des échanges unicast au lieu du broadcast pour éviter les pertes dues aux problèmes des stations cachées et exposées.[6]

### 9.2.4. ABR

ABR est un protocole de routage réactif qui repose sur le routage source. ABR classe les liens comme stables ou non stables selon l'âge de ces derniers. Chaque nœud détermine l'âge des liens avec ses voisins en se basant sur le nombre des beacons<sup>1</sup> (indice d'associativité) qu'il reçoit périodiquement de leurs parts. Si le nombre de ses beacons est supérieur d'un certain seuil, le lien est considéré stable. La phase de découverte de chemins dans ABR est semblable à celle dans DSR. Avant d'envoyer un paquet RREQ, un nœud doit d'abord inclure les indices d'associativité concernant tous ses voisins et son adresse dans le paquet RREQ. Un nœud qui reçoit un paquet RREQ retire tous les autres indices mis par son voisin et ne garde que celui qui lui concerne. De cette façon, quand un paquet RREQ arrive à destination il ne contiendra que les adresses des nœuds par lesquels il a passé ainsi que les indices d'associativité de chaque lien du chemin. La destination choisie parmi tous les chemins possibles celui constitué de liens les plus stables (avec des indices d'associativité les plus élevés), et envoie la réponse au nœud source. Dans ABR, un nœud qui détecte la rupture d'un lien initie localement une découverte de chemins.[6]

---

<sup>1</sup> Un beacon est un message de la couche liaison envoyé périodiquement, utilisé pour maintenir la connectivité

L'optimisation de route est considérée d'importance secondaire, et les plus longues routes sont souvent utilisées pour éviter le temps de découvrir de nouvelles routes. Le protocole exécute trois fonctions: création des routes, maintenance des routes et suppression des routes.

### 9.3. Les protocoles hybrides

Les protocoles hybrides combinent les approches réactives et proactives. Le principe est de construire les tables de routage avant que la demande ne soit effectuée dans son voisinage à une certaine distance, alors qu'il construit une table de routage lorsqu'un nœud effectue la demande à l'extérieur. Parmi les protocoles de routage hybrides :

#### 9.3.1. ZRP

Le protocole ZRP est un protocole de routage hybride (proactif/réactif). Le routage réactif se limite à la zone déterminée par le nombre de sauts. Le routage proactif sera limité aux voisins qui se trouvent jusqu'à deux sauts.

#### 9.3.2. ZHLS

Le protocole ZHLS est basé sur la décomposition d'un réseau en zone, la topologie d'un réseau est ainsi partagée en deux niveaux :

Un niveau nœud indique la façon dont les nœuds d'une zone sont connectés entre eux physiquement, un lien virtuel peut exister entre deux zones s'il y a au moins un nœud d'une autre zone, et un niveau zone qui renseigne sur le schéma de connexion des différentes zones.

Ces différents niveaux entraînent donc deux différents types de liens : les liens « *inter-nœuds* » et les liens « *inter-zones* ».

#### 9.3.3. CBRP

Dans le "Protocole de Routage Basé sur les Groupes" appelé CBRP, l'ensemble des nœuds du réseau est décomposé en groupes. Le principe de formation des groupes est le suivant : Un nœud  $p$  qui n'a pas de statut (i.e. qui n'est ni membre ni représentant de groupe), active un timer et diffuse un message « *Hello* ». Lorsqu'un représentant de groupe reçoit le message « *Hello* », il envoie immédiatement une réponse à l'émetteur. Lors de la réception de réponse, le nœud  $p$  change son état « *indécis* » à l'état « *membre* ». Si  $p$  dépasse un certain timeout en attendant la réponse et dans le cas où il possède un lien bidirectionnel vers au moins un nœud voisin, il considère lui-même un représentant de groupe. Dans le cas contraire,  $p$  reste dans l'état « *indécis* » et il répète la même procédure. [w3]

#### 9.4. Les avantages et les inconvénients des protocoles de routages des réseaux de mobiles ad-hoc

Les protocoles proactifs permettent le maintien d'une table de routage à jour par l'échange périodique des messages. Ces tables étant à jour, l'envoi de ces messages se fait rapidement. Cependant, on ne peut nier que l'émission régulière de ces paquets occupe une partie de la bande passante, qui risque d'augmenter en fonction du nom, le nombre de nœuds présents sur le réseau.

Les protocoles réactifs sont, comme expliqués précédemment, basés sur une construction du réseau à la demande. Leur avantage se trouve donc dans le fait qu'il n'y a pas de surconsommation de bande passante comme pour les protocoles proactifs. En revanche, on peut s'inquiéter du délai nécessaire, avant l'envoi d'un message, pour trouver la route.

Les protocoles hybrides ont les avantages des deux approches précédentes en utilisant une notion de découpage du réseau. Cependant, il rassemble toujours quelques inconvénients des deux approches proactives et réactives. [3]

### 10. Le streaming vidéo sur les réseaux de mobiles ad-hoc

Partant du fait que les applications multimédias sont de plus en plus présentes dans la vie nomade, nous pouvons facilement imaginer un scénario où les utilisateurs mobiles construisent leur propre réseau, spontanément, sans recourir à l'infrastructure, dans le but d'échanger des vidéos entre eux. Juste en s'appuyant sur les interfaces sans fils incorporées dans leurs mobiles. Comme ces appareils sont limités en termes de mémoire, ils ne peuvent pas stocker l'ensemble des flux vidéo, et seraient, sans doute, amenés à ne garder qu'une partie de celui-ci. Autrement, les performances de leurs mémoires diminuent. Considéré comme trop semblable aux topologies paires à paires, par un comportement similaire: les arrivées et les départs des nœuds, les réseaux de mobiles ad-hoc sont construits dans la plupart du temps par des appareils nomades avec un protocole spécifique (proactif, réactif ou hybride) pour router les données entre eux. Indépendamment du protocole de routage, les nœuds mobiles ont besoin d'une certaine quantité de mémoire de traitement comme un buffer pour faire face aux problèmes de rupture du flux vidéo en raison de leur mouvement.

L'architecture de streaming dans les réseaux de mobiles ad-hoc est proche d'un modèle client/serveur mais chaque nœud joue le rôle d'un fournisseur, consommateur ou routeur.



## 11. Les problèmes de streaming dans les réseaux de mobiles ad-hoc

La communication sur un canal sans fil n'est tout simplement pas en mesure d'atteindre la même qualité (débit, taux d'erreur,... etc.), assurée par son homologue filaire, ce qui réduit la qualité du contenu multimédia livré. En plus que ça, dans un environnement mobile, les conditions du canal peuvent changer rapidement en raison de la variation de la distance entre les stations (la mobilité des utilisateurs). D'autre part la problématique du maintien de la qualité de service des sessions multimédia lors de la mobilité des utilisateurs représente un grand défi.

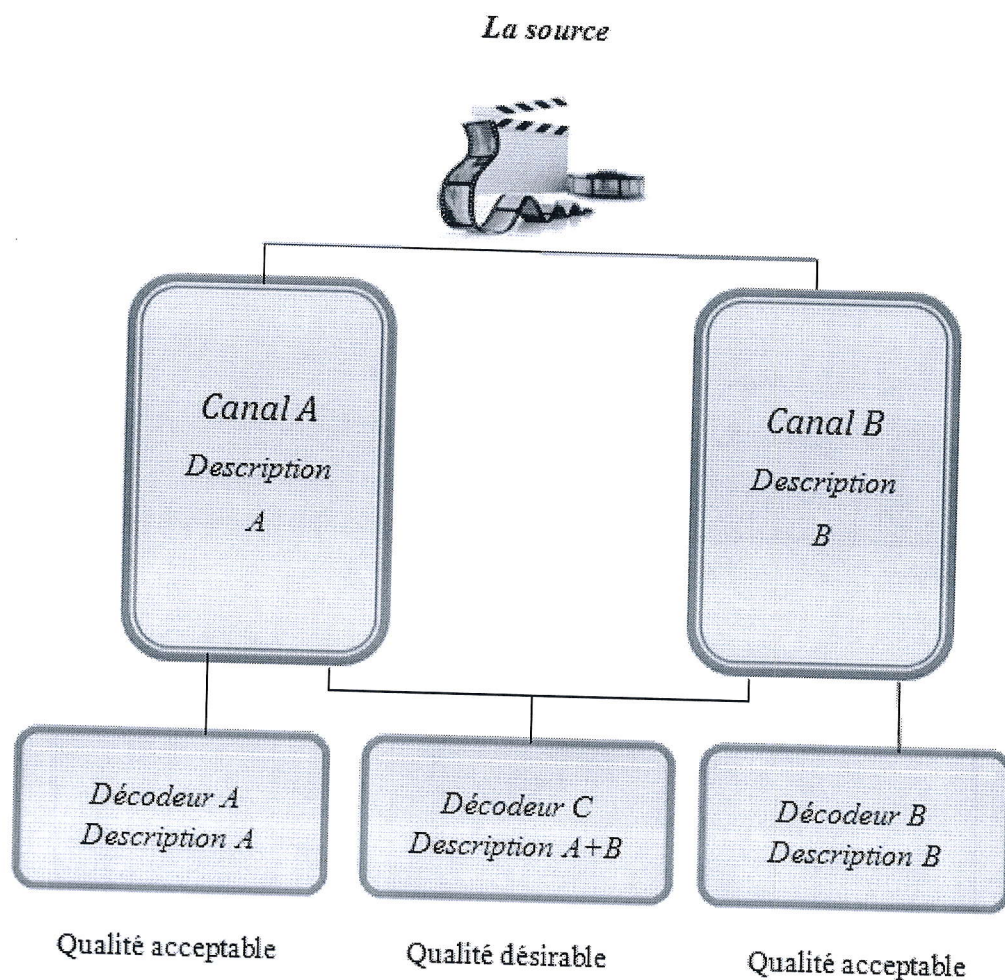
## 12. Les techniques de streaming dans les réseaux de mobiles ad-hoc

Les réseaux de mobiles ad-hoc sont composés d'un ensemble de nœuds mobiles indépendants qui communiquent dans leurs rayons de portée d'émission/réception. Les nœuds sont libres de se déplacer arbitrairement, ce qui caractérise la topologie dynamique du réseau, ainsi ils sont autonomes à leurs fonctionnements et à leurs participations à l'acheminement des informations du réseau. L'absence de l'infrastructure fixe et de l'administration centrale, élève le taux des pertes de différentes natures, et les caractéristiques relatives aux réseaux sans fil ad-hoc vont influencer la qualité de la vidéo reçue.

Afin de fournir un service de communication visuelle robuste et fiable, des techniques sont mises en œuvre. Il existe plusieurs techniques de streaming dans les réseaux de mobiles ad-hoc, parmi ces techniques :

### 12.1. Codage par description multiple

Cette technique appelée aussi MDC (Multiple Description Coding) est liée au codage conjoint source-canal, elle consiste à introduire de la redondance au niveau de la source pour gérer les paquets, elle vise à fragmenter le flux média en deux ou plusieurs sous-flux et les coder, ces sous-flux sont appelés descriptions, ils sont transmis sur des canaux indépendants, toute description peut être utilisée, cependant la qualité s'améliore par le nombre de descriptions reçues en parallèle.



**Figure 2-7 :** exemple MDC.

Dans ce schéma, un flux de données d'entrée est transformé en deux différentes descriptions indépendantes et envoyées sur différents canaux. Au niveau du récepteur si toutes les descriptions sont reçues correctement, les données d'origine seront reconstruites avec précision. Mais, dans le cas où certaines des descriptions ne parviennent pas à atteindre la destination, en raison de défaillance du canal, le reste des descriptions sont utilisées pour trouver une estimation des données d'origine (plus le nombre de descriptions reçues est grand, plus la qualité du signal reconstruit est élevée).

MDC peut traiter efficacement la perte de paquets sans avoir besoin de retransmission, il peut donc répondre aux exigences du réseau en améliorant la résistance aux erreurs de transmission sur les réseaux de mobiles ad-hoc grâce à la diversité des chemins.

## 12.2. Routage multi chemins

Un routage à chemins multiples consiste à considérer non plus une seule route entre une source et une destination pour la transmission de données, mais un ensemble de chemins. Dans le streaming vidéo l'objectif principal est de reconnaître les routes garantissant que la

presque obligatoire. En effet, la technique la plus utilisée pour la réalisation du vidéo streaming sur MANET est l'approche inter-couche.

### 13. La qualité de service

Dans les réseaux de télécommunication, l'objectif de la qualité de service est d'atteindre un meilleur comportement de la communication, pour que le contenu de cette dernière soit correctement acheminé, et les ressources du réseau soient utilisées d'une façon optimale. La qualité de service peut être définie comme le degré de satisfaction d'un utilisateur des services fournis par un système de communication .

La RFC 2386 caractérise la qualité de service comme un ensemble de besoins à assurer par le réseau pour le transport d'un trafic d'une source à une destination. Ces besoins peuvent être traduits en un ensemble d'attributs pré-spécifiés et mesurables en terme de :

- Délai de bout en bout.
- Variance de délai (gigue).
- Bande passante.
- Pertes de paquets.

La qualité de service est un point critique dans les systèmes de streaming dans les réseaux de mobiles ad hoc du fait que la mobilité et la nature des liens entre les terminaux (sans fil, la bande passante) peut affecter la bonne délivrance des séquences vidéo (décalage, blocage...).

### 14. Conclusion

Les réseaux de mobiles ad-hoc présentent de nombreuses caractéristiques intéressantes, le fait qu'ils sont auto-organisés et facilement déployables rendre cette technologie très puissante et très utilisée dans tous les domaines qui exigent la présence d'un réseau sans fil avec des nœuds capables de gérer et d'assurer son bon fonctionnement.

Dans ce chapitre nous avons présenter l'essentiel des réseaux de mobiles ad-hoc , leurs fonctionnement ,les protocoles de routage ,les domaines d'utilisation et les caractéristiques les plus importantes.



*La simulation avec*

**OMNET++**

## 1. Introduction

La communication sans fil a suscité un intérêt considérable dans la communauté de la recherche. De nombreux réseaux sans fil sont évalués en utilisant des simulateurs d'événements discrets comme OMNET++. Bien que OMNET++ fournisse un cadre de simulation puissant et clair, il manque de soutien direct et d'une chaîne de modélisation concise pour la communication sans fil, les deux sont fournis par MIXIM et INET. Ils rejoignent et prolongent plusieurs cadres de simulation existants développés pour les simulations sans fil et mobiles dans OMNET++. Ils fournissent des modèles détaillés de canal sans fil et de connectivité sans fil, des modèles de mobilité, des modèles pour les obstacles et de nombreux protocoles de communication en particulier au niveau de contrôle d'accès au support (MAC). En outre, il fournit une représentation graphique conviviale pour les réseaux sans fils et les réseaux mobiles en OMNET ++, en soutenant le débogage et la définition des scénarios sans fils, même complexes. Dans ce chapitre nous allons présenter plusieurs simulateurs de réseaux et les outils qu'on a utilisé pour le développement de notre application qui sont OMNET++ et INET

## 2. Généralités

De nos jours, la simulation connaît un essor considérable et ce grâce à l'intérêt que présente les modèles informatiques des systèmes simulés.

### 2.1. La simulation

La simulation consiste à la modélisation informatique d'un système quelconque, en offrant une représentation de toutes les entités de ce système, leurs comportements propres, ainsi que leurs interactions .elle met à la disposition de l'utilisateur un environnement d'expérimentation dont on peut faire varier les paramètres [4].

## 3. Types de simulation

Il existe deux types de systèmes de simulation :

### 3.1. Systèmes de simulation continus

Ce sont des systèmes pour lesquels les variables peuvent changer d'état à n'importe quel instant pendant la simulation.

### 3.2. Systèmes de simulation discrets

Ce sont des systèmes pour lesquels une simulation est une modélisation informatique où le changement de l'état d'un système, au cours du temps, est une suite d'événements discrets.

## 4. Les outils de simulation réseau

Il existe plusieurs simulateurs de réseau tel que : NS2, OMNET++, OPNET, GLOMOSIM, JSIM...etc. Parmi ceux, on va présenter quelques simulateurs comme NS2, OPNET, GLOMOSIM et le simulateur OMNET++.

### 4.1. NS2

NS2 est un simulateur à événements discrets très répandu dans le domaine de la recherche pour tout ce qui se réfère aux réseaux. Il est utilisé sur la plate forme Unix, Microsoft Windows. Sa licence est gratuite.

Il présente un support important pour la simulation des différents protocoles de transport (TCP, UDP), et les protocoles de routage. Le développement de NS2 suit une approche orientée objet qui utilise deux langages de programmation C++ et TCL.

### 4.2. OPNET++ [5]

L'outil de simulation OPNET permet de simuler les infrastructures des réseaux puissants, codé en C++ et conçu en orienté objet. Il permet aussi de concevoir et d'étudier des réseaux de communications, on se base sur ce simulateur surtout pour le réseau des capteurs sans fil, les nouvelles technologies...etc. Mais il est long lors de sont apprentissage. Il est composé de plusieurs éditeurs : le module « *nœuds* », le module « *process* » et le module « *simulation* » (exécution de la simulation qui apportera des changements aux nœuds).

OPNET englobe plusieurs fonctions propriétaires. Il gère aussi certains types d'objets tels que les liens et les formats de paquets.

OPNET est un simulateur assez complexe ce qui résulte les difficultés d'apprentissage. Il est basé sur deux méthodes qui sont l'utilisation des nœuds programmés et définition des modèles.

### 4.3. GIOMOSIM [4]

GLOMOSIM a été conçu selon une architecture orientée couche similaire que celle des sept couches OSI pour les réseaux .Il a été développé au laboratoire UCLA Parallel Computing Laboratory en utilisant le langage PARSEC<sup>1</sup>.

Ils ont été développés à partir du langage C à l'université UCLA (University California Los Angeles).

---

<sup>1</sup> PARSEC est un langage de programmation dérivé du langage MAISIE. Tous deux sont destinés à la simulation à événements discrets.

#### 4.4. OMNET++ [4]

OMNET++ est basé sur la plateforme Eclipse, c'est un environnement open source qui fournit des outils pour la création et la configuration des modèles de réseaux, et des outils pour l'exécution des programmes ainsi que pour l'analyse des résultats de simulation.

#### 5. Comparaison entre les simulateurs

Simulateurs	Avantages	Inconvénients
<b>NS2</b>	<ul style="list-style-type: none"> <li>- orienté objet.</li> <li>- simulation des protocoles standards.</li> <li>- plusieurs protocoles disponibles.</li> <li>- simulateur multicouches.</li> </ul>	<ul style="list-style-type: none"> <li>- Conçu pour les réseaux filaires</li> <li>- Difficulté d'ajout de nouveau modèle à cause des dépendances entre modules</li> <li>- Intégration difficile à d'autres applications</li> <li>- La faible performance des simulations de réseaux importants</li> <li>- Scénario de simulation décrit on OTCL</li> </ul>
<b>OPNET</b>	<ul style="list-style-type: none"> <li>- Concevoir et étudier des réseaux de communications, des nouvelles technologies, des protocoles et des applications avec facilité et évolutivité</li> </ul>	<ul style="list-style-type: none"> <li>- Apprentissage long.</li> </ul>
<b>GLOMOSIM</b>	<ul style="list-style-type: none"> <li>- Initialement conçu pour les réseaux sans fil (surtout les réseaux Ad Hoc).</li> <li>- Exécution des simulations relativement rapide.</li> <li>- Environnement de simulation parallélisable.</li> </ul>	<ul style="list-style-type: none"> <li>- Nécessité de maîtriser le parsec pour toute personnalisation autour du noyau.</li> <li>- Installation assez complexe.</li> </ul>
<b>OMNET++</b>	<ul style="list-style-type: none"> <li>- Architecture modulaire permettant l'intégration de nouveaux modèles.</li> <li>- Utilisation du C++ (et récemment du c#) pour le développement des noyaux.</li> <li>- Les classes de base du simulateur</li> </ul>	<ul style="list-style-type: none"> <li>- Ne prend pas en charge le cas des réseaux de capteurs.</li> <li>- Peu de modèles pour les réseaux sans fils.</li> <li>- Description des modèles en langage.</li> </ul>

	peuvent être étendues et personnalisées	
	- Conception de modèles se rapprochant de la réalité.	

**Tableau 3-1** : Comparaison entre les simulateurs.

## 6. Le simulateur OMNET++

### 6.1. Présentation d'OMNET++

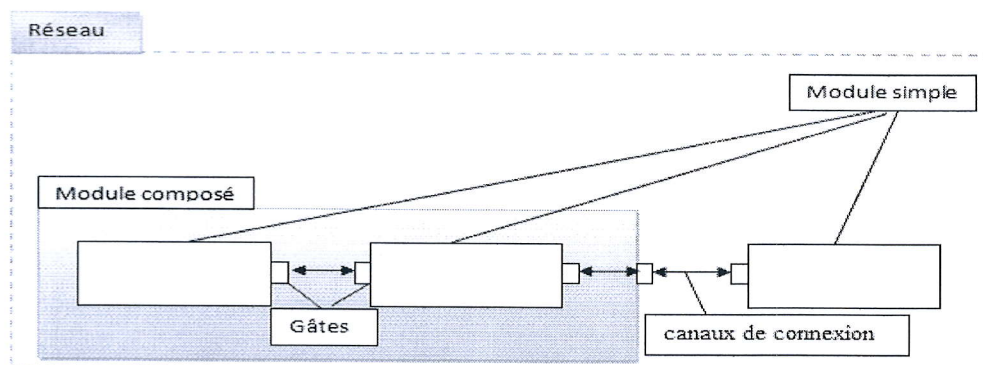
OMNET++ est un projet open source dont le développement a commencé en 1992 par Andras Vargas à l'université de Budapest .Il est utilisé pour la simulation des événements des réseaux discrets. OMNET++ représente une bibliothèque écrite en C++ permettant de faire des simulations non seulement des réseaux filaires et sans fils mais aussi la modélisation des protocoles de communications et des systèmes reparties, les architectures hardware...etc. En général il peut être utilisé pour n'importe quel système à événement discrets avec des entités communiquant par envoie des messages.

Parmi ses fonctionnalités le soutien pour les réseaux de capteurs, les réseaux ad-hoc sans fil, des protocoles internet, la modélisation des performances.

### 6.2. Description architecturale d'OMNET++

Un modèle OMNET++ se compose de modules hiérarchiquement imbriqués qui communiquent par l'envoi de messages à travers des ports (gâtes).Il ya deux types de modules : simples et composés, les modules simples ce sont les modules au niveau le plus bas de la hiérarchie du module appelés modules actifs, ils sont écrits en C++ et les modules composés sont des modules qui contiennent des sous-modules simples. Un module composé est constitué de simples modules ou d'autres modules composés connectés entre eux.

Le nombre de niveaux hiérarchiques est illimité. Le modèle entier, appelé réseau OMNET ++, est lui-même un module composé.



**Figure3-1** : module composé et simple.



Les modules se communiquent entre eux par envoi des messages, ces messages sont envoyés à travers des ports (gâtes) qui représentent les portes (ou interfaces) de sortie (output) et d'entrée (input), ces deux interfaces peuvent être reliées par des canaux de connexion.

Une connexion peut avoir (ou non) plusieurs paramètres comme : le délai de propagation, le taux d'erreur... etc.

### 6.3. Les fichiers principaux d'OMNET++

La conception d'un réseau se fait dans un fichier *.ned*, les différents paramètres de chaque module sont spécifiés dans un fichier *.ini*. OMNET++ génère à la fin de chaque simulation deux nouveaux fichiers *omnet.vec* et *omnet.sca* qui permettent de tracer des courbes et calculer des statistiques.

#### 6.3.1. Fichier *.ned*

L'OMNET++ utilise NED qui est un langage de description de la topologie du réseau pour décrire la structure des modules avec les paramètres (ex les ports, le nombre de nœuds, les adresses... etc). Un fichier *.ned* peut être écrit dans n'importe quel éditeur de texte en respectant ses règles d'écriture. OMNET++ fournit un excellent support pour les deux sens graphique et texte.

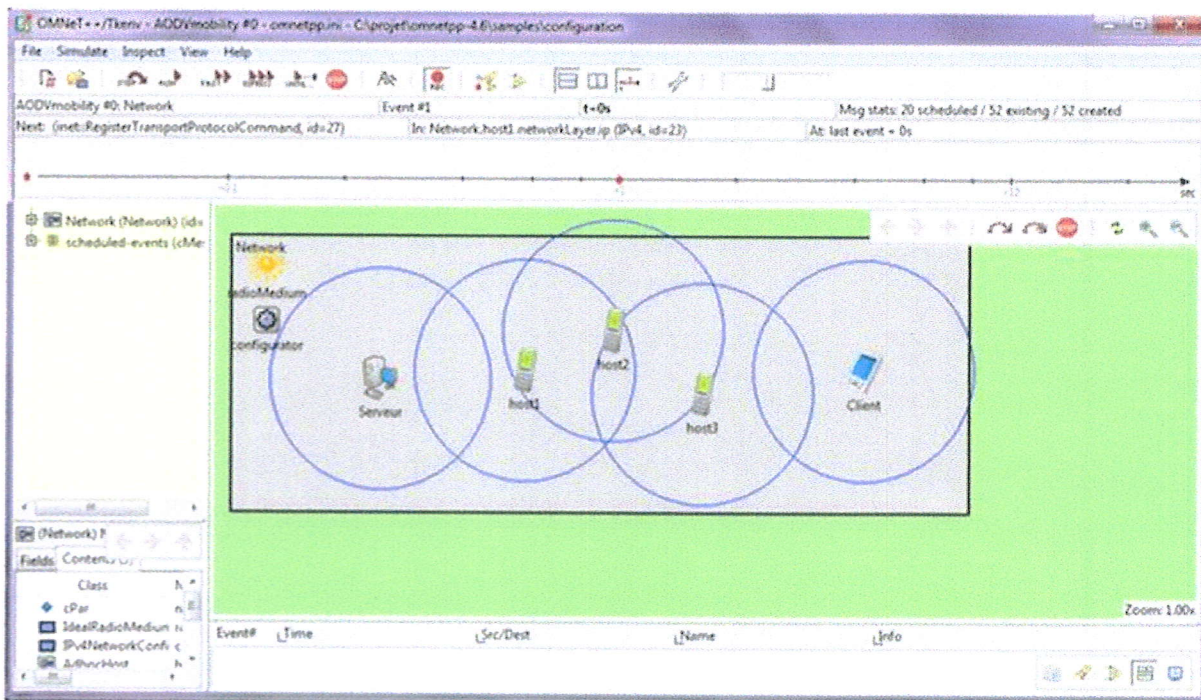


Figure 3-2 : fichier *.ned* en mode graphique.

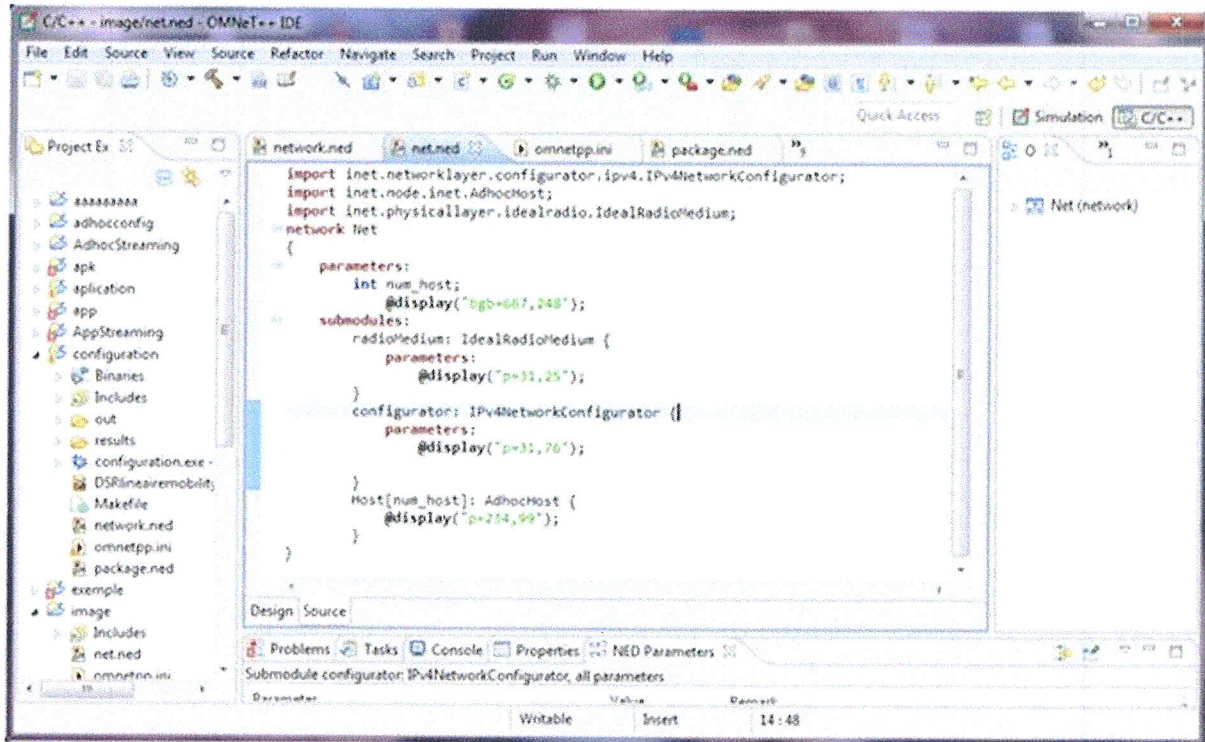


Figure 3-3: fichier .ned en mode texte.

### 6.3.2. Fichier.ini

Le fichier *.ini* permet d'initialiser les paramètres des différents modules ainsi la topologie du réseau.

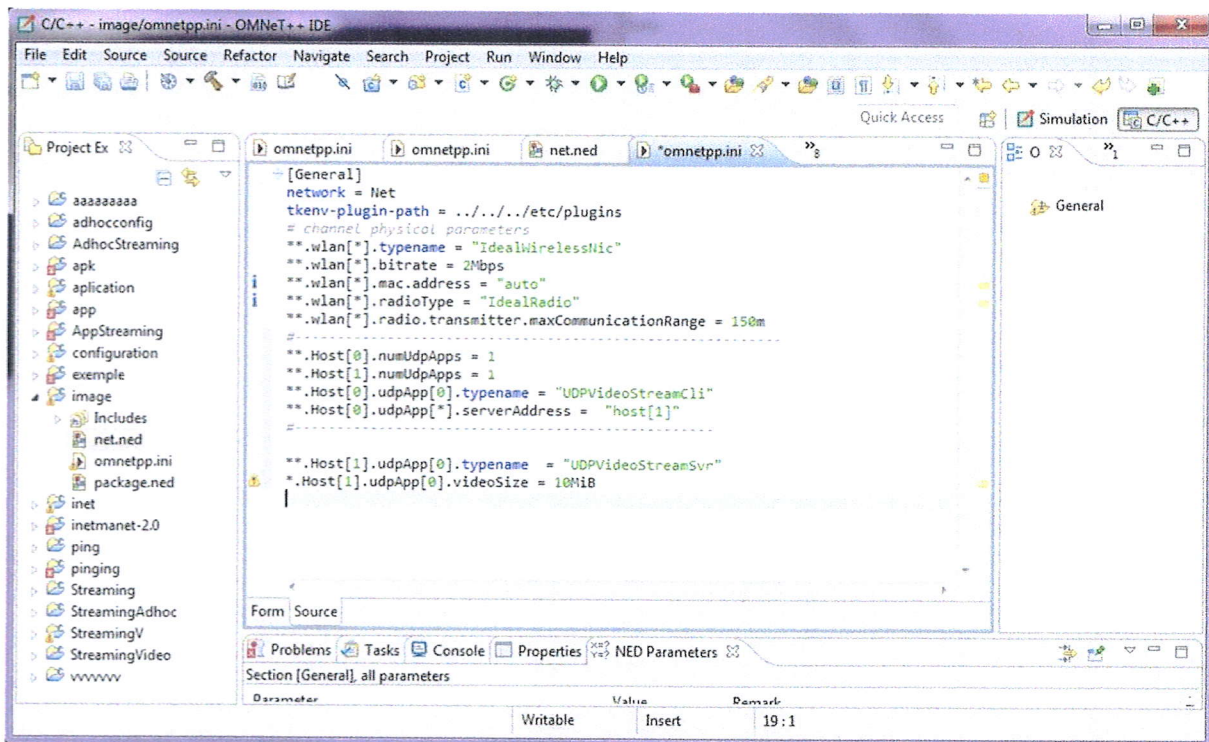


Figure 3-4 : exemple d'un fichier.ini.

### 6.3.3. Fichier.msg

Les modules d'OMNET++ se communiquent par envoi des messages qu'on peut les définir dans un fichier *.msg* qui contient les différents types des champs de données. Après la compilation OMNET++ génère des classes C++ qui correspondent au fichiers *.msg*.

```
message TicTocMsg15
{
    int source;
    int destination;
    int hopCount = 0;
}
```

Figure 3-5: exemple d'un fichier.msg.

## 7. Les plates formes d'OMNET++

Le simulateur OMNET++ n'est pas spécialisé pour les réseaux sans fils, pour cela il existe plusieurs plateformes et simulateurs basés sur OMNET++ qui essaient d'introduire ce manque comme "INET" et "MIXIM".

### 7.1. INET

#### 7.1.1. Définition

INET est une bibliothèque open source pour l'environnement de simulation OMNET++, il fournit des protocoles, des agents et d'autres modèles pour les chercheurs et les étudiants qui travaillent avec les réseaux de communication, il est particulièrement utile lors de la conception et la validation de nouveaux protocoles.

INET contient des modèles pour la pile d'internet (TCP, UDP, IPV4, IPV6...), des protocoles de couche de liaison filaire et sans fils (Ethernet, ppp, IEE 802.11,...), le soutien de mobilité, les protocoles de MANET, Diffserv, plusieurs modèles d'application et de nombreux autres protocoles et composants.

Plusieurs autres cadres de simulation prennent INET comme base, et l'étendent dans des directions spécifiques, tels que les réseaux de véhicules, superposition/réseaux pair à pair.

#### 7.1.2. Architecture d'INET

INET est construit autour du concept de modules qui communiquent par le passage de messages. Les agents et les protocoles réseau sont représentés par des composants, qui peuvent être combinés librement pour former des hôtes, des routeurs, des commutateurs et autres périphériques réseau. De nouveaux composants peuvent être programmés par l'utilisateur, et les composants existants ont été écrits afin qu'ils soient faciles à comprendre et à modifier.

INET profite de l'infrastructure fournie par OMNET++. En faisant usage des services fournis par le noyau de simulation et de la bibliothèque de l'OMNET++ (modèle de composant, le paramétrage, l'enregistrement des résultats, ...etc.).

## 7.2. MIXIM

### 7.2.1. Définition

MIXIM est une plateforme d'OMNET ++ de modélisation créé pour les réseaux mobiles et fixes sans fil (réseaux de capteurs sans fil, les réseaux ad-hoc, les réseaux de véhicules ...etc.). MIXIM se concentre sur les couches inférieures de la pile de protocole, et propose des modèles détaillés de propagation radio à ondes, interférences, estimations, radio émetteur-récepteur consommation d'énergie et les protocoles MAC sans fil. MIXIM est la fusion de plusieurs auparavant OMNET ++ cadres:

- Chism par Universität Paderborn.
- Mac Simulator par Technische Universiteit Delft.
- Cadre Mobilité par Technische Universität Berlindes télécommunications Networks Group.
- cadre par Technische Universiteit Positif Delft.

Il est prévu de fusionner MIXIM dans le cadre de l'INET.

### 7.3. Intégration d'INET et MIXIM

Pour pouvoir utiliser les différentes fonctionnalités offertes par INET et MIXIM il faut les faire intégrer dans votre projet et cela ce fait comme suit :

Vous créez un projet du façon normal (*file ->new project-> omnet++ project*), ensuite cliquez droite sur votre projet, choisissez *propriétés, références*, puis cochez *INET* et *MIXIM*.

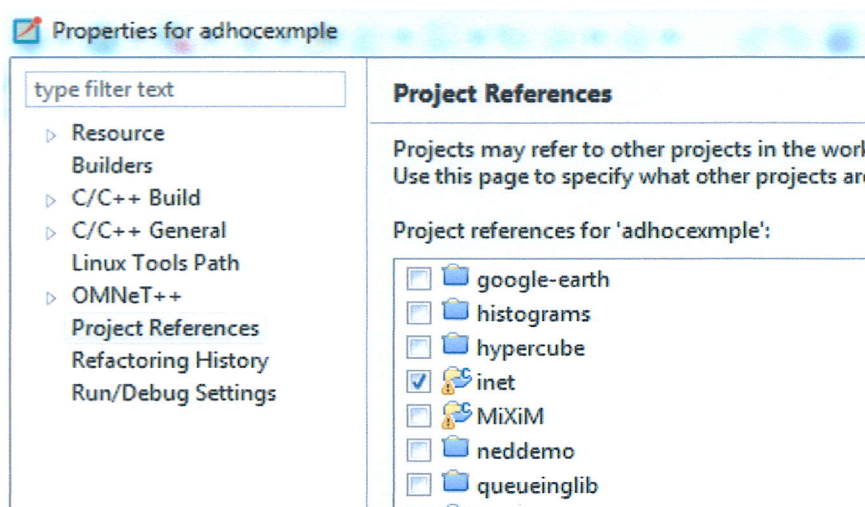


Figure 3-7 : Intégration de MIXIM et INET dans un projet OMNET.

## **8. Conclusion**

Après l'étude des différents simulateurs informatique existant, nous concluons que le choix de simulateur est une question délicate, et la réponse dépend en grand partie sur le besoin de l'utilisateur, pour notre projet, OMNET++ est imposé dans la définition de sujet.



*Analyse  
et Conception*

## **1. Introduction**

Après avoir présenté les notions de base nécessaires pour comprendre les besoins de notre projet, Nous consacrons ce chapitre, à la présentation de l'étape d'analyse et de conception de notre application.

## **2. Analyse**

### **2.1. Le but de projet**

Le but de ce projet est de concevoir puis de réaliser une application sous OMNET++ pour étudier la transmission du streaming vidéo dans les réseaux de mobiles ad-hoc.

### **2.2. Problématique**

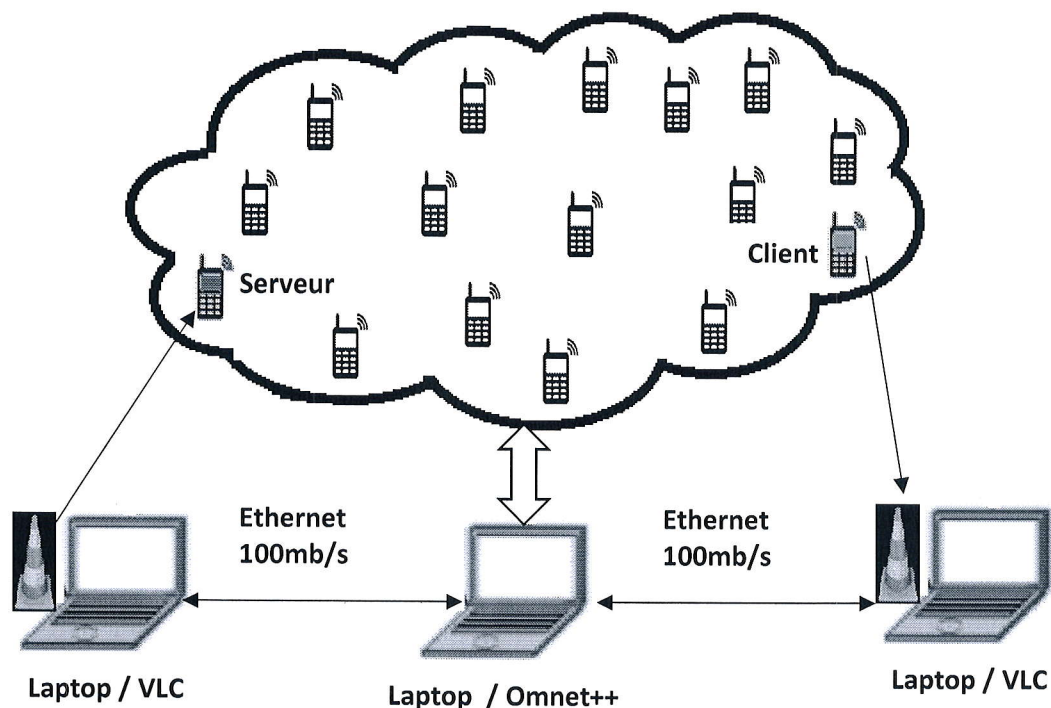
Initialement comme il est décrit dans l'énoncé de projet, l'objectif c'était d'étudier le streaming vidéo en UDP en unicast dans un réseau de mobiles ad-hoc à travers le logiciel de simulation/émulation réseaux OMNET++.

On souhaite envoyer les paquets vidéo à partir de VLC (serveur) en UDP à OMNET++. Cela permet de faire des tests en temps réel et percevoir l'impact des différents paramètres de réseau sur la vidéo reçue par les clients mobiles.

Notre projet est composé de deux parties : une partie consacrée à une émulation temps réel de streaming dont les flux sont générés par le logiciel de streaming VLC (Video LanClient). Et une partie consacrée à une émulation virtuelle dont les flux de streaming sont générés par le programme d'émulation d'une manière aléatoire.

### **2.3. Partie d'émulation temps réel**

Dans cette partie, on devra étudier une communication entre le Serveur VLC et le Client VLC en utilisant le protocole UDP en unicast à travers OMNET++ (Figure 4-1 : Communication client/serveur à travers OMNET++).



**Figure 4-1 :** émulation temps réel.

Pour cela nous devons coder des sockets serveur et des sockets client, chacune sera accompagner de leur module respectives (Serveur, Client), (Voir Figure 4-1).

Les sockets serveur devront récupérer les flux UDP envoyer par le Serveur VLC et le transmettre au Client dans OMNET ++.

Après que le Serveur ait réceptionné les flux son rôle sera de les envoyer vers le Client à travers le réseau d'émulation. Quand le Client reçoit les flux, ils les transmettent au Client VLC.

#### 2.4. Partie d'émulation virtuelle:

Dans cette partie, on devra étudier une communication entre deux modules d'OMNET, un module joue le rôle de serveur qui génère aléatoirement les flux de streaming et l'autre module joue le rôle de client ou de récepteur en utilisant le protocole UDP en unicast (Figure 4-2).



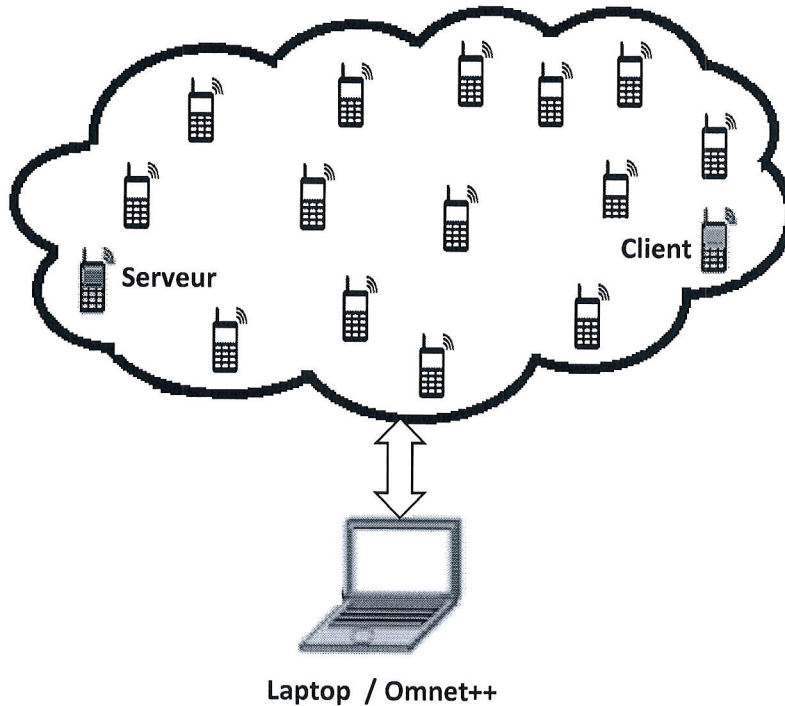


Figure 4-2 : émulation virtuelle

### 3. Conception

La partie analyse nous a permis de décrire les solutions adoptées. Il faut maintenant s'intéresser à la manière dont ces solutions doivent être implémentées. Dans cette section on va présenter la conception de nos modules et les types des messages échangés entre eux.

#### 3.1 Partie d'émulation temps réel:

Dans cette partie, nous avons défini la structure de message de communication « UdpPkt.msg » et codé trois modules : **Listener**, **Server** et **Client**. Les messages sont utilisés, pour encapsuler les flux lors de communication échangés entre les différents modules.

```

//..... UdpPkt.msg .....

packet UdpPkt

Debut
  string payload; // data
  int srcAddress; // adresse de la source
  int destAddress; // adresse de destination
  int destport; // port de destination
  simtime_t Ttransmission; // temps de transmission
  simtime_t Treception; // temps de réception
Fin

```

### 3.1.1 Module Listener

Ce module est une classe, son rôle est de récupérer les flux générés par le serveur VLC.

**Architecture de module Listener :** Le module «**Listener**» suit les étapes suivantes:

- Initialisation les paramètres de module
- Se mettre à l'écoute sur un port dit port d'écoute

**Tant que** n'est pas fini

Lorsqu'il y a des flux en provenance de serveur VLC sur le port d'écoute, les transmettre vers le module **Server**.

**Fin tant que**

### 3.1.2 Module Serveur

Son rôle est d'encapsuler les messages provenant de Listener dans des paquets de type UdpPkt, puis les envoyer vers les modules constituant le réseau de mobiles ad-hoc.

**Architecture de module Serveur :** Le module «**Serveur** » suit les étapes suivantes:

- Initialisation les paramètres de module
- Se mettre à l'écoute des événements produits par le module **Listener**

**Tant que** n'est pas fini

Lorsqu'il y a des messages en provenance de **Listener**

- Les encapsuler dans des paquets UdpPkt
- Les transmettre vers le module **Client** à travers le réseau d'émulation

**Fin tant que**

### 3.1.3 Module Client

Son rôle est de désencapsuler les paquets reçus, puis les envoyer vers le Client VLC.

**Architecture de module Client :** Le module «**Client** » suit les étapes suivantes:

- Initialisation les paramètres de module
- Se mettre à l'écoute sur un port dit port d'écoute

**Tant que n'est pas fini**

Lorsqu'il y a des paquets de type UdpPkt en provenance de réseau

- Les désencapsuler.
- Les transmettre vers le **Client VLC**

**Fin tant que**

### 3.2 Partie d'émulation virtuelle:

Dans cette partie, nous avons réalisé des simulations dans le réseau de mobile ad-hoc en utilisant deux modules principaux : un module **UDPvideostreamSrv** qui génère et distribue les flux vidéo d'une manière aléatoire suivant des demandes des clients, et un module **UdpvideostreamCli** qui demande des flux auprès de serveur.

#### 3.2.1 Module UdpvideostreamCli

Ce module est défini dans OMNET, son rôle est de demander un flux auprès de serveur.

- Initialisation les paramètres de module
- Préparer une demande de flux
- Envoyer la demande au serveur

**Tant que n'est pas fini**

Lorsqu'il y a des paquets en provenance de réseau

- Les supprimer.

**Fin tant que**

#### 3.2.2 Module UDPvideostreamSrv

Ce module est défini dans OMNET, son rôle est de générer aléatoirement des flux et de les envoyer vers le client.

- Initialisation les paramètres de module
- Se mettre à l'écoute sur un port dit port d'écoute

**Tant que** n'est pas fini

Lorsqu'il y a une demande de flux

- Générer aléatoirement un flux vidéo
- Le transmettre vers le demandeur

**Fin tant que**

#### **4. Conclusion**

Dans ce chapitre, nous avons présenté les étapes d'analyse et de conception de notre problème, nous avons présenté tous les modules développés ainsi leurs architectures pour effectuer notre étude. Dans Le chapitre suivant nous présenterons les outils technologiques utilisés et nous ferons une synthèse des résultats obtenus.

*Réalisation*

## 1. Introduction

Au cours de ce chapitre, nous allons décrire l'environnement de travail pour réaliser notre application. Ensuite nous allons faire une synthèse des résultats obtenus de nos simulations. Ces résultats sont présentés sous forme des courbes et des histogrammes calculés en fonction d'un ensemble de métriques.

## 2. L'environnement de travail

### 2.1. Configuration matériel

Dans notre expérimentation nous avons utilisé 3 ordinateurs portables.

Nom	Matériel	Système d'exploitation	Processeur	Logiciels
Portable 1	HP	Windows7	Intel® Core™ i3-2310 CPU @2.10GHZ	Serveur VLC
Portables 2	TOSHIBA	Windows7	Intel® Core™ i3CPU M380@2.53GHZ 2.53GHZ	Client VLC
Portable 3	Samsung	Windows7	Intel® Core™ i3CPU M350@2.27GHZ 2.27GHZ	OMNET++/INET

**Tableau 5-1 :** Equipements de manipulation.

### 2.2. Logiciels

#### 2.2.1. OMNET++

Dans ce projet, nous avons réalisé notre expérimentation à l'aide d'OMNET++ qui est un simulateur à événements discrets orienté objet, basé sur C++. Il a été conçu pour simuler les systèmes réseaux de communication, les systèmes multi processeurs, et d'autres systèmes distribués. OMNET++ est un projet open source dont le développement a commencé en 1992 par Andreas Vargas à l'université de Budapest. Actuellement, ce simulateur est utilisé par des dizaines d'universités pour la validation de nouveaux matériels et logiciels, ainsi que pour l'analyse de performance et l'évaluation de protocoles de communication. L'avantage de OMNET ++ est sa facilité, d'intégration de nouveaux modules et la modification de ceux déjà implémentés.

#### 2.2.2. Le Framework INET2.99.1

C'est un Framework pour les implémentations des protocoles comme : IPv4, IPv6, UDP, TCP...etc., Et plusieurs Modèles d'application. INET comprend également des modèles

de couche de liaison : PPP, Ethernet et 802.11, des implémentations de protocoles de routage, les modèles de mobilité et d'énergie. il soutient aussi la plus part des réseaux sans fil.

### 2.2.3. VLC

VLC est un lecteur de multimédia gratuit qui a eu son succès grâce à sa comptabilité avec la quasi-totalité de format fichier vidéo et audio, il intègre beaucoup de fonctionnalité notamment la lecture des fichiers multimédia en streaming, mais aussi de convertir les fichiers à d'autres format. VLC est disponible sur plusieurs plateformes comme Windows, Linux et Windows phone, de plus, VLC permet la sauvegarde et la diffusion sur réseau de flux en UDP et en TCP.

#### - Installation et configuration de VLC

L'installation de VLC se fait de la même manière que n'importe quel autre logiciels Windows, il faut exécuter l'exécutable et suivre les étapes d'installation.

#### - Configuration de serveur VLC

Le serveur VLC va être configuré pour permettre l'envoi d'un flux vers le réseau de mobile Ad-hoc implémenté dans OMNET.

(1) Dans la barre des taches cliquez sur Media puis Flux .

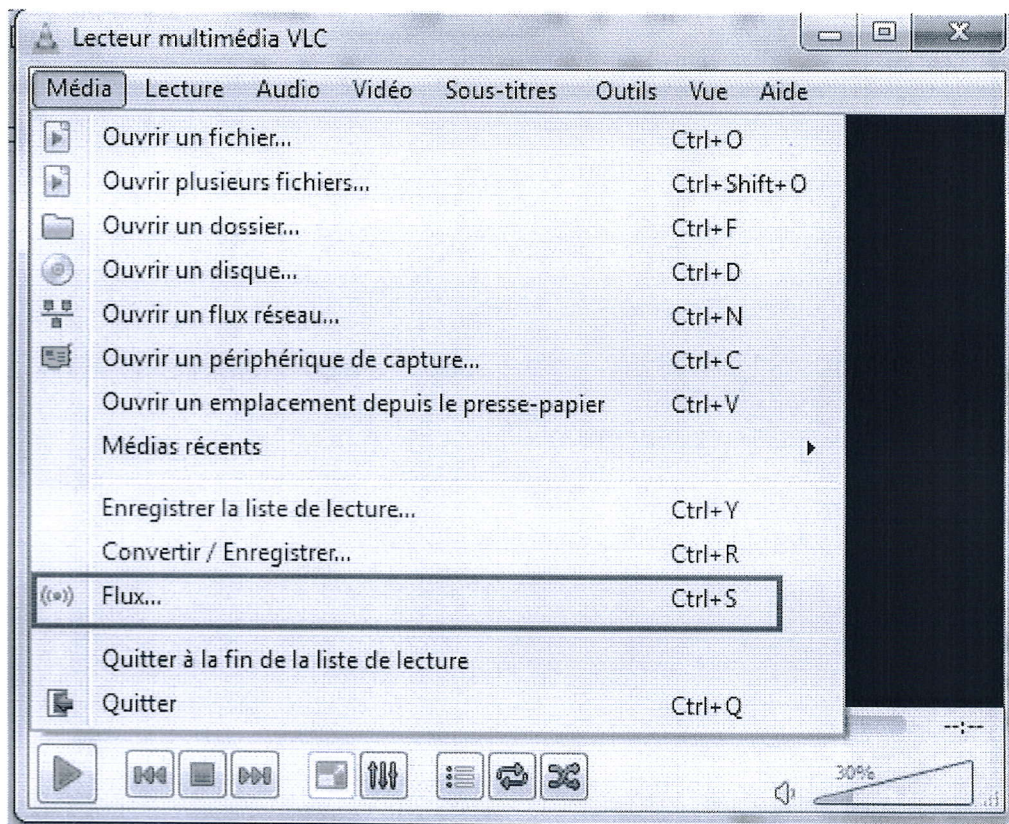


Figure 5-1 : L'option flux.

(2) Ajouter le fichier à transmettre et cliquer sur diffuser.

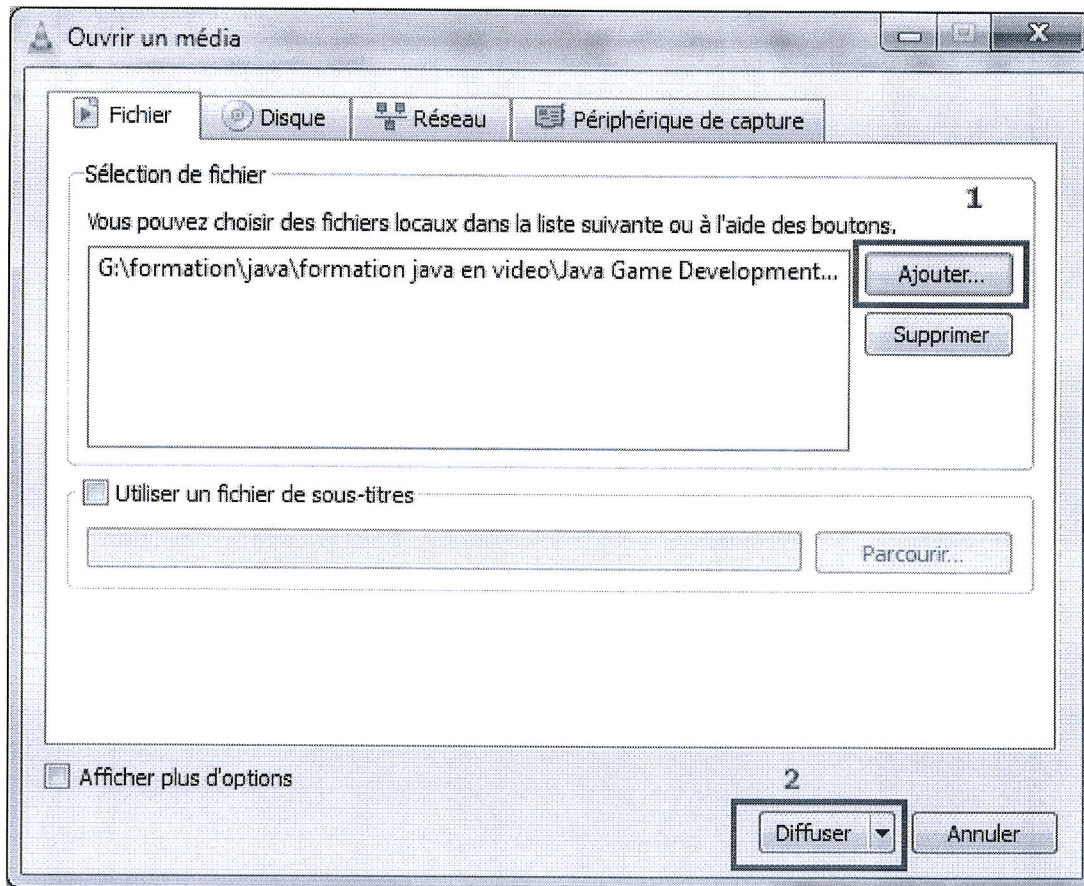


Figure 5-2 : La boîte de dialogue ouvrir média.

(3) Sélectionner le protocole de l'envoi et cliquer sur ajouter : dans notre étude on a utilisé le protocole UDP (Figure5-2).

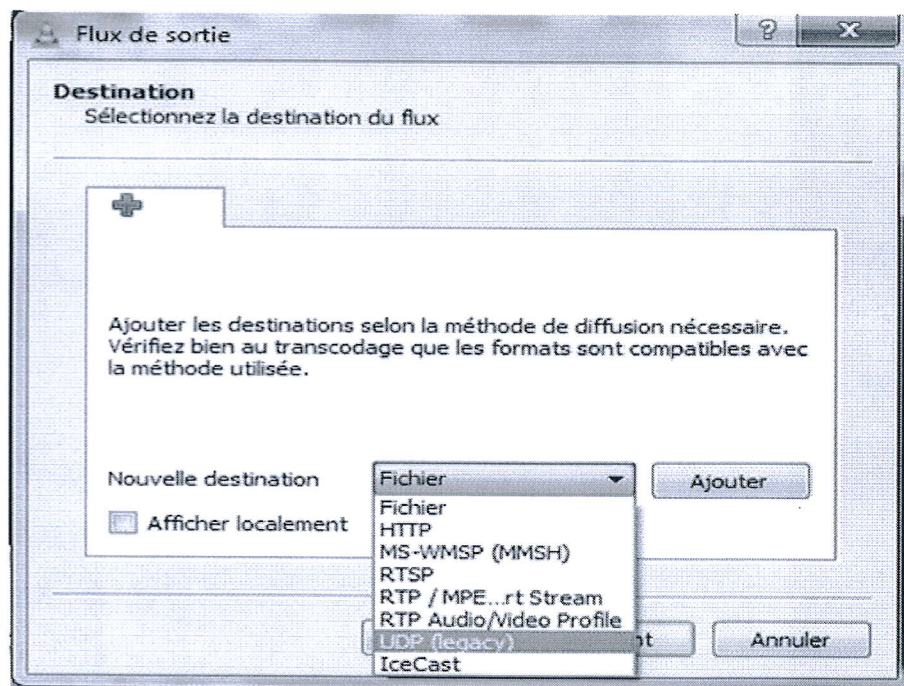


Figure 5-3 : Choix de protocole de transport.

(4) Ensuite configurer le port et l'adresse de diffusion (l'adresse de destination).



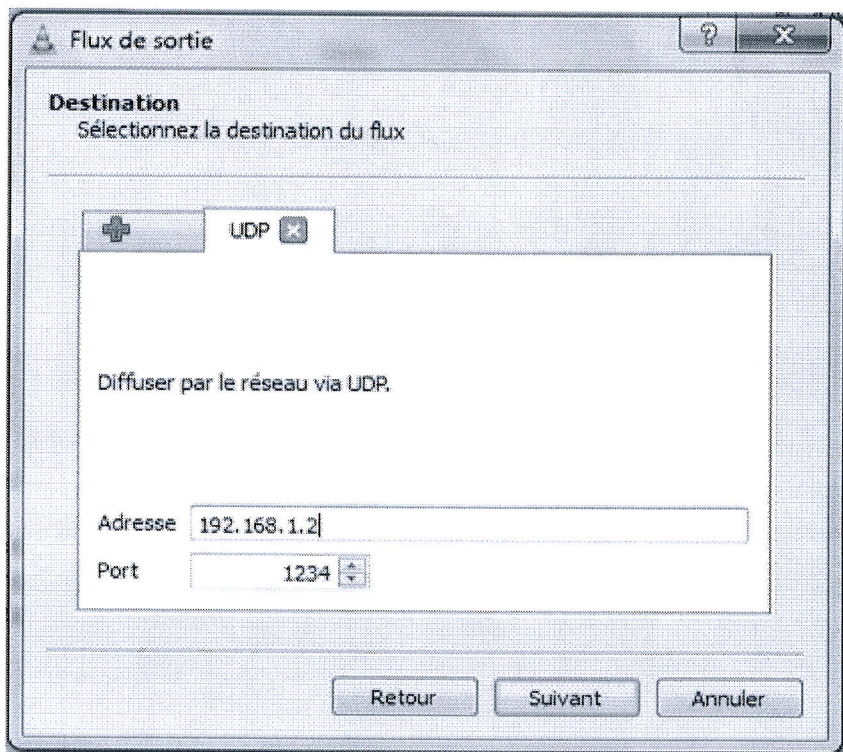


Figure 5-4 : Configuration de la destination.

(5) Sélectionner le type des flux à transmettre puis cliquer sur suivant (le transcodage) (Figure 5-5) Dans notre étude les flux sont de type MPEG-4.

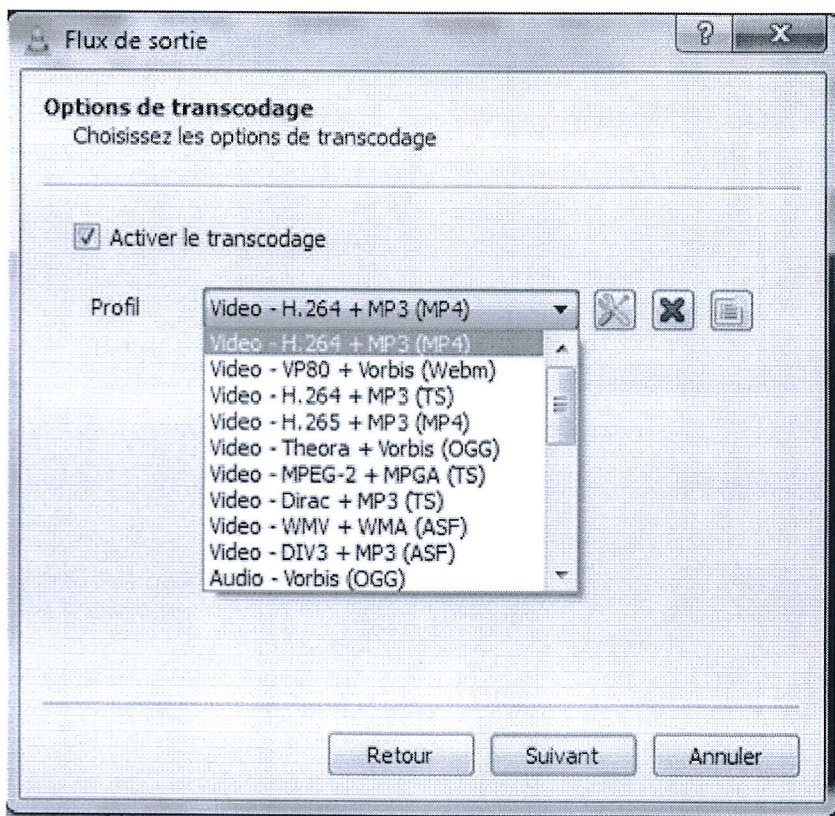


Figure 5-5 : Choix de transcodage.

### - Configuration VLC client

(1) Dans la barre de menu sélectionner Media, puis ouvrir un Flux réseau, une fenêtre s'affiche (Figure 5-6).

(2) Taper l'URL correspondant, au **protocole à l'adresse** et au **: port**.

Exemple : **udp://@192.168.1.1:12345**

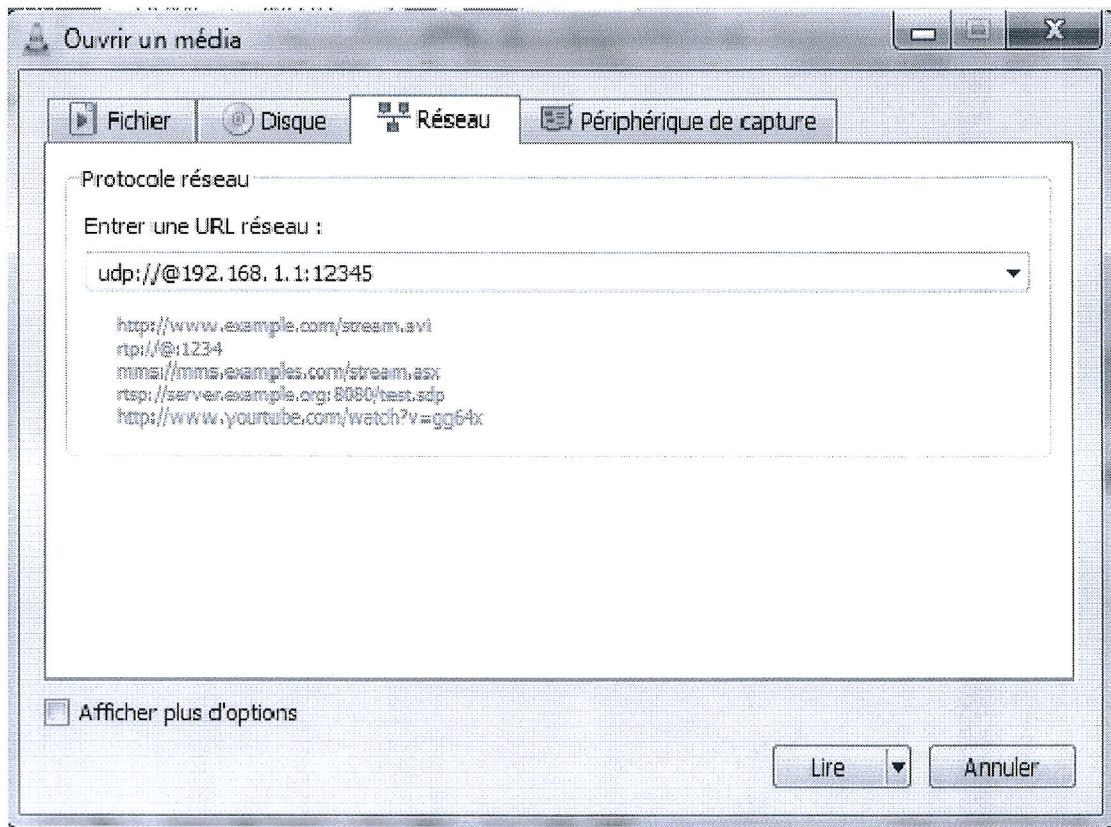
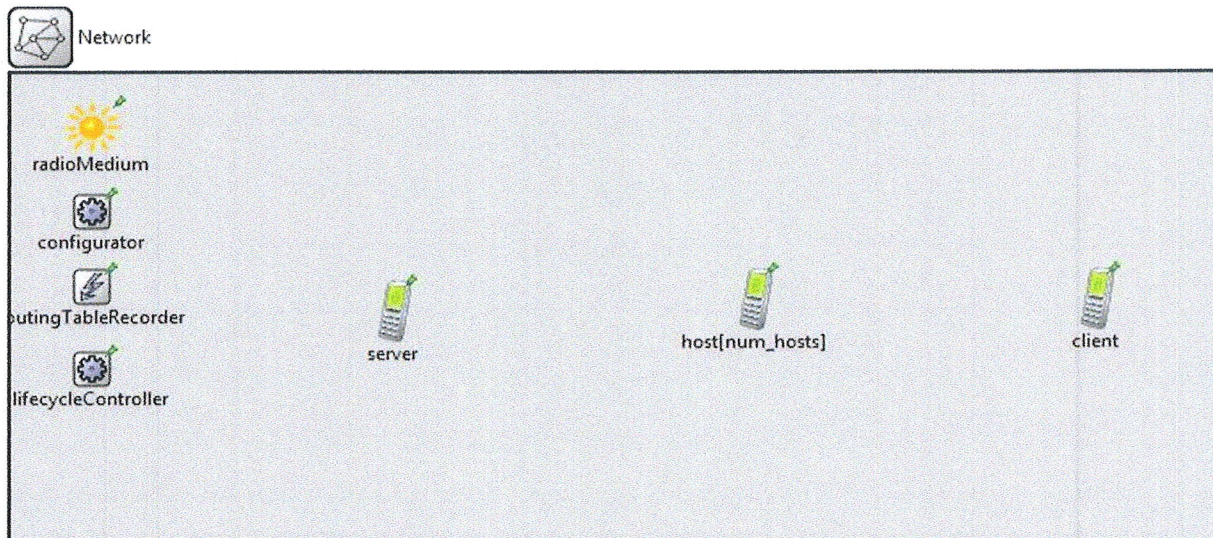


Figure 5-6 : Zone réseau.

### 3. Description de réseau sous OMNET++

Dans notre étude, Nous avons opté deux architectures. Une architecture d'émulation temps réel et une architecture d'émulation virtuelle. Dans l'architecture d'émulation temps réel, les flux de streaming sont générés par VLC, tandis que dans l'architecture d'émulation virtuelle,

les flux sont générés par le module UDPvideostreamSrv d'une manière aléatoire.



**Figure 5-7** : structure de réseau.

- Dans les deux architectures d'émulation, Les modules client, server, et host [num\_hosts] : sont de type **AdhocHost**. Ils prennent en charge la construction de réseau de mobile adhoc.
- **Configurator** : est de type **IPv4NetworkConfigurator**, il gère les adresses IP
- **radioMedium** : est de type *IdealRadioMedium* , il soutient la représentation des trames pour les modèles de la couche physique.
- **routingTableRecorder** : est de type **RoutingTableRecorder**, il enregistre les changements des tables de routages de tous les nœuds
- **lifecycleController** : est de type **LifecycleController**, il gère les opérations de démarrage et de redémarrage des nœuds.
- **connections** : gère les connections entre les nœuds, nous avons choisi une génération aléatoire pour se rapprocher plus aux concepts des réseaux de mobile ad-hoc. La figure 4-2 représente le code de réseau dans OMNET.

```

network Network
{
  parameters:
    int num_hosts;
    @display("bg=742,516");
  submodules:
    radioMedium: IdealRadioMedium {
      parameters:
        @display("p=50,34");
    }
}

```

```
configurator: IPv4NetworkConfigurator {
    @display("p=50,90");
}
routingTableRecorder: RoutingTableRecorder {
    parameters:
        @display("p=50,139");
}
host[num_hosts]: AdhocHost {
    @display("p=324,170");
}
server: AdhocHost {
    @display("p=125,248");
}
client: AdhocHost {
    @display("p=514,82");
}
lifecycleController: LifecycleController {
    @display("p=154,35");
}
connections allowunconnected:
}
```

## 4. Paramètres et variables de simulation

Pour faire notre étude, nous devons d'abord définir les valeurs pour les paramètres utilisés dans le simulateur, dans cette section, nous allons voir comment nous avons choisi ces paramètres et présenter les résultats de simulation qui en découlent. Pour cela nous avons réalisé plusieurs scénarios de simulation dont on varie dans chaque scénario un ensemble de paramètres.

### 4.1. Paramètres de simulation

Il est important de fixer les critères que l'on va prendre en considération. Dans notre travail nous nous sommes intéressés essentiellement aux critères suivants.

#### 4.1.1. Le débit

C'est le nombre de données qui peuvent être transmises d'un point à un autre en un laps de temps déterminé. Le débit détermine la vitesse de transmission des informations sur un réseau informatique. Certains transferts d'informations nécessitent un débit important, comme les transferts de fichiers vidéo.

#### **4.1.2. Les paquets de contrôle**

Détermine le nombre de paquets émis par un nœud dans le but de gérer le réseau (identification, recherche de route, maintien de la table de routage...etc). Comme chaque protocole a son propre algorithme de routage, nous espérons avec cette métrique, trouver lequel d'entre eux utilise le minimum de paquets de control pour un meilleur acheminement de flux.

#### **4.1.3. Les paquets utiles**

Détermine le nombre de paquets de données utiles émis par un nœud, le trafic visible dans la simulation n'est pas utile en entier, les messages de contrôles occupent une grande partie. Un bon protocole fini par acheminer le maximum de données utiles avec un minium de paquets de contrôle dans n'importe quelle topologie.

#### **4.1.4. Les paquets perdus**

Détermine le nombre de paquets de données perdus dans le réseau. Ces pertes sont issues de trafic important ou de temps d'attente assez élevé. Les nœuds perdent des paquets de données utiles ou même de contrôle quand les liaisons sont perdues à cause du déplacement des nœuds récepteurs ou ils sortent du rayon de propagation de leurs émetteurs. Un bon algorithme génère plusieurs chemins pour éviter les pertes dans des cas pareils.

#### **4.1.5. Le trafic émis**

Détermine le volume des paquets de données effectives émis par un nœud source au profit d'un nœud destinataire. Un nœud source peut ne pas avoir suffisamment de temps pour envoyer toutes les données qu'il souhaite envoyer. Les principales raisons sont soit le medium est trop occupé par les nœuds voisins, soit le nœud même est trop chargé par le routage des données d'autres nœuds sources.

#### **4.1.6. Le trafic routé**

Détermine le volume des paquets de données effectives routés par un nœud routeur. Un nœud peut être source, routeur ou destination. Alors l'idéal serait, que chacune des sources envoie directement à sa destination sans déléguer sa charge à des nœuds intermédiaires (routeurs). Cette situation est loin d'être la réalité, mais en choisissant des chemins plus courts, le routage sera diminué en conséquence.

#### **4.1.7. La qualité de la vidéo reçue**

Dans un environnement tel que les réseaux de mobiles ad hoc , le canal de transmission génère des perturbations significatives, notamment pour des applications temps réel de transmissions de vidéos . Le taux de perte est considéré comme l'un des motif majeurs affectant la qualité du vidéo streaming ,cette simulation examine la qualité de la

temps réel de transmissions de vidéos . Le taux de perte est considéré comme l'un des motifs majeurs affectant la qualité du vidéo streaming ,cette simulation examine la qualité de la vidéo reçue tout en tenant compte de l'ensemble des paramètres de réseau et des contraintes des applications de streaming.

#### 4.2. Modèle de topologie

Les réseaux de mobile ad hoc ne comportent pas l'entité fixe, tous les sites du réseau sont mobiles et se communiquent d'une manière directe en utilisant leurs interfaces de communication sans fil. L'absence de l'infrastructure ou du réseau filaire composé des stations de base, oblige les unités mobiles à se comporter comme des routeurs qui participent à la découverte et à la maintenance des chemins pour les autres hôtes du réseau.

Dans notre étude La topologie utilisée est un terrain ou les nœuds mobiles peuvent se déplacer dans un rectangle de **600 x 600 m<sup>2</sup>**

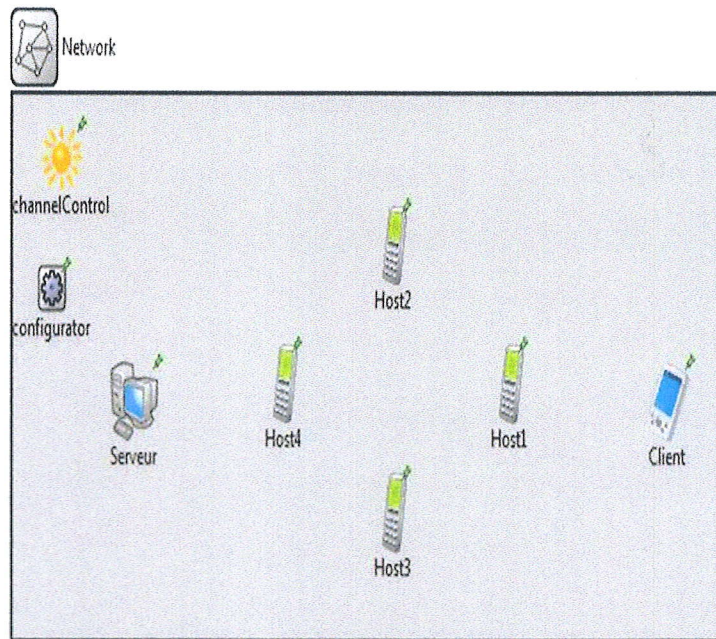


Figure 5-8 : modèle de topologie.

#### 4.3. Modèle d'énergie

Le modèle d'énergie utilisé est celui implémenté dans INET, il attribue aux nœuds une énergie initiale que nous changerons pendant notre simulation, l'énergie est consommée soit par une transmission ou une réception de messages.

#### 4.4. Les protocoles de routage utilisés

Parmi les protocoles de routage implémentés dans OMNET++, nous avons pris à utiliser le protocole AODV et le protocole DSR. Nous signalons, que nous avons trouvé des problèmes dans l'implémentation des autres protocoles, comme le protocole OLSR , DSDV.

#### 4.5. Nombre de nœuds

Au niveau de la topologie, nous avons varié la configuration de nombre des nœuds ou le nombre maximum est limité à 50 nœuds.

#### 4.6. Modèle de Mobilité

Les nœuds peuvent se déplacer selon plusieurs méthodes décrites par les modèles de mobilité, dans notre étude on va utiliser les modèles suivants :

- **Le modèle LinearMobility** : Modèle de mobilité linéaire avec des paramètres (vitesse, l'angle et l'accélération) plus de détails voir [7].
- **Le modèle MassMobility** : Modèle de mobilité aléatoire pour un hôte mobile avec une masse plus de détails voir [7].
- **Le modèle RandomWPMobility** : Les nœuds se déplacent selon des points aléatoires plus de détails voir [7].

#### 4.7. Modèle de trafic

En plus de la mobilité des nœuds, le trafic généré dans les réseaux mobiles ad-hoc dépend aussi de type de techniques de codage de la vidéo utilisées (section compression vidéo) et de protocole de transport utilisé (TCP ou UDP). Par exemple le codage de type MPEG-1 est moins volumineux par rapport au codage de type MPEG-4, cela revient au choix des algorithmes de compression utilisés lors du codage de la vidéo et ses caractéristiques (taux de compression, La qualité de compression...etc). Aussi, le choix de protocole de transport utilisé est très important, en effet TCP qui est un protocole orienté connexion est adapté pour le streaming de basses qualités sur des réseaux sans qualité de service comme l'internet. Tant que UDP est un protocole sans connexion et sans overhead (pas de message de contrôle), il permet une diffusion simple sans aucun contrôle en cas de problème sur le réseau. Le modèle de trafic utilisé dans notre étude ne prend en charge que les trafics du type MPEG-4 en UDP.

Le tableau suivant résume tous les paramètres de simulation.

Critère	valeur
Nombre de nœuds	Max=50
La taille de paquet	512B
La taille de la vidéo	20MO(temps 5 minutes)
Protocole de transport	UDP

Protocole de routage	AODV et DSR
Modèle de mobilité	MassMobility, RandomMobility, LinéaireMobility
zone d'itinérance	600x600 M <sup>2</sup>
Type de diffusion	Unicast
Codec de vidéo	MPEG-4

Tableau 5-2 : paramètres de simulation

4.8. Analyse des résultats

Les scénarios de 1 à 6 représentent les résultats de la partie émulation virtuelle, et les scénarios 7 et 8 représentent les résultats de la partie émulation temps réel.

4.8.1. Scénario 01 : Taux de réception (Trecep/vitesse de noeuds)

Dans ce scénario nous avons deux courbes de nombre de paquets reçus en fonction de la vitesse de déplacement selon les trois modèles de mobilité : MassMobility, LineaireMobility et RandomMobility

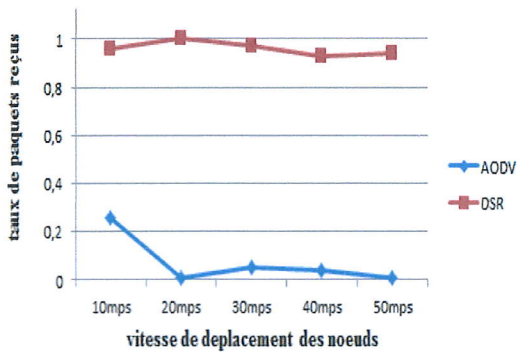


Figure 5-9 :MassMobility

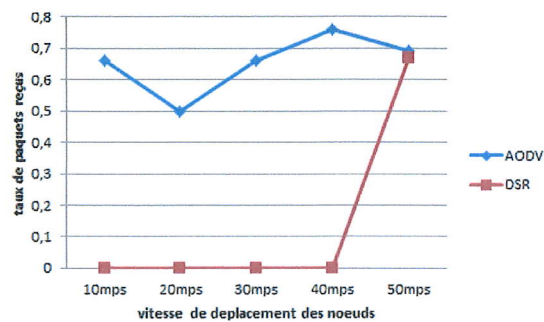


Figure 5-10 : RandomMobility

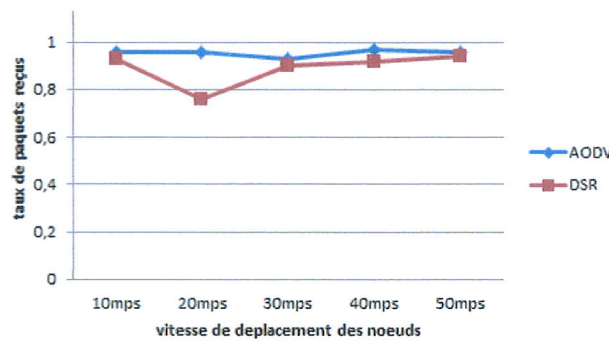


Figure 5-11 :LinéaireMobility

Nous remarquons dans la figure 5-9 que le taux de réception de paquets avec le protocole AODV se diminue en augmentant la vitesse de déplacement des noeuds, par contre avec DSR le taux de réception des paquets se change d'une manière irrégulière.



Dans la figure 5-10 le taux de réception avec le protocole AODV est presque converge vers à une valeur proche de 0.8. Par contre avec DSR, le taux de réception est de zéro avec des vitesses de mobilités inférieure à 40mps. Si la vitesse arrive à 50mps, DSR enregistre un taux de 0.7.

Dans la figure 5-11, on remarque que AODV est positionné au-dessus de DSR avec un comportement aléatoire.

Les différences entre les courbes sont dues à la différence de modèle de mobilité des nœuds et leurs positionnements sur le terrain.

En conclusion, on déduit que le protocole AODV est meilleur que le protocole DSR dans les deux modèles de mobilité LinéaireMobility et RandomMobility. Le protocole DSR est meilleur avec le modèle MassMobility.

**4.8.2. Scénario 02 : Taux des paquets perdus (Tauxpqt/vitesse de neuds)**

Pour ce scénario nous avons deux courbes qui illustrent le nombre de paquets perdus en fonction de la vitesse de déplacement des nœuds, dans les trois modèles de mobilité : MassMobility, LineaireMobility et RandomMobility

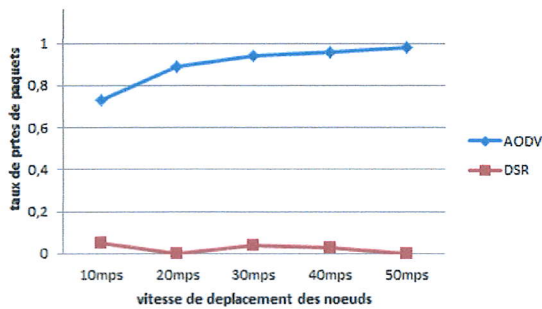


Figure 5-12 : MassMobility

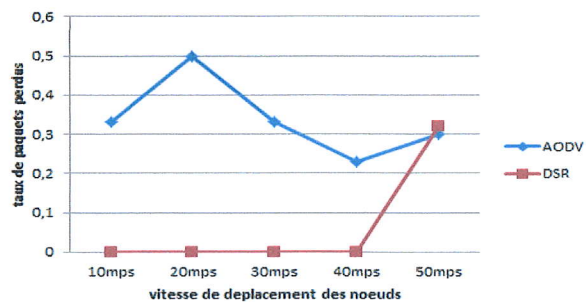


Figure 5-13 :RandomMobility

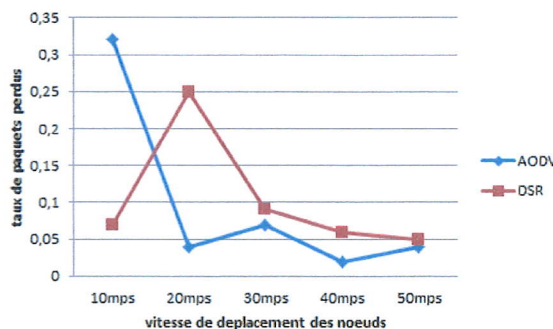


Figure 5-14 :linéaireMobility

La différence est bien visible, dans le modèle de Massmobility, DSR a une perte presque négligeable, contrairement à AODV qui perd plus de paquets avec l’augmentation de la mobilité et cela à cause des chemins qui ne restent plus valides (Figure 5-12)

Dans le modèle Randommobility(Figure 5-13), on remarque que AODV n’assure pas

un taux de 0.5 en moyen ,contrairement à DSR dont le taux de paquets perdus augmente d'une manière directement proportionnelle quant la vitesse dépasse 40mps.

Dans le modèle linearmobility (Figure 5-14), les deux protocoles enregistrent un taux de perte inférieur à 0.35, les perds diminuent lorsque la vitesse des nœuds augmente.

La logique théorique s'attend à une perte croissante avec la mobilité. AODV valide la théorie avec le type de mobilité MassMobility, cependant cette théorie ne pas être valide avec les autres modèles de mobilités à cause de la différences entre eux et la vitesse de déplacement des noeuds.

**4.8.3. Scénario 03 : Nombre de paquets de contrôle (nbreC / vitesse de nœuds)**

Les trois figures suivantes présentent le nombre de paquets de contrôle émis pour les deux protocoles AODV et DSR en fonction de la vitesse de déplacement des nœuds avec trois modèles de mobilité MassMobility, LineaireMobility et RandomMobility

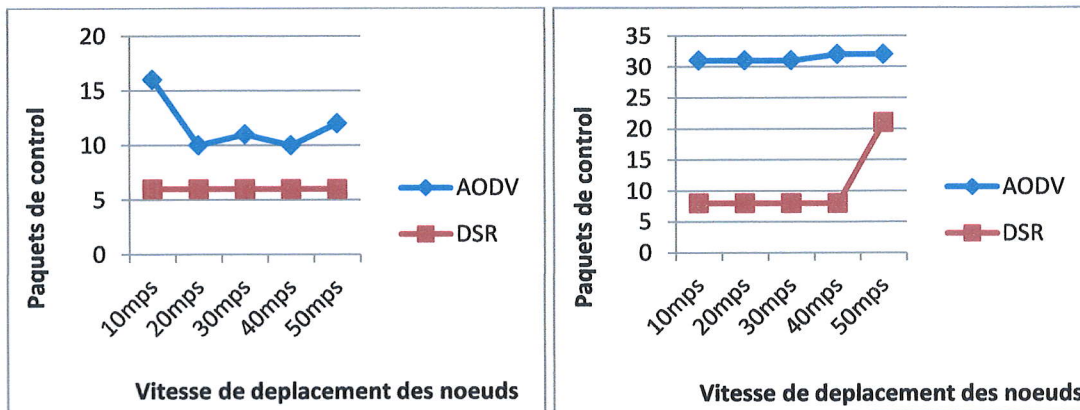


Figure 5-15 :MassMobility Figure 5-16 :RandomMobility

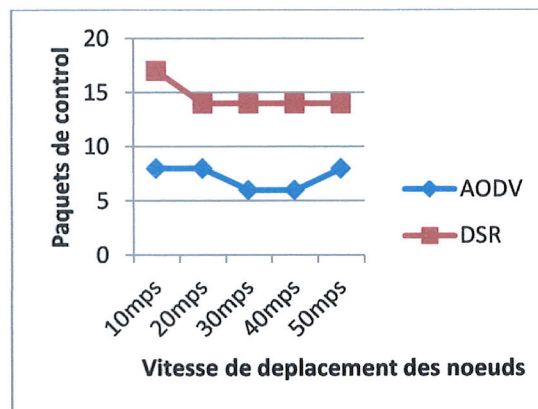


Figure 5-17 : LinéaireMobility

Dans le cas de MassMobility nous remarquons que la courbe du protocole AODV est positionnée au dessus de la courbe du DSR, cela signifie que AODV génère plus de paquets de contrôle par rapport à DSR même avec des forte mobilités.

Dans le cas de RandomMobility, nous remarquons que la courbe du protocole AODV

est positionnée au dessus de courbe du DSR mais avec un nombre de paquets de contrôle élevé et stable, par contre dans DSR le nombre de paquets reste à une valeur entre 5 et 10. Le nombre de paquets augmente vers 20 une fois la vitesse de mobilité atteint 40mps.

Dans le cas de LinéaireMobility, la position des deux courbes est inversée, la courbe de AODV est positionnée au dessous de celle de DSR mais avec le même comportement.

La logique théorique sollicite une augmentation du trafic de contrôle avec l'augmentation de la vitesse de mobilité. AODV augmente naturellement son besoin en paquets de contrôle dans le cas de Randommobility et Massmobility. Par contre, DSR semble baisser son utilisation de paquets de contrôle avec l'augmentation de la mobilité. Dans le cas de LinéaireMobility DSR génère plus de paquets de contrôle que AODV.

#### 4.8.4. Scénario 04 : délai de bout en bout en fonction de nombre de nœuds

Le délai de bout en bout est la durée de temps prise par le paquet, à partir de sa source d'envoi vers sa destination de réception. Nous présentons ici ce délai par rapport au nombre des nœuds mobiles dans le réseau pour les deux protocoles AODV et DSR.

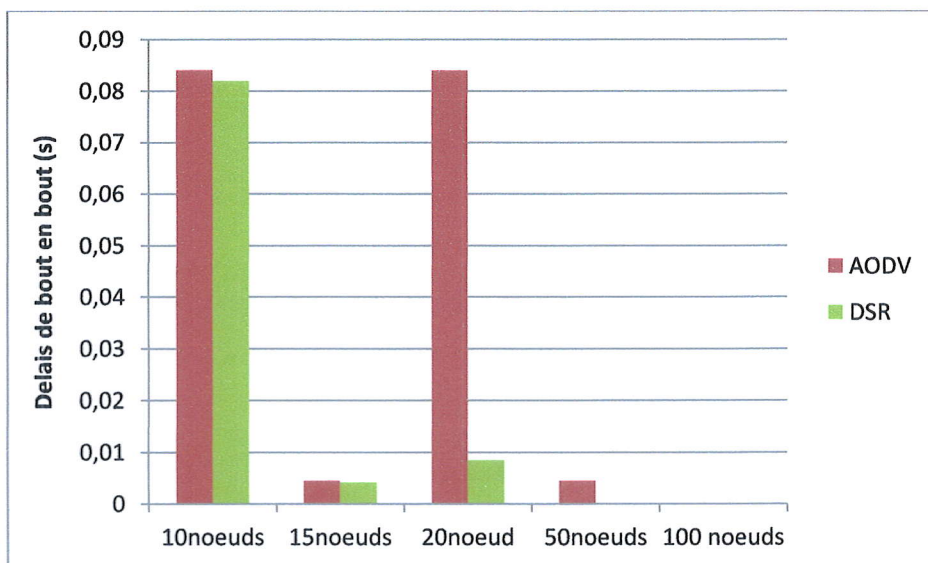


Figure 5-18 : délai de bout en bout Vs nombre de nœuds.

On remarque qu'avec un nombre de 10 nœuds les protocoles AODV et DSR ont presque la même valeur de délais de bout en bout qui atteint une valeur de 0.082s. Avec 15 nœuds les deux protocoles ont aussi presque la même valeur qui est proche de 0.005s.

Pour un nombre de 20 nœuds, le délai atteint une valeur de 0.082s avec AODV et 0.01s avec DSR.

Pour un nombre de 50 nœuds, le délai atteint une valeur de 0.005s avec AODV et 0 avec DSR.

En conclut que le protocole DSR est meilleur que le protocole AODV si la taille de

réseau agrandi.

#### 4.8.5. Scénario 05 : énergie restante en fonction de modèle de mobilité

Cette figure représente la moyenne de l'énergie restante en fonction des trois modèles de mobilité : MassMobility, LineaireMobility et RandomMobility

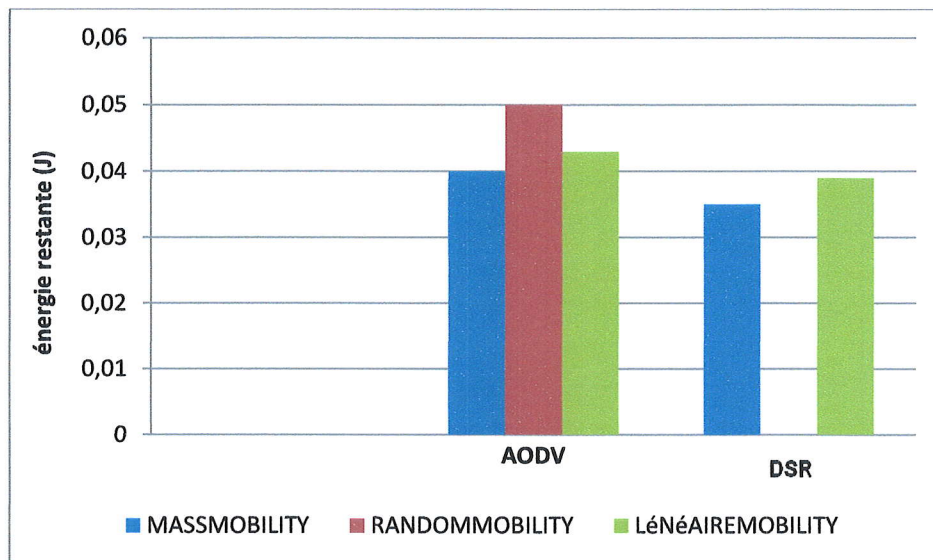
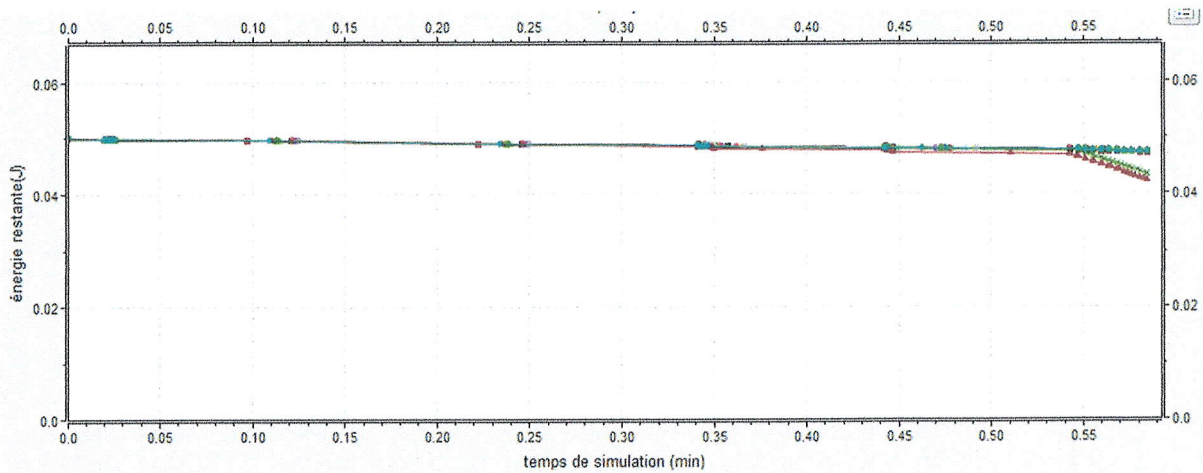


Figure 5-19 : énergie restante Vs le modèle de mobilité.

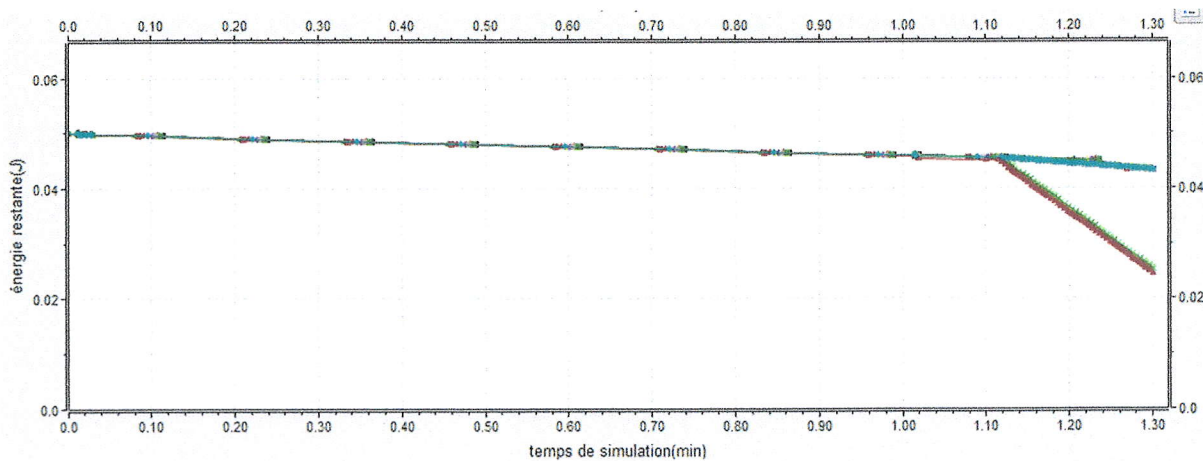
On remarque que la consommation d'énergie est presque la même dans les trois modèles avec le protocole AODV, par contre avec DSR, on observe que les nœuds consomment la totalité de l'énergie avec Randommobility, et avoir des valeurs de consommation entre 0.03 et 0.04 dans les deux autres modèles. On remarque aussi que les nœuds utilisant le protocole AODV conserve l'énergie mieux que les nœuds utilisant le protocole DSR.

#### 4.8.6. Scénario 06 : énergie restante restante en fonction de temps de simulation

Le temps de simulation joue un rôle important sur le comportement des nœuds pour cela nous avons joué sur ce paramètre pour évaluer la durée de fonctionnement du réseau avec les deux protocoles de routage AODV et DSR dans le cas de RandomMobility .



**Figure 5-20 :** énergie restante Vs temps de simulation pour le protocole AODV.



**Figure 5-21 :** énergie restante Vs temps de simulation pour le protocole DSR.

Les deux figures (5-20 et 5-21) présentent un aperçu sur la consommation d'énergie au niveau de quelques nœuds pour les deux protocoles AODV et DSR. On peut remarquer que la consommation d'énergie est presque la même pour les deux protocoles.

On remarque que la consommation se diminue avec le temps et cela est logique.

#### 4.8.7. Impact du facteur de paquet perdu sur la qualité de réception de la vidéo

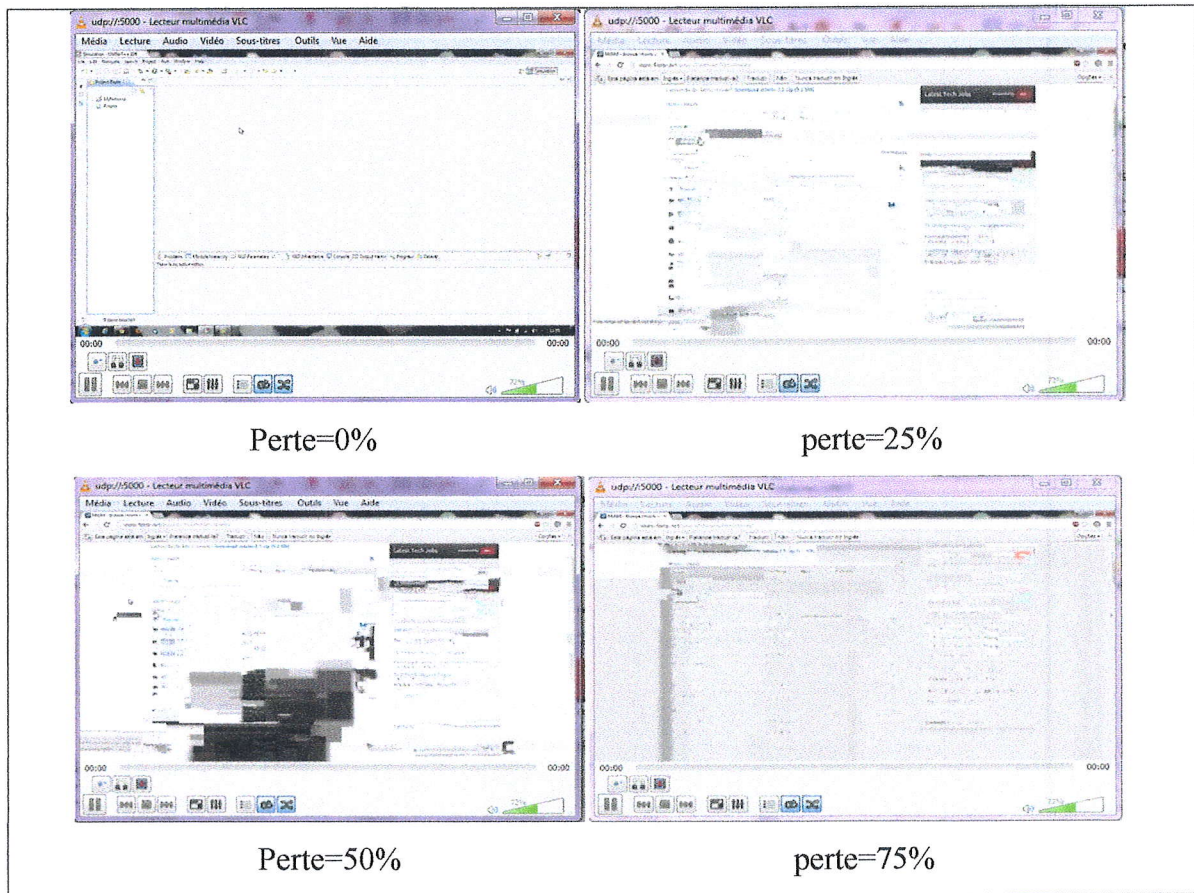
Dans cette simulation on a essayé d'étudier l'effet des paquets perdus sur la qualité de la vidéo reçue. Les résultats obtenus sont les suivants (Figure 5-22) :

Pour 0% de paquets perdus: la qualité de vidéo est bonne.

Pour 25% de paquets perdus: la qualité de vidéo est pauvre

Pour 50% de paquets perdus: la qualité est médiocre

Pour plus de 75% de paquets perdus : la qualité de vidéo est très médiocre

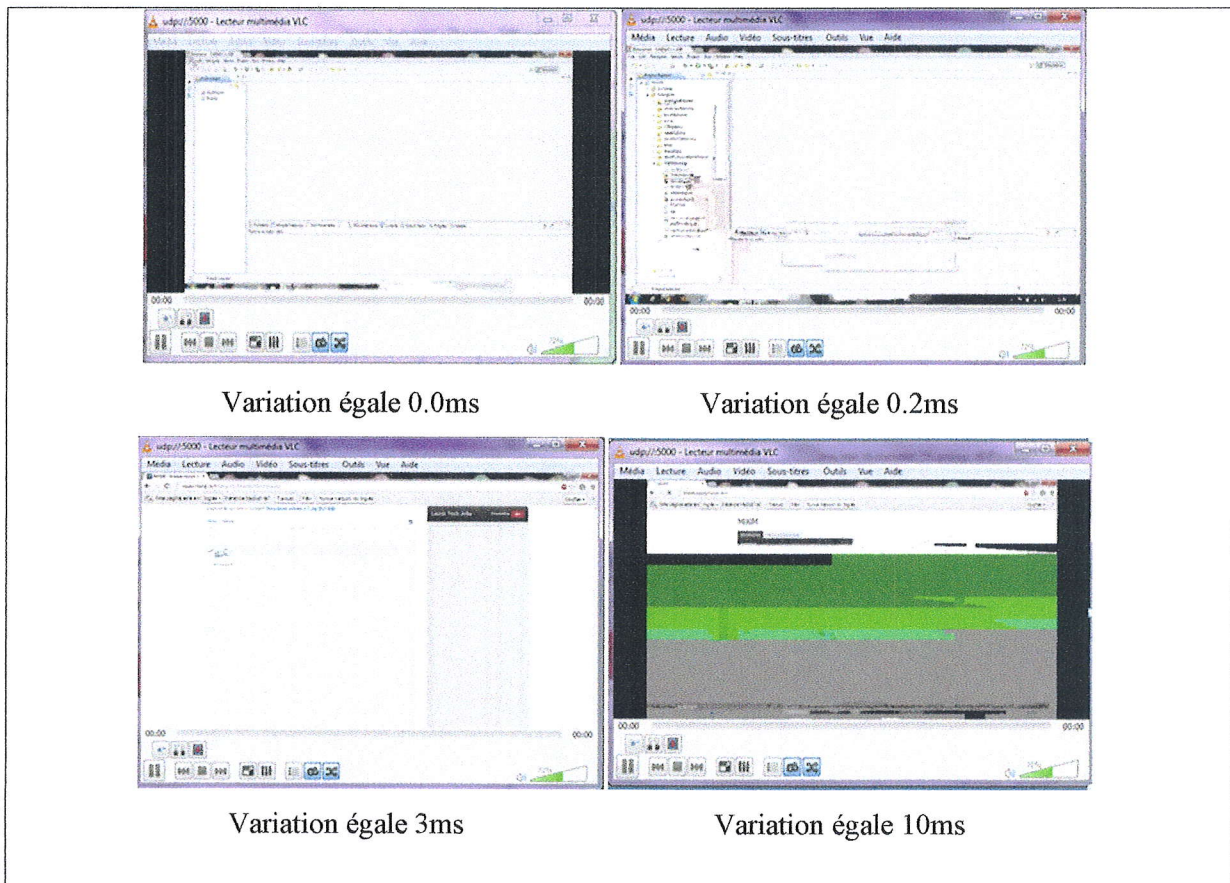


**Figure 5-22 :** Impact du facteur de paquet perdu sur la qualité de réception de la vidéo

En conclusion : on remarque que le streaming n'est pas sensible beaucoup aux pertes de paquets.

#### 4.8.8. Impact du facteur de variation de délai D (Gigue) sur la qualité de la réception de la vidéo

Dans cette simulation on a essayé d'étudier l'effet de la gigue sur la qualité de réception de la vidéo. Pour cela nous avons varié la gigue entre 0ms et 10ms. Et on a remarqué que la qualité de la vidéo de réception se dégrade à chaque augmentation de la gigue (Figure 5-23).



**Figure 5-23 :** Impact du facteur de la gigue sur la qualité de la réception de la vidéo

En conclusion : on remarque que le streaming est très sensible pour la gigue

### 5. Conclusion

Ce chapitre a été consacré à la présentation des résultats de simulation de notre projet. Nous avons décrit au début l’environnement de réalisation de notre projet. Ensuite nous avons présenté les paramètres et les facteurs qui rentrent dans notre simulation et en fin nous avons commenté les résultats des simulations obtenus.

## *Conclusion générale*

Dans ce projet de fin d'étude, nous nous sommes intéressés à étudier le streaming vidéo dans les réseaux de mobiles ad hoc à travers le simulateur OMNET++.

Tout d'abord nous avons commencé par la technologie du streaming, nous avons présenté son architecture, son principe de fonctionnement, les différentes modes de distribution, les champs d'application, ses contraintes et quelques avantages et inconvénients, aussi, nous avons défini les réseaux de mobiles ad hoc, leurs caractéristiques, leurs fonctionnement, quelques protocoles de routage et leurs domaines d'utilisation. Cette étude nous a permis de découvrir les problèmes de streaming vidéo dans les réseaux de mobiles ad hoc qui se situent notamment dans la mobilité des nœuds, la communication sans fil, et la qualité de service des sessions multimédias lors de la mobilité.

Ensuite, un passage par une étude des simulateurs de réseaux et les outils que nous avons utilisés pour le développement de notre application.

Finalement, afin de faire notre étude nous avons réalisé plusieurs scénarios de simulation. Ces simulations nous ont permis de faire une étude comparative entre le protocole AODV et le protocole DSR, et voir l'impacte de délai et de la gigue sur la qualité de réception de la vidéo.

L'élaboration de ce projet de fin d'étude nous a permis de :

- Faire une étude approfondie des réseaux de mobiles ad-hoc et de streaming.
- Apprendre la simulation avec OMNET++.
- Maîtrise de la programmation avec C++.
- L'utilisation d'INET qui nous a aidées a simulé les réseaux de mobiles ad-hoc.
- Le passage des flux vidéo en temps réel avec VLC

En perspective, nous cherchons à développer des nouveaux protocoles pour faire le streaming dans les réseaux de mobiles ad-hoc.



## *Bibliographie*

- [1] Amine Chaouche, « Gestion de la qualité de service pour le streaming de contenu audiovisuel sur les réseaux Pair-à-Pair », Mémoire de Magister, Ecole Nationale Supérieure d'Informatique, année 2010.
- [2] Kamil Chebira, « Etude et analyse de la stabilité des protocoles de routage dans les réseaux ad-hoc », Mémoire de Magister, Université Hadj Lakhdar BATNA ,année 2007.
- [3] Abdelaziz Bouzaher, « Approche agent mobile pour l'adaptation des réseaux mobiles ad hoc »,Mémoire de Magister , université Mohamed Khidar Biskra.
- [4] Leila Imane Niar, « Analyse Graphique pour la surveillance dans un réseau de capteurs sans fils (RCSF) », Mémoire de Magister, 2012.
- [5] Nour el houda Boussadi,k.dahina,N.Abderrhmane, « les simulateurs réseaux et technologie reseaux » , année 2014
- [ 6] Saloua Chettibi, « Protocole de routage avec prise en compte de la consommation d'énergie pour les réseaux mobiles ad-hoc », Mémoire de Magister , université Mentouri Constantine, 2008.
- [7] Manuel d'utilisation de INET dans OMNet++ : « Framework INET for OMNet++ » : generated on May 2015

## *webographie*

[w1] :[https://fr.wikipedia.org/wiki/Routage\\_ad\\_hoc#Protocole\\_proactif](https://fr.wikipedia.org/wiki/Routage_ad_hoc#Protocole_proactif)

[w2]:[http://www.memoireonline.com/01/09/1878/m\\_Les-technologies-sans-fil-Le-routage-dans-les-reseaux-ad-hoc-OLSR-et-AODV2.html](http://www.memoireonline.com/01/09/1878/m_Les-technologies-sans-fil-Le-routage-dans-les-reseaux-ad-hoc-OLSR-et-AODV2.html)

[w3] [http://www.memoireonline.com/12/07/738/m\\_effets-mobilite-protocoles-routage-reseaux-ad-hoc17.html](http://www.memoireonline.com/12/07/738/m_effets-mobilite-protocoles-routage-reseaux-ad-hoc17.html)

