



**MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE  
LA RECHERCHE SCIENTIFIQUE  
UNIVERSITÉ ABDELHAMID IBN BADIS MOSTAGANEM**

**Faculté des Sciences Exactes & Informatique  
Département d'Informatique**

**MEMOIRE DE FIN D'ETUDES  
Pour l'Obtention du Diplôme de Master en Informatique  
Option : Ingénierie des Systèmes d'Information**

**Thème**

**Routage réactif et proactif (Unicast) dans les réseaux  
mobiles Ad Hoc**

***Présenté par :***

**BENALI Amal**

**DABO Mohamed**

***Encadré par:***

**Mme SAICHI SOUAD Benyamina**

**Année Universitaire 2011/ 2012**

# Remerciements

---

*Dédicaces*

AMEL

# *Dédicaces*

DARO

## Introduction Générale

L'évolution récente des moyens de communication sans fil a permis la manipulation d'informations au travers d'unités de calcul portables aux caractéristiques bien particulières (faible capacité de stockage, source d'énergie autonome, puissance limitée, etc....) qui accèdent au réseau par le biais d'une interface de communication sans fil.

Les environnements mobiles permettent la mise en réseau de sites dont le câblage serait trop couteux à réaliser dans leur totalité, voire même impossible. Contrairement à l'environnement statique, l'environnement mobile permet aux unités de calcul une libre mobilité et ne pose aucune restriction sur la localisation des usagers. La mobilité et le nouveau mode de communication utilisé engendrent des problèmes propres à l'environnement mobile : une fréquente déconnexion, un débit de communication et des ressources modestes et des sources d'énergie limitées.

Les réseaux mobiles sans fil, peuvent être classés en deux catégories : les réseaux avec infrastructure qui utilisent généralement le modèle de la communication cellulaire, et les réseaux sans infrastructure ou les réseaux mobiles ad hoc.

Plusieurs systèmes utilisent déjà le modèle cellulaire et connaissent une très forte expansion à l'heure actuelle (réseaux GSM) mais nécessitent une importante infrastructure logistique et matérielle fixe.

Les réseaux mobiles ad hoc, par contre n'ont besoin d'aucune infrastructure fixe préexistante. Ce type de réseau s'adapte spontanément, pour répondre à un besoin, d'où la terminologie ad hoc (en latin : « **qui va vers, ce vers quoi il doit aller** », c'est-à-dire « **formé dans un but précis** ») et se configure de façon complètement autonome et dynamique.

Les réseaux mobiles ad hoc sont des réseaux sans fil formés par des appareils (ordinateurs, des PDA, des téléphones mobiles, etc....), appelés nœuds, qui communiquent entre eux sans passer par une autre infrastructure et sans que ces communications nécessitent une administration centrale. Chaque nœud du réseau est équipé d'une interface radio, qui peut être différente d'un nœud à l'autre : Bluetooth, Wifi... et reste libre d'intégrer ou de quitter le réseau. Du fait que le rayon de propagation des transmissions des hôtes soit limité, et afin que le réseau reste connecté, il se peut qu'un hôte mobile se trouve dans l'obligation de demander de l'aide à un autre hôte pour pouvoir communiquer avec son correspondant. Il se peut donc que l'hôte destination soit hors de la portée de communication de l'hôte source, ce qui nécessite l'emploi d'un routage interne par des nœuds intermédiaires afin de faire acheminer les paquets de messages à la bonne destination. La gestion de l'acheminement de données ou

le routage, consiste à assurer une stratégie qui garantie, à n'importe quel moment, la connexion entre n'importe quelle paire de nœuds appartenant au réseau. La stratégie (*ou protocole*) de routage doit prendre en considération les changements de la topologie ainsi que les autres caractéristiques du réseau mobile ad hoc (bande passante, nombre de liens, ressources...).

Les protocoles de routage peuvent être classés principalement en deux catégories :

Les protocoles de routage proactifs qui entretiennent à l'avance toutes les routes du réseau en se basant sur l'échange périodique des tables de routage, ainsi que sur l'état de lien. Avec ces protocoles, les nœuds échangent entre eux des informations sur les routes existantes et chaque nœud met à jour sa table de routage, ce qui lui permet d'avoir une vue globale du réseau. Contrairement aux protocoles proactifs, les protocoles réactifs ne cherchent à calculer une route qu'à la demande. Un protocole réactif réduit le nombre de diffusions de messages en créant les routes lors du besoin et en limitant la répercussion des modifications topologiques aux seules routes en cours d'utilisation.

Notre projet qui s'intitule «*Routage réactif et proactif (unicast) dans les réseaux mobiles ad hoc* » entre dans le cadre de l'étude du problème de routage dans les réseaux mobiles Ad Hoc. Notre étude offre principalement une étude synthétique des différents mécanismes de routages, dans le but de résoudre le problème d'acheminement de données entre les hôtes mobiles du réseau.

Beaucoup d'études existent sur les réseaux mobiles ad hoc, vu que c'est un domaine contenant des sujets très intéressants. La méthode la plus utilisée pour étudier ces réseaux, consiste à faire cette étude par des simulations. Une simulation est une technique informatique permettant de reconstituer le déroulement d'un phénomène en le modélisant par ordinateur et en introduisant quelques paramètres dont la valeur est choisie par l'utilisateur.

Les simulations sont plus rapides, moins coûteuses en ressources, répétables et en plus, permettent d'isoler des paramètres qui affectent les performances. [1]

Plusieurs logiciels existent pour la simulation des réseaux mobiles ad hoc (NS-2, GloMoSim, NetSim, JavaSim, Qualnet, OMNET ++, OPNET, etc....). Pour simuler les réseaux mobiles ad hoc, nous avons besoin d'exprimer le comportement des nœuds dans la simulation ce qui est fait à partir d'un modèle de mobilité qui a pour rôle de faire bouger les nœuds pendant l'exécution de la simulation. Nous allons donc étudier les différents modèles de mobilité afin d'avoir une vue sur les modèles existants. Nous allons ensuite proposer un ensemble de métriques pour caractériser ces modèles, les comparer et les classer.

Ce mémoire se divise en quatre chapitres :

Le chapitre 1 porte sur un état de l'art sur les réseaux mobiles ad hoc. Dans cette partie nous présentons les environnements mobiles et les principaux concepts liés à ces environnements. Nous commençons par la définition de l'environnement et citer les deux classes qui le constituent. Ensuite nous introduisons le concept de réseau mobile ad hoc, ces caractéristiques inhérentes et quelques domaines d'application.

Dans le chapitre 2, nous introduisons le concept de routage et protocoles de routage dans le réseau mobile ad hoc. Ceci permet d'un coté de situer les réseaux mobiles ad hoc par rapport à l'environnement mobile, et d'un autre coté de mettre l'accent sur le problème de routage, problème qui est propre à l'environnement du réseau mobile ad hoc. Nous définissons le problème d'acheminement de données dans de tels environnements et nous soulignons sa difficulté et les principales contraintes que la stratégie de routage doit respecter.

Nous décrivons aussi les principales caractéristiques et fonctionnalités des stratégies de routage les plus connues permettant d'assurer l'acheminement des données entre les hôtes mobiles du réseau.

Dans le chapitre 3, nous présenterons les outils d'implémentation tels que les langages tcl et otcl qui sont utilisés pour décrire les conditions de la simulation, et enfin nous présentons le simulateur NS2, que nous allons utiliser pour l'implémentation des différents protocoles de routage.

Le chapitre 4 sera consacré à l'implémentation et Simulations des différents protocoles de routage étudiés, afin de les évaluer et interpréter selon les métriques utilisés.

Enfin on termine ce mémoire par une conclusion générale.

# Chapitre I : État de l'art sur les réseaux mobiles Ad Hoc

## 1. Introduction :

L'évolution récente de la technologie dans le domaine de la communication sans fil et l'apparition des unités de calcul portables poussent aujourd'hui les chercheurs à faire des efforts à fin de réaliser le but des réseaux : [2]

*« L'accès à l'information n'importe où et n'importe quand ».*

Le concept des réseaux mobiles ad hoc essaie d'étendre les notions de la mobilité à toutes les composantes de l'environnement.

Ici, contrairement aux réseaux basés sur la communication avec infrastructure (cellulaire), aucune administration centralisée n'est disponible, ce sont les hôtes mobiles eux-mêmes qui forment une infrastructure du réseau. Aucune supposition ou limitation n'est faite sur la taille du réseau ad hoc, le réseau peut contenir des centaines ou des milliers d'unités mobiles.

## 2. Les environnements mobiles : [3]

Les environnements mobiles permettent une grande flexibilité d'emploi. En particulier, ils permettent la mise en réseau des sites dont le câblage serait trop onéreux à réaliser dans leur totalité, voire même impossible.

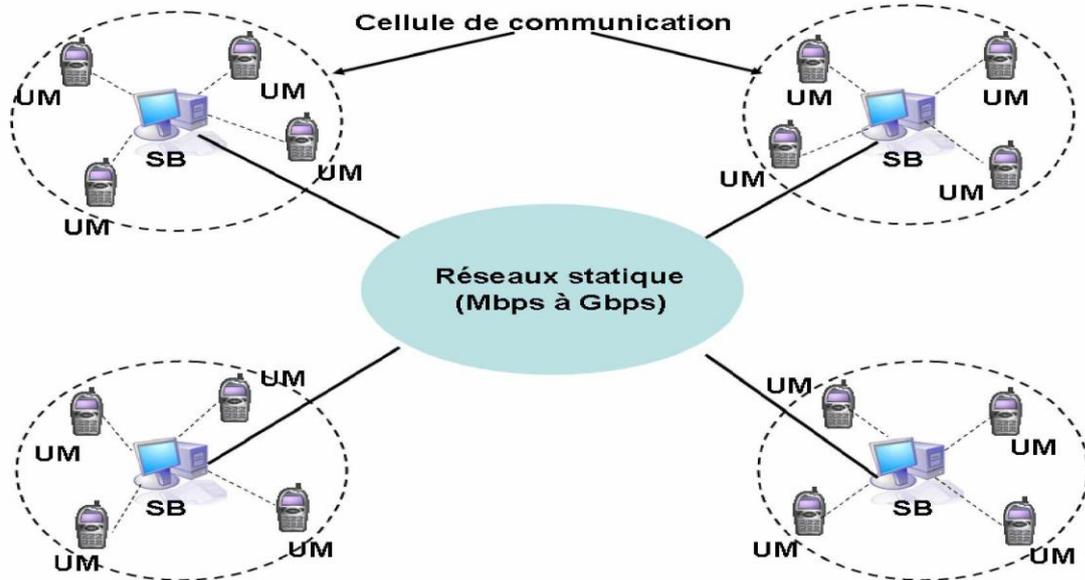
Un environnement mobile est un système composé de sites mobiles et qui permet à ses utilisateurs d'accéder à l'information indépendamment de leurs positions géographiques.

Les réseaux mobiles ou sans fil, peuvent être classés en deux classes :

- Le **réseau avec infrastructure** intègre deux ensembles d'entités distinctes :
  1. Les « sites fixes » d'un réseau de communication filaire classique (Wired network).
  2. Les sites mobiles (Wireless network)

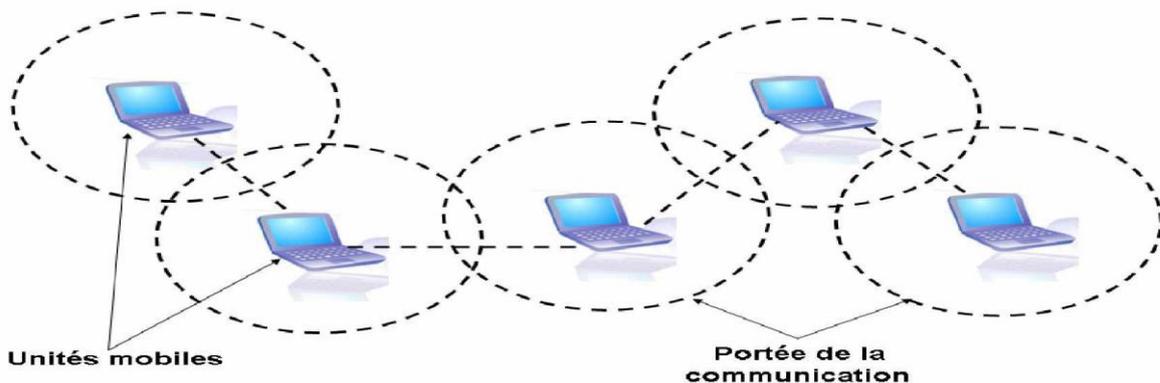
Certains sites fixes, appelés stations support mobile (Mobile Support Station) ou station de base (SB) sont munis d'une interface de communication sans fil pour la communication directe avec les sites ou unités mobiles (UM), localisés dans une zone géographique limitée, appelée cellule. A chaque station de base correspond une cellule à partir de laquelle des unités mobiles peuvent émettre et recevoir des messages. Alors que les sites fixes sont interconnectés entre eux à travers un réseau de communication filaire, généralement fiable et d'un débit élevé, les liaisons sans fil ont une bande passante limitée qui réduit considérablement le volume des informations échangées. Dans ce modèle, une unité mobile ne peut être, à un instant donné, directement connectée qu'à une seule station de base. Elle

peut communiquer avec les autres sites à travers la station à laquelle elle est directement rattachée.



*Figure 1: Le réseau avec infrastructure*

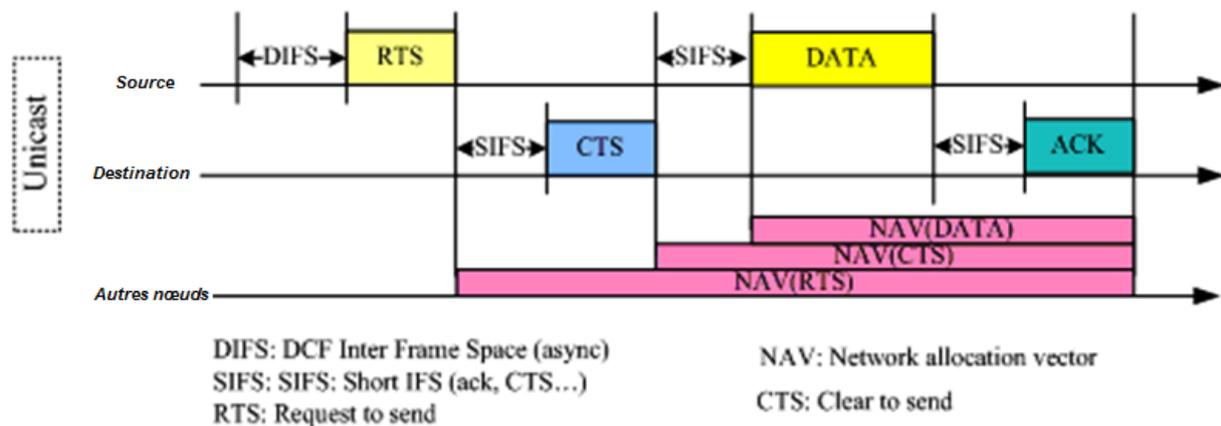
- Le modèle de **réseau sans infrastructure** préexistante ne comporte pas l'entité "site fixe" : tous les sites du réseau sont mobiles et communiquent d'une manière directe en utilisant leurs interfaces de communication sans fil. L'absence de l'infrastructure ou du réseau filaire composé des stations de base oblige les unités mobiles à se comporter comme des routeurs qui participent à la découverte et la maintenance des chemins pour les autres hôtes du réseau.



*Figure 2 : Le modèle des réseaux sans infrastructure*

### 3. L'utilisation des ondes radio mobiles et les couches : [4]

Dans un réseau sans fil, deux mobiles peuvent communiquer en émettant des ondes radios. Ils se partagent un médium unique mais ne peuvent émettre en même temps. Afin de gérer les accès concurrents à un même médium partagé le mécanisme d'accès au médium a été défini. Ce mécanisme fait partie de la famille des mécanismes de gestion des accès multiples par détection de porteuse avec évitement de collisions (Carrier Sense Multiple Access with Collision Avoidance / CSMA/CA). Il associe une technique de détection de porteuse avant transmission à une technique d'attente aléatoire permettant de limiter le nombre et l'impact des collisions. En plus, le standard définit une technique supplémentaire RTS/CTS (Request To Send/Clear To Send) pour éviter les collisions et le problème des nœuds cachés.



*Figure 3 : Processus de transmission des trames*

- La couche physique : (notée *PHY*), proposant trois types de codages de l'information (IR, DSSS, FHSS) définit la modulation des ondes radioélectriques et les caractéristiques de la signalisation pour la transmission de données
- La sous-couche MAC : la couche liaison de données est composée de deux sous-couches : le contrôle de la liaison logique (**Logical Link Control**, ou **LLC**) et le contrôle d'accès au support (**Media Access Control**, ou **MAC**). La sous-couche MAC, quant à elle, est spécifique à 802.11. Elle définit, entre autres, l'interface entre le bus de la machine et la couche physique, notamment une méthode d'accès et des règles de communication entre les différentes stations.

La partie de la norme 802.11 qui nous intéresse est la partie **MAC**. Pour la partie physique nous utiliserons un codage DSSS couplé à une antenne omnidirectionnelle.

En ce qui concerne les couches plus hautes, elles maintiennent des comportements similaires à ceux de leurs équivalents TCP/IP.

Les protocoles de routage sont associés à la couche réseau car c'est dans cette couche qu'on détermine les routes qui doivent être empruntées par les paquets de données.

Réseaux filaires		Réseaux sans fil
OSI	TCP/IP	Ad hoc
Application	Application (HTTP, FTP, TELNET,...)	Application
Présentation		
Session	Transport	Transport
Transport		
Réseau	Internet (IP)	Réseau (Mobil IP)
Liaison	Interface Réseau	Liaison de données (MAC et LLC)
Physique		Physique

**Tableau 1:** La structure en couches des réseaux mobiles Ad hoc par rapport à OSI et TCP/IP

#### 4. Les réseaux mobiles Ad hoc:

##### 4.1. Historique des réseaux mobiles Ad-Hoc :

Au début, le développement des réseaux mobiles Ad-Hoc a été le résultat de la demande du milieu militaire pour le déploiement rapide d'infrastructures de télécommunication pouvant survivre aux pannes et aux attaques. Un réseau centralisé autour de stations de base n'est pas une bonne option dans ce milieu car elles doivent être déployées en premier lieu (presque impossible dans un terrain hostile) et le réseau est vulnérable dans le cas où une ou plusieurs de ces stations de base sont détruites.

Face à ces limites, en 1972 le département de la défense américaine, en particulier DARPA (The Defense Advanced Research Projects Agency), a sponsorisé le programme de recherche PRNet (Packets Radio Network) [5]. Ce projet traitait en particulier la problématique de routage et l'accès au média dans un réseau de communication multi-sauts par onde radio.

En 1983, ce projet a évolué vers le programme SURAN (SURvivable RADIO Networks) qui traitait en particulier la problématique de la sécurité, la gestion d'énergie et la capacité de traitement [6]. Dès 1990, les ordinateurs portables ont été équipés de cartes sans fil et de ports infrarouges qui permettaient la communication directe et sans intermédiaire entre les ordinateurs portables. L'IEEE (Institute of Electrical and Electronics Engineers) adoptait alors le terme réseaux Ad-Hoc pour le standard IEEE 802.11 des réseaux locaux sans fil.

En 1994, [7] le DARPA sponsorisait les programmes GloMo (Global Mobile Information Systems) et NTDR (Near-term Digital Radio). Ces programmes avaient pour but le

développement des réseaux Ad-Hoc sans fil qui offraient un environnement de communication multimédia n'importe quand et n'importe où.

Un certain nombre de standards ont suivi ce développement des réseaux Ad-Hoc. C'est ainsi que le groupe de travail MANET (Mobile Ad-Hoc Networks) a été fondé au sein de l'IETF (The Internet Engineering Task Force). Ce groupe avait pour but d'essayer de standardiser les protocoles de routage dans les réseaux Ad-Hoc.

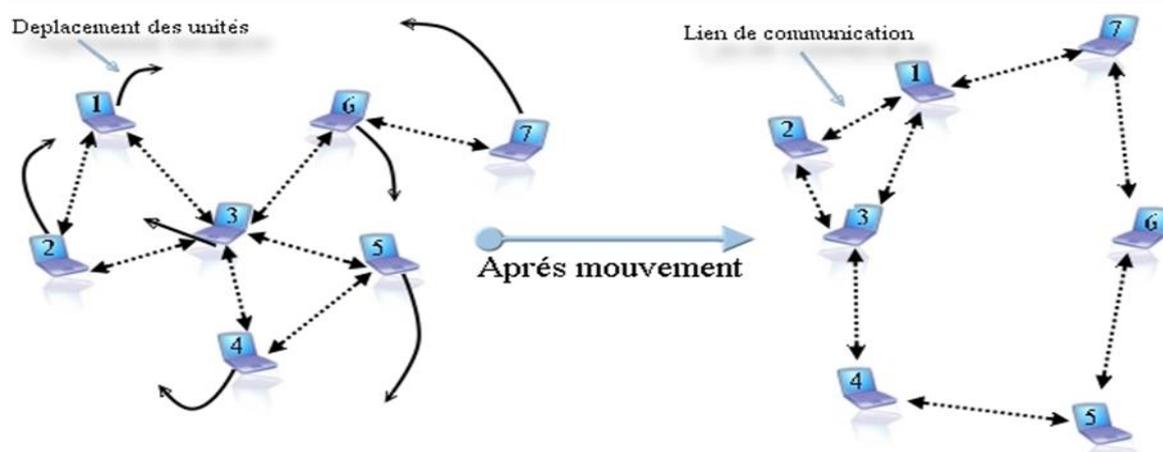
Plusieurs applications militaires et civiles ont suivi, par la suite, cette émergence des réseaux mobiles Ad-Hoc.

#### 4.2. Définition :

Les réseaux ad hoc (en latin : « **qui va vers, ce vers quoi il doit aller** », c'est-à-dire « **formé dans un but précis** », telle qu'une commission ad hoc, formée pour régler un problème particulier) sont des réseaux sans fil capables de s'organiser sans infrastructure définie préalablement [8].

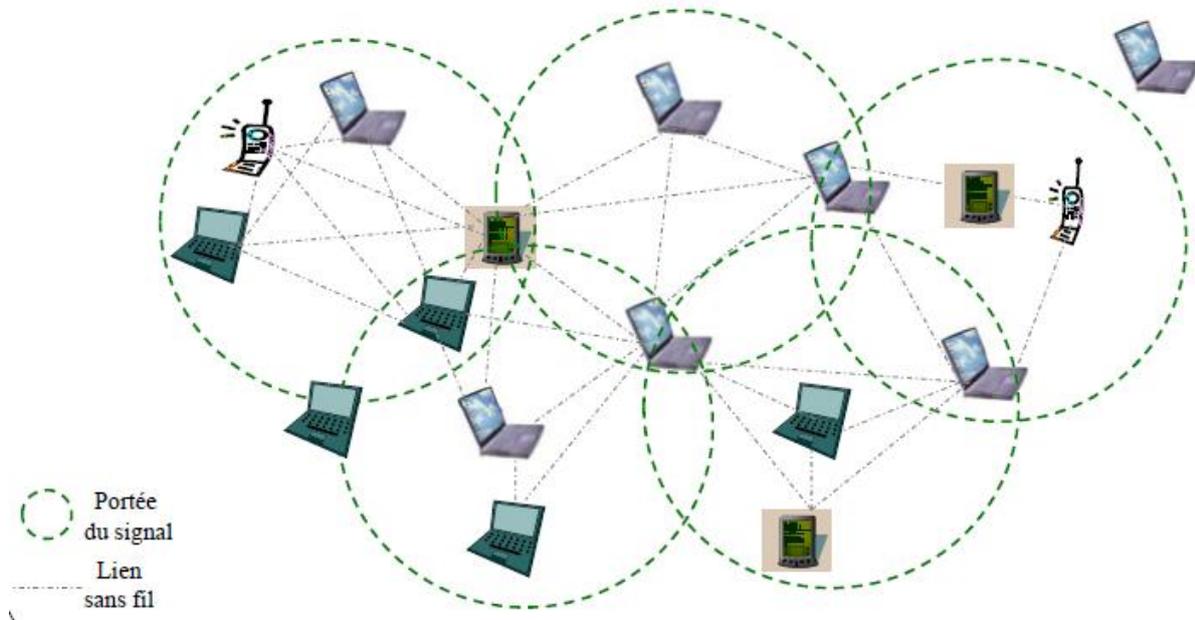
Cependant, l'IETF (Internet Engineering Task Force), qui représente l'organisme responsable de l'élaboration de standards pour Internet, définit les réseaux ad hoc de la manière suivante : Un réseau mobile ad hoc ou MANET (*Mobile Ad hoc NETWORK*) est un système autonome de plates-formes mobiles (interconnexion des différents hôtes et équipements sans fil) appelées nœuds qui sont libres de se déplacer aléatoirement et sans contrainte. Ceci provoque des changements rapides et imprédictibles de la topologie du réseau.

La topologie du réseau peut changer à tout moment (*figure 4*), elle est donc dynamique et imprévisible, ce qui fait que la déconnexion des unités est très fréquente. [8], [21]



*Figure 4: Le changement de la topologie des réseaux mobile ad hoc*

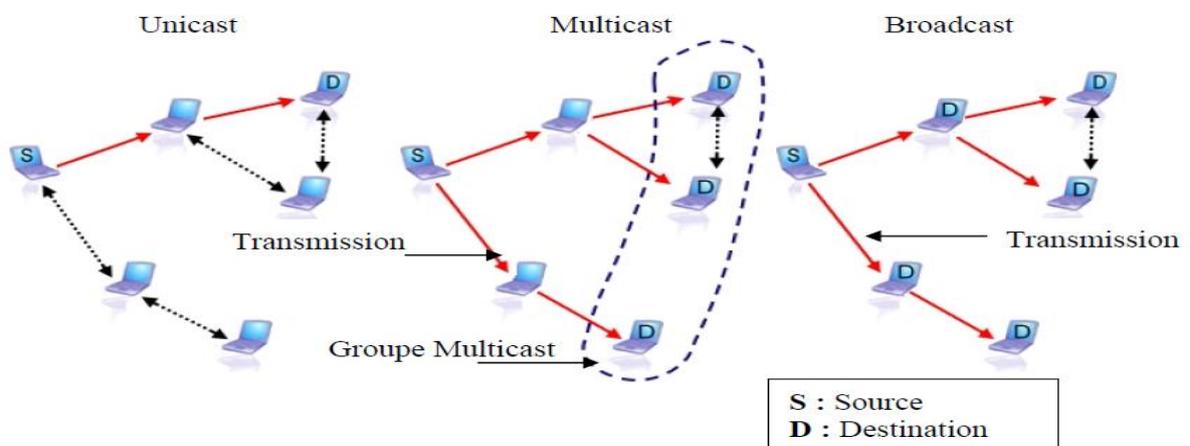
Un ensemble d'unités portables (laptop, palm, PDA, etc....) reliées entre eux par des liaisons sans fil est un bon exemple d'un réseau ad hoc. Chaque lien indique qu'une liaison directe existe entre les deux extrémités de ce lien.



*Figure 5: Exemple d'un réseau Ad-Hoc*

#### 4.3. Modes de communication dans les réseaux mobile Ad Hoc : [9]

La communication dans les réseaux mobiles Ad Hoc utilise plusieurs modes dont : la communication « point à point » ou « Unicast », la communication « multipoint » ou « Multicast », et la diffusion « Broadcast ». Ces trois modes de communication peuvent être schématisés par la figure.



*Figure 6: Les différents modes de communication*

#### 4.4. Les caractéristiques des réseaux ad hoc :

Comparativement aux réseaux fixes traditionnels, les réseaux ad hoc permettent de réduire considérablement les coûts aussi bien de déploiement que de maintenance des infrastructures grâce à une auto-organisation du réseau. Ils présentent en contre partie des contraintes additionnelles :

- **Absence d'infrastructure** : Les réseaux ad hoc se distinguent des autres réseaux mobiles par la propriété d'absence d'infrastructures préexistante et de tout genre d'administration centralisée. Les hôtes mobiles sont responsables d'établir et de maintenir la connectivité du réseau d'une manière continue.
- **Une topologie dynamique** : La topologie des réseaux ad hoc change rapidement, et aléatoirement, ceci est causée par la mobilité arbitraire des nœuds du réseau. Le changement de la topologie change les routes entre les nœuds et provoque la perte de paquets.
- **Une bande passante limitée** : Une des caractéristiques primordiales des réseaux basés sur la communication sans fil est l'utilisation d'un médium de communication partagé. Ce partage fait que la bande passante réservée à un hôte soit modeste.
- **Des contraintes d'énergie** : Les hôtes mobiles sont alimentés par des sources d'énergie autonomes comme les batteries ou les autres sources consommables.
- **Une sécurité physique limitée** : Les réseaux mobiles ad hoc sont plus touchés par le paramètre de sécurité, que les réseaux filaires classiques. Cela se justifie par les contraintes et limitations physiques qui font que le contrôle des données transférées doit être minimisé.
- **L'absence d'infrastructure** : Les réseaux ad hoc se distinguent des autres réseaux mobiles par la propriété d'absence d'infrastructures préexistante et de tout genre d'administration centralisée. Les hôtes mobiles sont responsables d'établir et de maintenir la connectivité du réseau d'une manière continue.
- **Interférences** : Les liens radios ne sont pas isolés, deux transmissions simultanées sur une même fréquence ou, utilisant des fréquences proches peuvent interférer.

#### 4.5. Les applications des réseaux mobiles ad hoc : [11]

Les communications et les opérations dans le domaine militaire sont les premières applications des réseaux ad hoc. Cependant, avec l'avancement des recherches dans le domaine des réseaux et l'émergence des technologies sans fil (ex : Bluetooth, IEEE 802.11); d'autres applications civiles sont apparues. On distingue:

- **Les services d'urgences** : opération de recherche et de secours des personnes, tremblement de terre, inondation, feux, dans le but de remplacer des infrastructures filaires.
- **Le travail collaboratif et les communications dans des entreprises ou bâtiments** : dans le cadre d'une réunion ou d'une conférence par exemple.
- **Home network** : partage d'applications et communications des équipements mobiles.
- **Réseaux de senseurs** : pour des applications domestiques (contrôle des équipements à distance) ou environnementales (climat, activité de la terre, suivi des mouvements des animaux, etc...).
- **Réseaux en mouvement** : informatique embarquée et véhicules communicants.
- **Réseaux Mesh** : c'est une technologie émergente qui permet d'étendre la portée d'un réseau ou de le densifier.

## 5. Conclusion :

Ce chapitre a été consacré sur les concepts des environnements mobiles plus particulièrement sur les réseaux mobiles ad hoc, et l'utilisation de la technologie de communication sans fil.

Ces environnements sont caractérisés par de fréquentes déconnexions des nœuds et des restrictions sur les ressources utilisées, surtout si tous les usagers du système sont mobiles ce qui est le cas pour les réseaux mobiles ad hoc. Ces limitations transforment certains problèmes, ayant des solutions évidentes dans l'environnement classique, en des problèmes complexes et difficiles à résoudre.

Bien que ces réseaux présentent des avantages énormes, mais malheureusement beaucoup de problèmes restent à résoudre, notamment le problème du routage que nous verrons dans le chapitre suivant.

# Chapitre II : Le routage dans les réseaux mobiles Ad hoc et les protocoles de routage

## 1. Introduction :

Avec une topologie évoluant constamment en fonction des mouvements des mobiles, le problème qui se pose dans le contexte des réseaux ad hoc mobiles est l'adaptation de la méthode d'acheminement utilisée avec le grand nombre d'unités existant dans un environnement caractérisé par de modestes capacités de calcul et de sauvegarde.

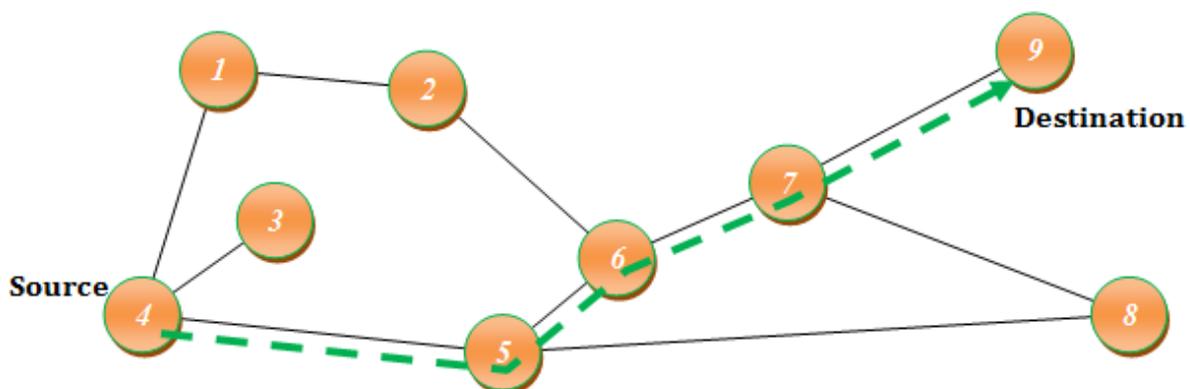
Il se peut qu'un hôte destination soit hors de la portée de communication d'un hôte source, ce qui nécessite l'emploi d'un routage interne par les nœuds intermédiaires afin de faire acheminer les paquets de message à la bonne destination.

La gestion de ce routage consiste à établir une sorte d'architecture globale où l'on doit tenir compte de la mobilité des nœuds et de la versatilité du médium physique.

Le problème de routage est loin d'être évident dans les réseaux mobiles ad hoc, où c'est difficile de localiser la destination à un instant donné. La conception des stratégies de routages doit tenir compte de tous les facteurs et limitations physiques imposés par l'environnement dans le but de concevoir un protocole de routage qui ne dégrade pas les performances du système.

## 2. Définition :

Le routage est une méthode d'acheminement des informations à la bonne destination à travers un réseau de connexions défini. Son intérêt consiste à trouver le chemin optimal au sens d'un certain critère de performance (bande passante, délai, etc.). Il doit aussi être capable de s'adapter aux événements venant perturber le réseau (panne, congestion, etc...). [12]



*Figure 7: Le chemin utilisé dans le routage entre la source et la destination*

### **3. Problématiques de Routage dans les Réseaux Ad-Hoc :**

Comme nous avons déjà vu, l'architecture d'un réseau mobile ad hoc est caractérisée par une absence d'infrastructure fixe préexistante, à l'inverse des réseaux de télécommunication classiques. Un réseau ad hoc doit s'organiser automatiquement de façon à être déployable rapidement et pouvoir s'adapter aux conditions de propagation, au trafic et aux différents mouvements pouvant intervenir au sein des unités mobiles.

Dans le but d'assurer la connectivité du réseau, malgré l'absence d'infrastructure fixe et la mobilité des stations, chaque nœud est susceptible d'être mis à contribution pour participer au routage et pour retransmettre les paquets d'un nœud qui n'est pas en mesure d'atteindre sa destination ; tout nœud joue ainsi le rôle de station et de routeur.

Le problème qui se pose dans le contexte des réseaux mobiles ad hoc est l'adaptation de la méthode d'acheminement utilisée avec le grand nombre d'unités existant dans un environnement caractérisé par de modestes capacités de calcul et de sauvegarde. [21]

Dans la pratique, il est impossible qu'un hôte puisse garder les informations de routage concernant tous les autres nœuds, dans le cas où le réseau serait volumineux.

Le problème ne se pose pas dans le cas de réseaux de petites tailles, car l'inondation faite dans ces réseaux n'est pas coûteuse ; par contre dans un réseau volumineux, le manque de données de routage concernant les destinations peut impliquer une diffusion énorme dans le réseau, et cela si on considère seulement la phase de découverte de routes.

Le trafic causé par la diffusion, dans ce cas, est rajouté au trafic déjà existant dans le réseau ce qui peut dégrader considérablement les performances de transmission du système caractérisé principalement par une faible bande passante.

Dans le cas où le nœud destination se trouve dans la portée de communication du nœud source le routage devient évident et aucun protocole de routage n'est initié.

Malheureusement, ce cas est généralement rare dans les réseaux mobiles ad hoc.

Une station source peut avoir besoin de transférer des données à une autre station (nœud intermédiaire) qui ne se trouve pas dans sa portée de communication ce qui nécessite un protocole de routage approprié. [12], [13]

## 4. Généralités sur le routage :

Le problème du routage ne se résume pas seulement à trouver un chemin entre les deux nœuds du réseau, mais encore à trouver un acheminement optimal et de qualité des paquets de données.

### 4.1. Routage par la source : (*source routing*)

C'est une technique de routage dans laquelle l'émetteur du paquet (source) détermine la séquence complète de nœuds à travers laquelle le paquet doit passer, l'émetteur met explicitement le chemin dans l'entête en identifiant chaque saut (hop) par l'adresse du prochain nœud à qui on doit envoyer le paquet.

### 4.2. Le multihopping : (*multi-sauts*)

Dans les communications de type cellulaire les communications passent par des stations dites de base et un réseau filaire : les stations mobiles ne servent jamais de routeurs intermédiaires, ce modèle est donc dit Single Hop. Dans un modèle de communication sans infrastructures de ce type (le cas des réseaux ad-hoc), les nœuds participent au routage : le modèle est dit multihop. [14]

### 4.3. La méthode du vecteur de distance : (*Distance Vector*)

Dans cette méthode, chaque nœud diffuse à ses nœuds voisins sa vision des distances qui le séparent de tous les hôtes du réseau. En se basant sur les informations reçues par tous ses voisins, chaque nœud de routage fait un certain calcul pour trouver le chemin le plus court vers n'importe quelle destination. Le processus de calcul se répète, s'il y a un changement de la distance minimale séparant deux nœuds, et cela jusqu'à ce que le réseau atteigne un état stable. Cette technique est basée sur l'algorithme distribué de Bellman-Ford. [14]

### 4.4. La méthode état de lien : (*Link State*)

Dans cette méthode, chaque nœud garde une vision de toute la topologie du réseau, et ce, par l'intermédiaire des requêtes périodiques portant sur l'état des liaisons avec les nœuds voisins. En effet la mise à jour dans cette méthode se fait pour chaque nœud diffusant l'état des liens des nœuds voisins dans le réseau. Cette opération est aussi faite en cas de changement dans l'état des liens. [14]

## 5. La conception des stratégies de routage :

L'étude et la mise en œuvre d'algorithmes de routage pour assurer la connexion des réseaux ad hoc au sens classique du terme (tout sommet peut atteindre tout autre), est un problème complexe. L'environnement est dynamique et évolue donc au cours du temps, la topologie du

réseau peut changer fréquemment. Il semble donc important que toute conception de protocole de routage doive étudier les problèmes suivants : [15]

✚ **La minimisation de la charge du réseau :**

L'optimisation des ressources du réseau renferme deux autres sous problèmes qui sont l'évitement des boucles de routage, et l'empêchement de la concentration du trafic autour de certains nœuds ou liens.

✚ **Offrir un support pour effectuer des communications multipoints fiables :**

Le fait que les chemins utilisés pour router les paquets de données puissent évoluer, ne doit pas avoir d'incident sur le bon acheminement des données. L'élimination d'un lien, pour cause de panne ou pour cause de mobilité devrait, idéalement, augmenter le moins possible les temps de latence.

✚ **Assurer un routage optimal :**

La stratégie de routage doit créer des chemins optimaux et pouvoir prendre en compte différentes métriques de coûts (bande passante, nombre de liens, ressources du réseau, délais de bout en bout,...etc..). Si la construction des chemins optimaux est un problème dur, la maintenance de tels chemins peut devenir encore plus complexe, la stratégie de routage doit assurer une maintenance efficace de routes avec le moindre coût possible.

✚ **Le temps de latence :** La qualité des temps de latence et de chemins doit augmenter dans le cas où la connectivité du réseau augmente.

## 6. Le routage réactif, proactif (*unicast*) dans les réseaux mobiles ad-hoc :

Le routage est la tâche d'acheminement de flux des données à partir des nœuds sources vers les nœuds destinations. Si une seule destination est impliquée dans la communication, alors il s'agit d'un *routage unicast*; si encore tous les nœuds du réseau ou juste un sous ensemble sont concernés par la réception des données alors on parle du *broadcast* et du *routage multicast*, respectivement.

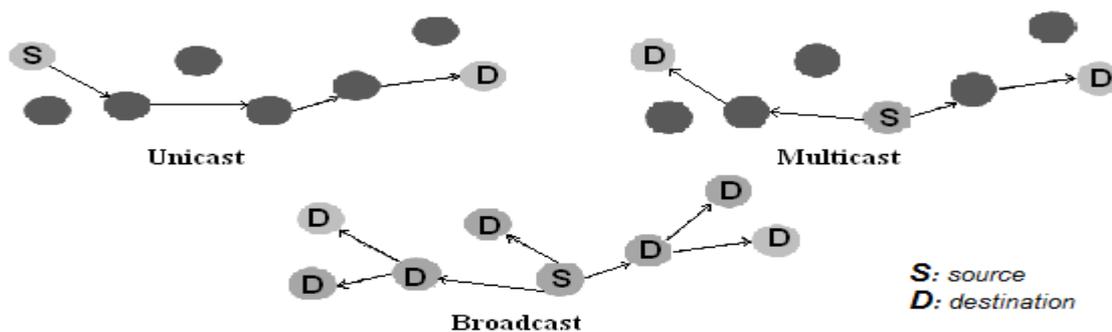


Figure 8 : Illustration du routage unicast, multicast et broadcast

Suivant la manière de création et de maintenance de routes lors de l'acheminement des données, on peut distinguer deux principaux types de routage, le routage proactifs et le routage réactifs. Le routage proactif est basé sur l'établissement des routes à l'avance et en se basant sur l'échange périodique des tables de routage, alors que le routage réactif se base sur la recherche des routes à la demande.

Une troisième classe existe, celle de routage hybride. Ce type de routage utilise, simultanément, un routage proactif et un routage réactif, afin de combiner les avantages des deux approches.

Il semble a priori normal d'avoir différents types de routage car les scénarios d'usage des réseaux mobiles ad hoc sont aussi très variés.

Dans ce qui suit, nous allons nous intéresser aux routages *réactif* et *proactif* en *Unicast* dans les réseaux mobiles ad hoc. Nous décrirons leurs principales caractéristiques et fonctionnalités qui permettent d'assurer l'acheminement des données entre les différentes unités mobiles.

Le routage unicast est une méthode d'acheminement qui consiste à trouver un chemin pour la transmission d'un message à partir d'un *seul nœud source* vers une *seule destination*. [18]

Pour offrir une connectivité plus étendue au sein d'un réseau mobile et auto-organisé, comme les MANETs, il faut créer dynamiquement une connectivité multi sauts entre un ensemble de nœuds sans fil qui peuvent être en mouvement. Ce besoin d'un protocole de routage pour trouver ces routes « *multi sauts* » est un véritable challenge face aux interférences et aux limitations de puissance.

La normalisation MANET fait apparaître plusieurs protocoles de routage différents, répartis en deux grandes familles : les protocoles de routage réactif (basés sur l'inondation du réseau) et les protocoles de routage proactif (basés sur une découverte topologique du réseau).

### **6.1. Routage proactif :**

Les protocoles de cette catégorie sont basés sur les algorithmes classiques d'état de liens et de vecteur de distance déjà présentés dans la section précédente. Le principe de base de ces protocoles est de maintenir à jour les tables de routage, et ce par la mise en place d'un système d'échange périodique des paquets de contrôle. Cette manière de procéder permet aux nœuds de construire de façon distribuée la topologie du réseau.

Il existe, à cet effet, deux types de paquets de contrôle : [19] les paquets envoyés localement (à un saut) pour la découverte du voisinage et les paquets diffusés dans tout le réseau pour communiquer aux autres nœuds les informations sur l'état du voisinage (généralement

l'ensemble des voisins ou un sous ensemble) rassemblés par le premier type de messages de contrôle.

Lorsqu'un nœud reçoit un paquet de contrôle, il met à jour ses tables de routages. Ainsi, de nouvelles routes seront construites sur la base des informations topologiques transportées par les trames de contrôle. Ce processus est déclenché aussi à chaque changement de topologie pour reconstruire à nouveau les routes.

La caractéristique principale de ces protocoles est la disponibilité immédiate de la route lors du besoin. Les principaux protocoles proactifs sont : OLSR (*Optimized Link State Routing*) et DSDV (*Dynamic Destination-Sequenced Distance Vector*).

## 6.2. Routage réactif :

Les protocoles de routage proactifs engendrent un trafic très important ce qui conduit, souvent, à la saturation rapide du réseau. Pour y remédier, les protocoles réactifs évitent au maximum les inondations qui consomment beaucoup de ressources.

L'idée de cette approche est de lancer le processus de recherche de routes uniquement au besoin (à la demande). Ainsi, quand un nœud demande de relayer ses données vers une destination quelconque dont il ne dispose pas de route valide, le mécanisme de recherche de route est déclenché. Le principe est le suivant :

- ✚ le nœud émetteur diffuse une requête « Route Request RREQ » au niveau de son rayon de propagation.
- ✚ le mécanisme d'inondation permet à cette requête de se propager sur tout le réseau.
- ✚ à chaque fois que le paquet passe par un nœud mobile, jouant dans ce cas le rôle d'un routeur, une information portant l'identifiant du nœud est ajoutée à la route jusqu'à ce que le paquet atteigne sa destination.
- ✚ à la réception du paquet, le nœud destinataire renvoie le tracé du chemin à la source en suivant le chemin inverse.
- ✚ parmi toutes les routes renvoyées par le nœud destinataire, le nœud source sélectionne la plus petite et il la sauvegarde dans son cache afin de l'utiliser en cas de besoin.

Cependant, du fait que l'on ne dispose pas immédiatement de la route vers la destination, le délai nécessaire à l'acheminement des paquets vers la destination est plus important en comparaison avec les protocoles proactifs.

Les principaux protocoles réactifs sont : AODV (*Ad Hoc On demand Distance Vector Routing*) et DSR (*Dynamic Source Routing*).

## 7. Les protocoles de routage dans les réseaux mobiles ad hoc :

Les protocoles décrits par la suite sont issus du groupe de travail MANET de l'IETF. Ces protocoles sont représentatifs de diverses techniques et sont les plus avancés sur la voie d'une normalisation. *AODV* et *OLSR* font désormais l'objet d'une Request For Comment (*RFC*).

Le choix porté sur ces protocoles est motivé par le fait que ces protocoles soient standardisés.

### 7.1. Ad hoc On demande Distance Vector (AODV) :

Spécifié par la RFC 3561, AODV (*Ad hoc On Demande Vector*) [5] est un protocole de routage réactif spécifique, comme son nom l'indique, aux réseaux mobiles ad hoc. Il est basé sur le routage de vecteur de distance en minimisant sensiblement le nombre de diffusions de messages en créant le chemin à la demande, c'est-à-dire qu'il ne construit de route entre nœuds que lorsqu'elle est demandée par un nœud source, ce nœud la maintient durant le temps qu'il en fait usage. Le protocole utilise le principe des numéros de séquence permettant aux nœuds d'utiliser les routes les plus fraîches.

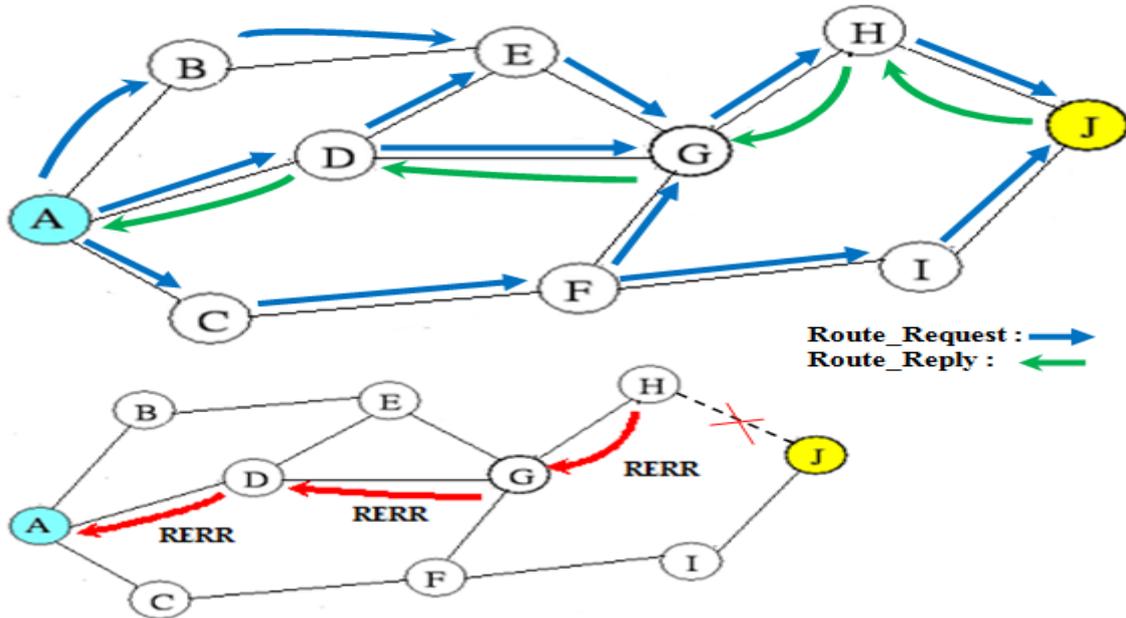
L'établissement et le maintien des routes sont assurés par l'échange de différents types de messages :

- ✓ Route Request(**RREQ**) : “*J'ai besoin d'une route*”, message diffusé à tous les nœuds voisins par une source désirant envoyer des paquets de données vers une destination.
- ✓ Route Reply(**RREP**): “*Annoncer la route*”, une fois la destination reçoit le *RREQ*, elle répond par un *RREP* comme accusé de réception, chemin inverse de *RREQ*.
- ✓ **Hello** message : “*Vous êtes là?*”, message diffusé périodiquement vers le nœud immédiatement voisin pour voir s'il est encore là, s'il n'y a pas de message Hello qui arrive d'un nœud particulier, le voisin suppose que ce nœud est déplacé et marque ce lien comme interrompu.
- ✓ Route Error (**RERR**): “*Annuler la route*”, message envoyé par un nœud lorsqu'il détecte que la liaison avec son voisin est rompue (route invalide).

Une entrée de la table de routage contient essentiellement :

- ✚ Adresse du nœud destination : c'est l'adresse IP du nœud destinataire à atteindre.
- ✚ Adresse du nœud suivant : l'adresse IP du nœud auquel on va envoyer un paquet à router pour joindre une destination.
- ✚ La distance en nombre de nœud (nombre de nœud nécessaire pour atteindre la destination).
- ✚ Le numéro de séquence destination qui garantit qu'aucune boucle ne peut se former.
- ✚ Durée de vie pour laquelle la route reste à la disposition du nœud source.

- Liste des voisins qui utilisent cette route : adresses IP d'éventuels nœuds précurseurs qu'utilise le nœud courant comme un prochain saut pour atteindre la destination.



Fi

Figure 9: Requêtes pour la découverte de routes et lors de la rupture de lien dans AODV

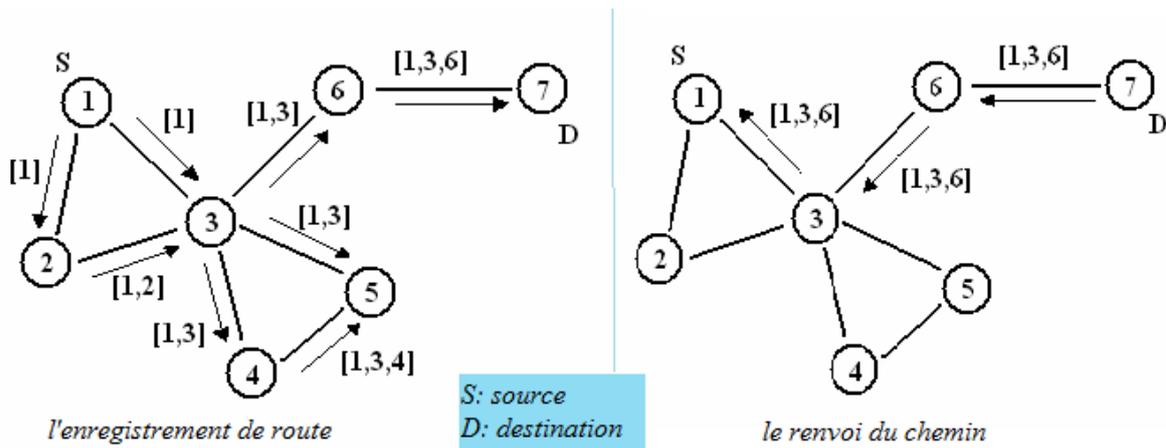
## 7.2. Dynamic Source Routing (DSR) :

Le protocole Routage à Source Dynamique (*DSR : Dynamic Source Routing*) est un protocole de routage réactif unicast, efficace et dédié aux réseaux ad hoc mobile multi sauts. Ce protocole est basé sur l'utilisation de la technique "routage source". Avec cette technique, la source des données détermine la séquence complète des nœuds à travers lesquelles, les paquets de données seront envoyés.

*Découverte de la route* : Quand une source n'a pas de route vers une destination, elle envoie un **Route\_Request**, chaque nœud intermédiaire recevant ce message lui ajoute son identificateur ID et le renvoie. Dès que le **Route\_Request** atteint la destination (ou bien une station intermédiaire ayant une route vers la destination), elle détermine une route complète de la source à la destination. La destination envoie un message **Route\_Reply** contenant la route sélectionnée vers la source. Ce **Route\_Reply** peut être envoyé sur la route trouvée (les liens sont bidirectionnels) ou bien incorporé dans un autre **Route\_Request** mais cette fois de la destination vers la source. Quand la source reçoit le **Route\_Reply**, elle enregistre cette route dans sa mémoire et l'inclut dans les entêtes de chaque paquet de données. Les nœuds intermédiaires acheminent les paquets selon la route spécifiée dans les entêtes.

*Maintenance de la route* : Si une panne d'un lien est détectée, un paquet peut être sauvé par un nœud intermédiaire qui remplace la route brisée par une autre, et un message **Route\_Error** est

envoyé vers la source. Le système de déduction est aussi utilisé pour les messages **Route\_Error**, ainsi les nœuds voisins ou intermédiaires peuvent retirer (ou remplacer) les routes défectueuses de leurs tampons (ceci minimise le risque de présence de routes invalides). Les changements de topologie peuvent mener à la formation de routes plus courtes, ainsi les nœuds intermédiaires peuvent envoyer à la source un **Route\_Reply** lui indiquant la route la plus courte.



**Figure 10:** Découverte de la route dans le protocole DSR

### 7.3. Dynamic Destination-Sequenced Distance Vector (DSDV):

Le protocole de routage « DSDV » : signifie « Vecteur de Distance à Dynamique Séquencée » basé sur l'idée classique de l'algorithme distribué de Bellman-Ford en rajoutant quelques améliorations.

Chaque station mobile maintient une table de routage qui contient :

- ✚ Toutes les destinations possibles.
- ✚ Le nombre de nœud (ou de sauts) nécessaire pour atteindre la destination.
- ✚ Le numéro de séquences (SN) qui correspond à un nœud destination.

Le NS est utilisé pour faire la distinction entre les anciennes et les nouvelles routes, ce qui évite la formation des boucles de routage.

La mise à jour dépend donc de deux paramètres : le temps, c'est à dire la période de transmission, et les événements qui peuvent surgir (déplacement de nœuds, apparition d'un nouveau voisin ...etc.)

Un paquet de mise à jour contient :

- ✚ Le nouveau numéro de séquence incrémenté, du nœud émetteur.
- Et pour chaque nouvelle route :
- ✚ L'adresse de la destination.
- ✚ Le nombre de nœuds (ou de sauts) séparant le nœud de la destination.

- ✚ Le numéro de séquence (des données reçues de la destination) tel qu'il a été estampillé par la destination.

Le DSDV élimine les deux problèmes de boucle de routage "routing loop", et celui du "counting to infinity".

Cependant dans ce protocole, 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, dans la table de distance. Ce qui fait que le DSDV est lent.

Le DSDV utilise une mise à jour périodique et basée sur les événements, ce qui cause un contrôle excessif dans la communication.

#### 7.4. Optimized Link State Routing (OLSR):

OLSR (*Optimized Link State Routing*) [13] spécifié dans la RFC 3626, il s'agit, comme son nom l'indique, d'une optimisation du protocole à état de lien, conçu pour être adapté aux spécifications d'un réseau mobile ad hoc, et pour fonctionner dans un environnement complètement distribué sans aucune entité centrale.

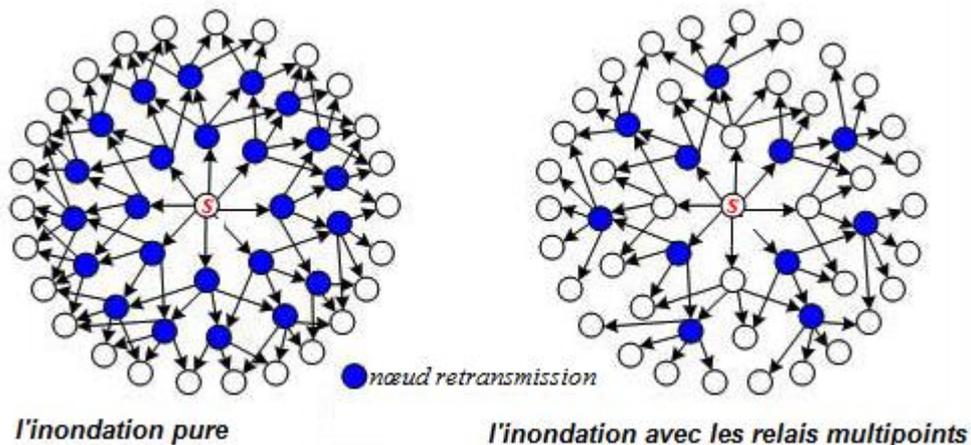
Dans un protocole à état de lien classique, un nœud envoie à tous les autres nœuds une information au sujet de ses voisins. Donc, chaque station possède une copie complète de la carte du réseau et exécute un calcul local des meilleures routes. Par contre, dans le cas d'OLSR les nœuds ne déclarent qu'une sous partie de leur voisinage grâce à l'utilisation de la technique des Relais MultiPoints (**MPR**). Le principe est que chaque station construit le plus petit sous-ensemble, parmi ses voisins, qui permet d'atteindre tous ses voisins à deux sauts.

Les nœuds de cet ensemble servent à acheminer et retransmettre les messages qu'ils reçoivent. Ceci permet de minimiser les messages (type broadcast) échangés dans le réseau.

Le processus de construction des routes dans OLSR passe par les étapes suivantes :

- ✚ Détection du voisinage : La découverte du voisinage est la première étape dans le processus de découverte de topologie. Elle est basée sur l'échange périodique des messages **HELLO**. Pendant cette phase, chaque nœud tente de recenser l'état des liens qui le relie à ses voisins à un saut et vérifie la validité de ceux-ci.
- ✚ Élections des MPRs : La liste des **MPR** d'un nœud n est définie comme le plus petit sous ensemble de ses voisins à un saut qui couvre tous ses voisins à deux sauts. Le reste des voisins, considérés comme redondants pour le calcul du plus court chemin, sont ignorés. Le mécanisme de calcul de cet ensemble doit faire face aux changements fréquents de la topologie du réseau de manière à permettre de recalculer cette liste à chaque changement dans le voisinage d'un nœud.

- ✚ Annonce des *MPRs* et diffusion des voisinages : En plus des messages *HELLO* utilisés pour découvrir et construire la liste des *MPRs* de tous les nœuds du réseau, le protocole OLSR utilise un autre type de message pour le contrôle de la topologie appelé *TC* (Topology Control). Chaque nœud envoie, périodiquement, le paquet *TC* pour déclarer la liste des liens vers les nœuds de son ensemble *MPRs*. Les informations transportées dans le message *TC* permettent de construire une carte du réseau contenant tous les nœuds et un ensemble partiel de liens, mais suffisant pour la construction de la table de routage. Ces informations seront stockées dans une table appelée Table de Topologie. Cette table est mise à jour à chaque réception d'un paquet *TC*, et sert pour la construction des tables de routages.
- ✚ Calcul des tables de routages : Après avoir détecté son voisinage, et choisi et annoncé son ensemble des *MPR*, chaque nœud procède au calcul des routes sur la base de ses informations stockées localement dans des tables. Ces routes sont maintenues dans la table de routage du nœud. Chaque modification de topologie ou dans le voisinage est répercuté sur le calcul des routes et implique, ainsi, la mise à jour des tables de routage.



**Figure 11:** les Relais MultiPoints (*MPR*)

La Figure 11 donne un exemple de gain en nombre de retransmissions sur un graphe simple. Supposons qu'un nœud émette un message, et que pour diffuser cette information au réseau ses voisins répètent cette information. Dans le premier graphique, à gauche, où tous les voisins d'un nœud retransmettent, vingt-et-deux répétitions (les nœuds en Bleu) sont nécessaires. Par contre, en utilisant la retransmission par les relais multipoints seuls (à droite), on économise onze retransmissions.

## 8. Tableaux récapitulatifs et Comparaison de performances :

Les différentes techniques ou protocoles de routage sont en fait différents moyens adaptés à une situation de topologie dynamique, d'effectuer une «distribution des chemins» en limitant les sauts (chemin optimal), en évitant les boucles (très néfastes car causant des pertes de paquets au sein du réseau) et en minimisant l'overhead (données supplémentaires liées au contrôle de transfert et à la correction d'erreur). Toutes ces solutions présentent des profils différents avec chacune leurs avantages et leurs inconvénients suivant le profil des réseaux et leur utilisation. [19]

Les performances de chaque type de routage sont dépendantes de différents paramètres du réseau, tels que le nombre de nœud (taille du réseau), le nombre de liens reliant chaque nœud (densité du réseau) et la disponibilité du réseau (probabilité qu'un lien connu soit en effet fonctionnel). Pour nuancer ce jugement, et mieux comprendre l'intérêt de ces différentes méthodes de routage, il faut mener des simulations ou effectuer des mesures expérimentales.

	Avantages	Inconvénients
Proactif	<ul style="list-style-type: none"> <li>• Pas de temps de réaction</li> <li>• Adaptés aux réseaux denses de taille moyenne</li> <li>• Adaptés aux réseaux à forte mobilité</li> </ul>	<ul style="list-style-type: none"> <li>• Trafic de contrôle important</li> <li>• Capacité d'échange du réseau limitée</li> <li>• Forte Consommation énergétique</li> </ul>
Réactif	<ul style="list-style-type: none"> <li>• Trafic de contrôle faible</li> <li>• Adaptés aux grands réseaux</li> <li>• Consommation énergétique réduite</li> </ul>	<ul style="list-style-type: none"> <li>• Temps de réaction long</li> <li>• Problème en cas de forte mobilité des nœuds</li> </ul>

**Tableau 2 :** Avantages et Inconvénients des routages proactif et réactif [17]

	Approche proactive	Approche réactive
Disponibilité de chemins	Toujours disponibles	À la demande.
Coût de routage	Élevé, mais il existe des tentatives de réduction	Inférieure
MAJ périodiques	Oui	Non
Gestion de la mobilité	Généralement des MAJ à des intervalles fixes	Reconstruction de chemins initiée par la source généralement
Espace de stockage requis	Élevé.	Généralement inférieur qu'en proactive ; Dépend du nombre de chemins actifs.
Délais	Les chemins sont prédéfinis.	Plus élevé qu'en proactive.
Niveau de scalabilité	Jusqu'à 100 nœuds	En cas de routage source, jusqu'à quelques centaines de nœuds. Pour le routage saut-par-saut, un plus haut niveau de scalabilité peut être atteint.

**Tableau 3 :** Comparaison des performances des routages proactif et réactif [18]

## 9. Conclusion :

On peut dire explicitement que chaque type de routage proactif ou réactif est plus ou moins adapté à un type de réseau. Il convient donc de définir précisément les caractéristiques du réseau (taille, mobilité des nœuds, ressources des nœuds, volume d'information à échanger...) afin de bien choisir un protocole de routage adapté.

Les protocoles des deux catégories (proactifs et réactifs) essaient de s'adapter aux contraintes imposées par le réseau ad hoc, et cela en proposant une méthode qui soit de moindre coût en capacités et ressources, et qui garantit la survabilité du routage en cas de panne de lien ou de nœuds.

Les protocoles de routage étudiés offrent différents avantages qui sont en réalité complémentaires et préférables pour différents types d'applications.

La conclusion qu'on peut tirer de l'étude des différentes stratégies est que la conception d'un protocole de routage pour les réseaux ad hoc doit tenir compte de tous les facteurs et limitations physiques imposés par l'environnement afin que la stratégie de routage ne dégrade pas les performances du réseau.

---

## Chapitre III : Implémentation Et Simulations

### 1. Introduction :

Les constituants des réseaux mobiles ad hoc, à savoir les protocoles, les applications ou la modélisation du réseau, doivent être évalués afin de mesurer les performances de la stratégie utilisée et de tester sa fiabilité avant d'être utilisés dans de vraies applications. L'utilisation d'un réseau mobile ad hoc réel dans une évaluation est difficile et coûteuse, en outre de telles évaluations ne donnent pas généralement des résultats significatifs. Le réseau réel n'offre pas la souplesse de varier les différents paramètres de l'environnement et pose en plus le problème d'extraction de résultats; c'est pour cela la majorité des travaux d'évaluation de performances utilisent le principe de simulation vu les avantages qu'il offre.

L'objectif du chapitre est de déterminer l'existence d'une meilleure stratégie de routage qui apporte le plus de gain de performance à une adaptation à la mobilité du système.

Nous cherchons alors à déterminer quelle est la stratégie qui est la plus adaptée avec la performance du système constitué par le réseau mobile ad hoc.

En considérant que le protocole de routage est le facteur prépondérant de performance du système, nous modélisons celui-ci par un protocole d'accès standard (IEEE 802.11) et différentes options de protocoles de routages, qui sont les principaux protocoles de l'IETF (AODV, DSR, DSDV et OLSR). Nous simulons ensuite le système pour différents scénarios (avec six simulations par scénario) de mobilité aléatoire que nous moyennons pour obtenir un mouvement général, de façon à obtenir des résultats comparatifs entre les principaux protocoles.

### 2. Métrique d'étude de performances

Les métriques sont des paramètres de test du protocole de routage qui permettent de mesurer les performances de celui-ci à bases des quelles la comparaison entre les protocoles sera effectuée. Ces métriques sont évaluées en fonction de nos paramètres de simulation tels que la taille (densité) du réseau (variation du nombre de nœuds), la mobilité des nœuds du réseau (changement du temps de pause et de vitesse), le nombre de connexions (trafic) dans le réseau.

Dans notre étude comparative, nous avons utilisé les métriques de performance les plus importantes pour évaluer les performances des protocoles de routage dans les MANET à savoir :

### 2.1. Average End-to-End Delay (*Délai moyen de bout-en-bout*):

Métrique très intéressante, qui mesure le temps mis pour que les paquets arrivent à destination. En d'autres termes, c'est le temps qu'un paquet met entre la source et la destination. Un protocole a de meilleures performances s'il garantit un petit délai de bout-en-bout. On note cette métrique par délai on a :  $\text{Délai}_i = \sum_{i \in pr} \text{délai}_i / \#pr$  où

$\#pr$  : est l'ensemble des paquets reçus par tous les nœuds destinataires ;

$\text{délai}_i$  : est le délai de transfert du paquet  $i$  tel que :

$$\text{délai}_i = \text{temps d'arrivée du paquet } i - \text{temps d'envoi du paquet } i$$

Cette métrique est très importante pour étudier la qualité de service de protocole.

### 2.2. Packet Delivery Fraction (*Taux d'arrivée des paquets de données*) :

Le pourcentage de paquet délivré (PDF) est une métrique qui permet de calculer le nombre total de paquets délivrés à une destination par rapport au nombre de paquets envoyés.

Ce nombre représente l'efficacité d'un protocole dans l'envoi des données aux récepteurs prévus à travers le réseau. Cette métrique est calculée selon les formules suivantes :

$$\text{PDF} = \frac{\text{Nombre des paquets de données reçus}}{\text{Nombre des paquets de donnée envoyée}} * 100$$

### 2.3. Overhead (*Nombre de paquets de contrôle*) :

C'est le nombre de paquets de contrôle générés par le protocole de routage durant la simulation, d'une manière plus formelle on a :

$$\text{Overhead} = \sum \text{overhead}_i$$

où  $\text{overhead}_i$  est le nombre de paquet de contrôle du nœud  $i$ .

### 2.4. Normalized Routing Load:

Définit le nombre total de paquets de contrôle transmis pendant la simulation par rapport au nombre de paquets de données reçus par les destinations. Elle montre donc à quel point un protocole consomme de la bande passante avec ses messages de routage.

### 3. Environnement de simulation :

Le but général de notre expérimentation est d'une part, de confronter par la simulation, à l'aide de NS-2, les performances des quatre protocoles de routage Ad hoc: AODV, DSR, DSDV et OLSR et d'autre part, d'examiner l'impact de la charge du trafic, de la mobilité et de la densité des nœuds sur le comportement de ces protocoles.

Les résultats de chaque expérimentation est le résultat d'une moyenne de six (06) simulations indépendantes. Ceci permet d'obtenir un intervalle de confiance plus élevé. Dans chaque simulation, les positions des nœuds dans la topologie sont générées d'une manière aléatoirement et indépendante. Pour ce faire nous avons utilisé le simulateur NS-2.

#### 3.1. Le simulateur NS-2: [36]

NS-2(*Network Simulator2*) est un outil logiciel de simulation à source ouvert et à événements discrets permettant l'étude, la conception et la gestion des protocoles pour les réseaux informatiques. Il a été le fruit de la collaboration entre l'université de *Berkeley*, USC (*University of Southern California*) et Xerox PARC dans le cadre du projet VINT (*Virtual Inter Network Testbed*). Ce projet est soutenu par le DARPA (*Defense Advanced Research Projects Agency*). NS-2 contient des bibliothèques pour la génération des fonctions (topologie, trafic, routage, MAC, LLC,...) et des outils graphiques pour l'interprétation (*Xgraph*) et la visualisation (*network animator NAM*) des résultats. Il est principalement bâti avec les idées de la conception objet, de réutilisation du code et de modularité.

Le simulateur NS-2 contient les fonctionnalités nécessaires pour l'étude des méthodes d'accès au médium, des algorithmes de routage point à point ou multipoint, des protocoles de transport, de session, de réservation de ressources, des protocoles d'application. L'ensemble de ces capacités utilisées conjointement ont permis l'étude des différents mécanismes au niveau de différentes couches de l'architecture réseau.

Application	Web, ftp, Telnet, générateur de trafic (CBR, ...).
Transport	TCP, UDP, RTP, SRM.
Réseau	Routage Statique et dynamique unicast et multicast (vecteur de distance, DSR, AODV)
Gestion de file d'attente	RED, DropTail, Token bucket.
Discipline de service	CBQ, SFQ, DRR, Fair queueing
Système de transmission	CSMA/CD, CSMA/CA, lien point à point ;

**Tableau 4** : Les principaux composants de NS-2

### 3.2. Architecture et Implémentation de NS-2 :

L'architecture réseau de NS-2 est fortement basée sur le modèle des couches OSI.

Le simulateur NS-2 permet de simuler différents types de réseau, incluant les réseaux filaires et sans fil. Il est écrit en deux langages C++ et OTcl. Les composantes en C++ sont utilisées pour faire fonctionner le corps du simulateur. OTcl est utilisé comme interface et interprète pour les scripts de simulation, la configuration des nœuds et pour faire la liaison avec les classes objets de C++ de *ns-2*. OTcl est une extension orientée objet de Tcl (*Tool command language*) et permet une facile intégration avec d'autres langages.

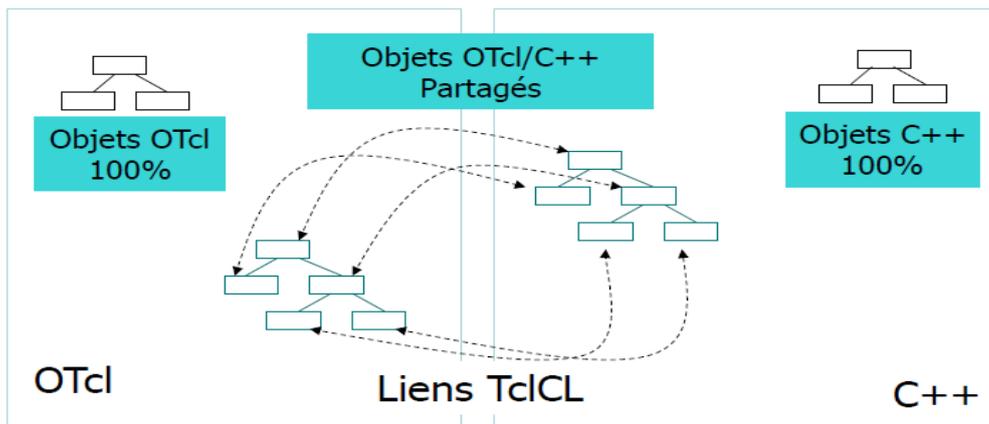


Figure 12 : Dualité des classes OTCL et C++

#### 3.2.1. Intégration d'olsr dans NS-2 :

Par défaut, le simulateur intègre plusieurs protocoles Ad-Hoc tels que DSR et AODV mais pas le protocole OLSR. Mais il existe des extensions pour *ns-2* afin d'implémenter le protocole OLSR. Notre choix a été *um-olsr*. Ce package permet d'implémenter OLSR (RFC-3626) dans le simulateur *ns-2*. Ceci explique notre choix de la version de *ns-2* à utiliser comme plateforme de simulation. Le développement de notre version du protocole OLSR pour *ns-2* a été basé sur le package *um-olsr*. Nous avons par la suite intégré le nouveau protocole OLSR dans *ns-2* sous une machine Linux (Kubuntu Version11.10) avec les commandes ci-dessous :

On copie notre paquet «*um-olsr-0.8.8.tgz*» dans le dossier «*ns-allinone-2.35/ns-2.35*» et ouvre un terminal en root puis taper ces commandes :

```
$ cd ns-allinone-2.35/ns-2.35/
$ tar zxvf um-olsr-0.8.8.tgz
$ ln -s ./um-olsr-0.8.8 ./olsr
$ patch -p1 < olsr/um-olsr_ns-2.35_v0.8.8.patch
$ make
```

Et finalement, la validation de notre implémentation a été faite grâce à des modèles de topologie bien déterminés.

### 3.2.2. Composition d'un modèle en NS-2 : [15]

Un modèle de réseau en ns est constitué:

- ✚ Nœuds : endroits où est généré le trafic, ou nœuds de routage;
- ✚ Liens de communication entre les nœuds du réseau.
- ✚ Agents de communication, représentant les protocoles de niveau transport (*tcp*, *udp*); ces agents sont attachés aux nœuds et connectés l'un à l'autre, pour échange de données.
- ✚ Applications qui génèrent le trafic de données selon certaines lois, et sont liées aux agents.

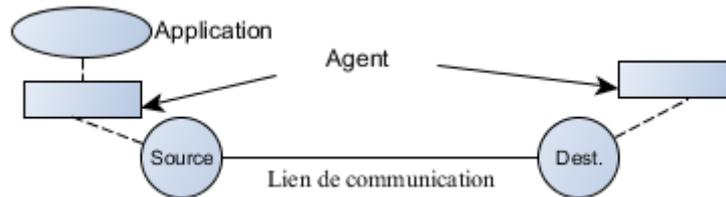


Figure 13: Composition d'un modèle de réseaux en NS

### 3.2.3. Les modèles de mobilité sous NS-2 :

L'évaluation d'un protocole de routage dépend soigneusement du choix d'un modèle de mobilité pour illustrer les mouvements réalistes des nœuds mobiles.

#### ✚ Random Waypoint Model (RWM) :

Dans ce modèle la mobilité des nœuds est typiquement aléatoire. En effet, la destination et la vitesse de chaque nœud mobile, désirant se déplacer, est aléatoire, et est limité à un intervalle bien déterminé. Après son déplacement le nœud mobile s'immobilise pour un temps fini, puis se déplace à nouveau de la même manière que la première fois.

#### ✚ Random Direction Model (RDM) et Modified Random Direction Model (MRDM):

De la même façon que le modèle précédant, mais dans ces modèles, le nœud en déplacement doit atteindre ou de ne pas forcément atteindre les bornes de la surface de simulation, puis s'immobilise. Notons que le modèle RWM reflète bien les caractéristiques des réseaux mobiles Ad hoc (*MANET*), car il offre une mobilité aléatoire aux nœuds mobiles appartenant au réseau. Contrairement aux modèles RDM et MRDM qui, d'une manière indirecte, conditionne le mouvement des nœuds. Cette constatation nous a poussés à choisir ce modèle (RWM) pour notre simulation.

### 3.2.4. Les Nœuds Mobiles sous NS-2 :

Lorsqu'on crée un nœud mobile dans une simulation, on lui associe un objet *MobileNode*, un agent de routage et la pile réseau. Ensuite ces composants sont interconnectés et la pile est connectée au canal.

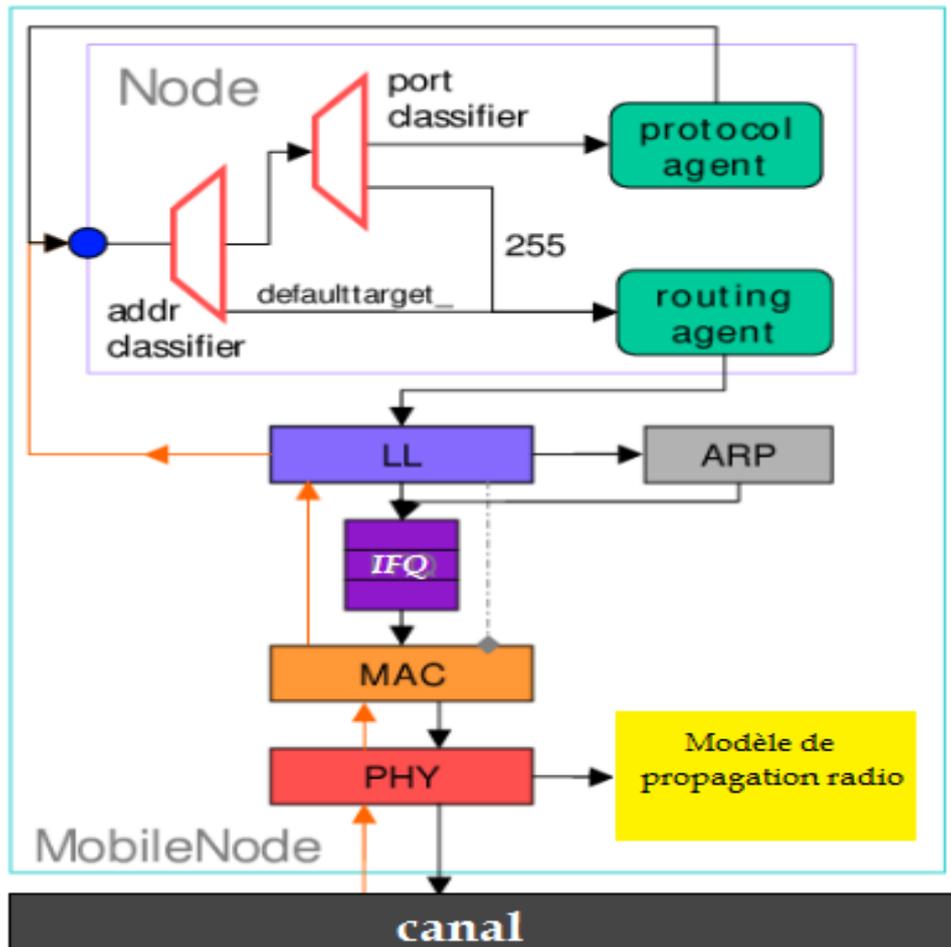


Figure 14 : composants réseaux d'un nœud mobile

- ✦ Un **classifieur de l'adresse** pour donner des paquets au classifieur de port ou à un agent de routage. La cible par défaut du classifieur de l'adresse est souvent l'agent de routage, tenir en compte d'expédition du paquet.
- ✦ Un **classifieur de port**, utilisé pour donner des paquets aux agents attachés au nœud mobile.
- ✦ Un **agent de routage** pour la gestion de la table et l'expédition des paquets. L'agent de routage devrait mettre le champ next-hop du paquet pour indiquer leur next-hop destination.
- ✦ Une **couche de liaison (LL)** pour convertir les adresses réseau aux adresses matérielles (avec l'aide du module ARP) et préparer des paquets pour être mis sur un canal sans fil.
- ✦ Un module **ARP** qui résout les adresses réseau aux adresses matérielles (MAC).
- ✦ Une file d'interface (**IFQ**), pour stocker les paquets qui devraient être envoyés à l'extérieur.
- ✦ Une **couche MAC** pour la gestion d'accès au canal sans fil.
- ✦ Une **interface réseau** qui envoie et reçoit des paquets sur le canal sans fil.
- ✦ Un **modèle de la propagation radio** qui détermine la fréquence du signal reçu, et d'où, si un paquet peut être reçu par une interface réseau ou pas.
- ✦ Un **canal** sans fil (Wireless Channel) sur lequel les paquets sont distribués [40]

#### 4. La simulation par NS-2 :

Chaque simulation dans ns-2 est exécutée à partir d'un script écrit en Tcl avec la commande : **ns « fichier.tcl »**. Dans ce script, on spécifie les paramètres de simulation (temps de simulation, modèle de propagation radio, type de canal sans fil, modèle de mobilité, nombre de nœud, type de protocole, bande passante, type de trafic, etc.). Les résultats des simulations sont fournis dans ns-2 sous forme de fichiers traces. Chaque événement survenu durant la simulation est inscrit dans une ligne du fichier texte.

L'interface NAM (Network Animator) permet d'avoir une visualisation graphique de la simulation. Cette extension de ns-2 est idéale pour avoir une idée visuelle sur le type de la topologie, la densité, la mobilité et les données échangées.

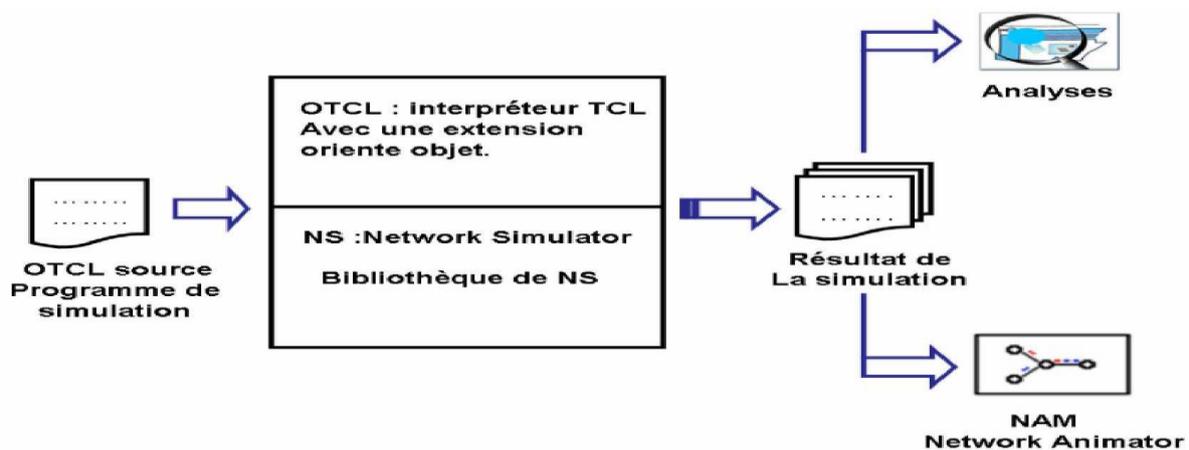


Figure 15 : les étapes de nos simulations avec NS

##### 4.1. Les différentes phases de simulation

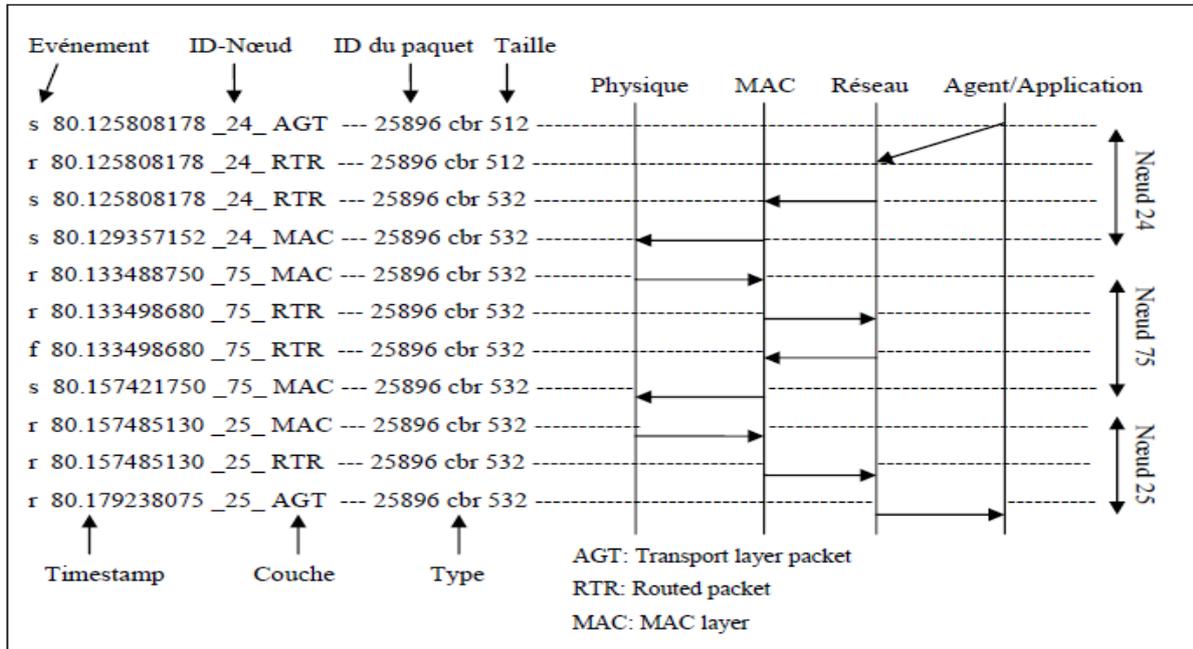
- Créer les nœuds du réseau (nœud d'extrémité et de transit).
- Définir des liens entre ces nœuds (débit, délai, type file ...)
- Définir le routage.
- Créer des agents de transport et les lier aux nœuds (*tcp, udp ...*)
- Créer des générateurs de trafic (*applications*) et les lier aux agents de transport.
- Lancer la simulation et créer des fichiers de trace pour effectuer des mesures
- Visualiser le réseau et tracer des graphes.

Le résultat de simulation donne deux types de fichiers : fichier.**tr** et fichier.**nam**

##### 4.2. Le format de traces de simulations :

Les résultats de simulation sont fournis sous forme de fichiers trace et animation en mode texte où chaque ligne décrit un événement qui s'est produit lors de la simulation (opérations entre les couches). Le format de fichier trace a un effet direct sur la rapidité des simulations et

en particulier sur l'utilisation de CPU et la mémoire de la machine de simulation. En effet, dans le cas de réseaux denses et pour des simulations de 100 nœuds et plus, les fichiers traces de chaque simulation peut dépasser un gigabit de taille.



**Figure 16:** Extrait et détails importants dans les traces des simulations sous ns-2

Les événements les plus importants dans les fichiers de traces sont la transmission (s), la réception (r), renvoi (f) et le rejet(d) d'un paquet. Chaque paquet généré durant une simulation, a un identifiant unique et permet de suivre son état dans le réseau et de calculer par la suite les performances de chaque protocole.

Pour être exploitables, ces informations doivent être traitées par un filtre qui analyse chaque ligne du fichier pour donner des informations interprétables. On a choisi l'outil de filtrage AWK [38]. Les graphes sont obtenus en utilisant l'Excel ou l'outil gnuplot [39] pour produire des courbes concernant les différentes métriques du réseau.

### 4.3. Statistiques et visualisation :

Pour visualiser, animer et interpréter les données fournies à travers les fichiers trace et donner un compte-rendu graphique nous utiliserons l'outil d'animation NAM (Network AniMator), basé sur le langage TCL/TK, et le langage AWK.

**NAM (Network AniMator) :** permet d'avoir une visualisation graphique du réseau décrit dans le fichier de simulation TCL tout en animant les liens entre les nœuds du réseau, le type de la topologie, la densité, la mobilité et les données échangées. Les scénarios simulés génèrent des fichiers avec l'extension .nam, la visualisation de ces scénarios avec NAM se fait à l'aide de la commande : **nam « fichier.nam »**

**Le langage AWK :** AWK est un outil qui permet d'effectuer des recherches, simples ou complexes, dans des fichiers textes. Il tire son nom des trois programmeurs qui l'ont développé : *Alfred V. Aho, Peter J. Weinberger et Brian W. Kernighan* [38].

L'exécution d'un programme AWK s'effectue par la commande :

```
awk -f «fichier.awk» «fichier.tr».
```

La commande exécute séquentiellement le programme se trouvant dans ce fichier programme sur chaque ligne des fichiers en entrée.

- ✿ « -f » donne accès aux utilitaires awk pour exécuter les instructions du « fichier.awk » afin de traiter les données du « fichier.tr »
- ✿ « fichier.awk » est le fichier d'instructions awk enregistré sous l'extension .awk.

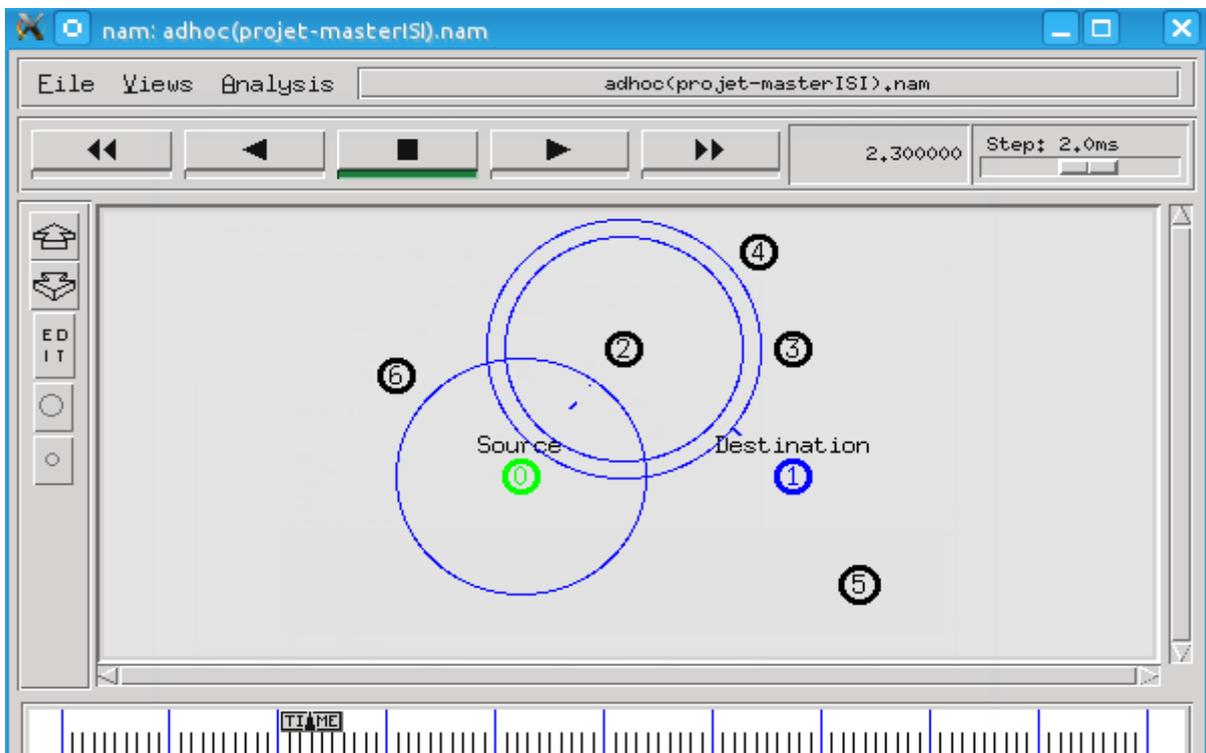


Figure 17 : Animation d'une simulation de sept(7) nœuds dans NAM

#### 4.4. Modèle de simulation :

Pour simuler le type de communication dans *ns-2*, on utilise des applications de trafic à débit constant CBR (*Constant Bit Rate*) et qui modélisent la couche application sur des agents de transport UDP (*User Datagrammes Protocol*). On génère les scénarios de trafic aléatoires grâce à notre générateur de trafic *cbrgen.tcl* en TCL sous *ns-2*. Chaque nœud source émet, à un instant donné, choisi aléatoirement entre 0 seconds et 100 seconds, des paquets de taille 512 octets et se déplace dans une surface carrée de dimension 500m x 500m.

Les tableaux 5 et 6 présentent le sommaire de nos simulations en mode dynamique. Il faut noter qu'on a effectué 114 simulations par protocole dans l'étude comparative des quatre protocoles. Ses paramètres de simulation sont utilisés dans nos scripts de simulation écrits en TCL. Nous avons également développé un script en Bash afin d'automatiser toutes nos simulations et calculs des valeurs.

Modèle de simulation	Nbre de réseau	Nbre de nœuds mobiles par réseau	Vitesse max	Temps de pause	Trafic CBR par réseau (avec udp)	Scénarios de mobilité
Étude de la Densité	5	10-20 -30 - 40-50	20 m/s	2 secondes	8	6 scénarios
Étude de la Mobilité	4	50	5-10-15-20 m/s	2 secondes	25	6 scénarios
	5	50	20 m/s	0-25-50-75-100 secondes	25	6 scénarios
Étude de la Charge	5	50	20 m/s	2 s	5-10-15-20-25 connexions	6 scénarios
Nombre total de simulations				$5*6*4+4*6*4+5*6*4+5*6*4 = 456$ simulations		

**Tableau 5:** Sommaire des paramètres de simulation

Chaque réseau créé est simulé avec six scénarios de mobilité que nous moyennons les résultats pour obtenir un mouvement général.

Dans les scénarios que nous avons réalisés, un ensemble de paramètres communs aux scénarios est fixé. Ces paramètres sont récapitulés dans le tableau suivant :

Paramètre	Valeur
Modèle de propagation radio	Two Ray Ground
Couche MAC	IEEE 802.11, DCF
Protocole de routage	AODV, DSR, DSDV et OLSR
Couche transport	UDP
Modèle de Mobilité	Random Waypoint Model
Vitesses de déplacement	5-10-15-20 m/s
Trafic	CBR ( <i>Constant Bit Rate</i> )
Taille d'un paquet	512 octets
Topologie	500 x 500 m et 1000 x 1000 m
Portée de Transmission	250 m
Nombre de nœuds	10 à 100
Temps de Simulation	60 et 100 secondes

**Tableau 6:** Paramètres communs pour la simulation

Chaque simulation accepte en entrée deux fichiers (mobilité et trafic) décrivant le scénario de la simulation : ils définissent le mouvement exact de chaque nœud et l'ordre exact des paquets lancés par chaque nœud.

#### 4.4.1. Modèle de mobilité :

La création des mouvements des nœuds pour chaque scénario est faite par notre commande de générateur de mouvements des nœuds qui génère un fichier décrivant la mobilité des nœuds

utilisant l'algorithme «Random Waypoint» pour chaque réseau. On exécute notre commande sous le répertoire `ns-allinone-2.35\ns-2.35\indep-utils\cmu-scen-gen\setdest`, comme suit :

```
./setdest [-n nbre_nœud] [-p pausetime] [-M maxspeed] [-t simtime] [-x maxx] [-y maxy] > [file name]
```

**Exemple:** `./setdest -n 10 -p 1 -M 20 -t 100 -x 500 -y 500 > set-10-1`

Cet exemple génère un modèle de mobilité de 10 nœuds qui se déplacent avec une vitesse maximale de 20 m/s, un temps de pause de 1s entre chaque mouvement, et dans une zone carrée limitée de 500 x 500 m. Le réseau se trouve alors mieux contrôlé.

#### 4.4.2. Modèle de trafic :

Afin de créer un trafic dans le réseau, certains paramètres doivent être spécifiés : le nombre de nœuds, le nombre maximal de connexions ainsi que la fréquence d'envoi. On crée un générateur de trafic permettant de créer le trafic CBR (*Constant Bit Rate*).

Ce générateur s'exécute sous `ns-allinone-2.35\ns-2.35\indep-utils\cmu-scen-gen`, comme suit :

```
ns cbrgen.tcl [-type cbr|tcp] [-nn nodes] [-seed seed] [-mc connections][[-rate rate] > [file name]
```

**Exemple:** `ns cbrgen.tcl -type cbr -nn 10 -seed 1 -mc 5 -rate 0.2 > cbr-10-5`

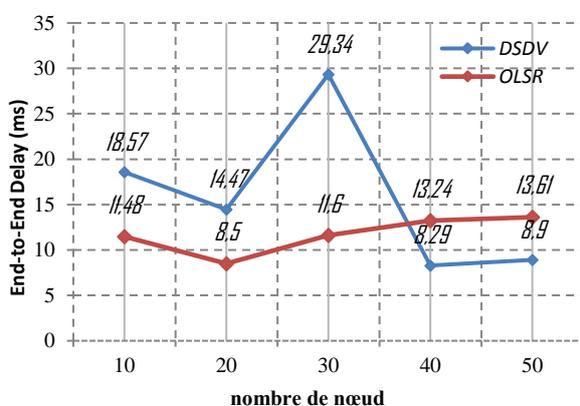
Cet exemple génère un modèle qui permet de créer aléatoirement des connexions de type CBR qui commencent à des instants distribués uniformément entre 0 et 100 secondes avec une fréquence d'envoi de 1 paquet par 5 seconds. La taille des paquets de données est 512 octets, et le réseau possède 10 nœuds et 5 connexions.

## 5. Interprétation des résultats :

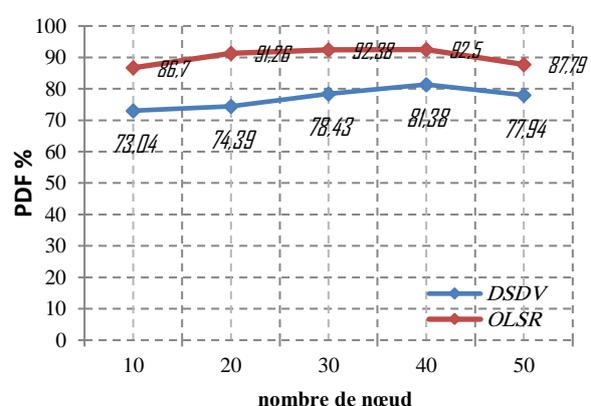
### 5.1. Routage proactif :

Nous comparons le comportement des deux protocoles de routage proactif OLSR et DSDV.

#### 5.1.1. Résultats de l'étude de l'effet du nombre de nœuds mobiles dans le réseau:



**Figure 18:** End to end Delay en fonction du nombre de nœuds



**Figure 19:** PDF en fonction du nombre de nœuds

À la première vue on remarque que DSDV est en retard en temps de réponse par rapport à OLSR jusqu'à un certain nombre de nœud élevé (40 nœuds), cela est dû à l'utilisation des

relais multipoints(MPRs) par OLSR pour minimiser la surcharge du réseau. À partir de 50 nœuds le protocole OLSR prend un léger retard sur DSDV.

Si le protocole OLSR utilise les MPRs pour acheminer les paquets à destination il est tout à fait évident qu'il livre un taux élevé de paquets de données par rapport à DSDV qui prend un peu de temps à découvrir les chemins. Ce schéma nous donne cette variation de taux avec un avantage pour OLSR.

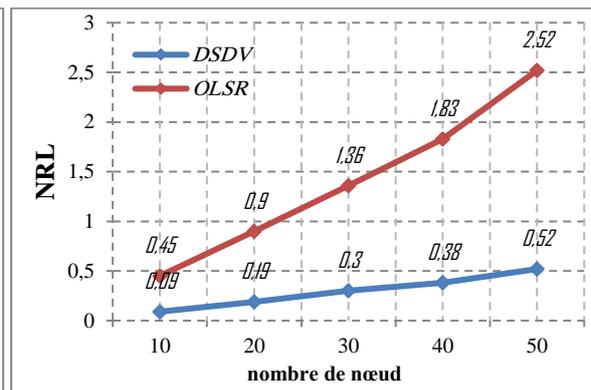
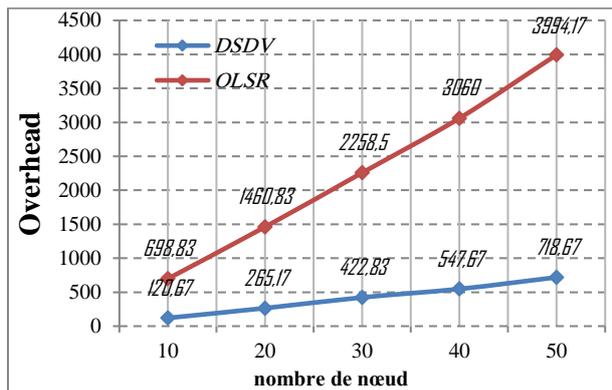


Figure 20 : Overhead en fonction du nombre de nœuds Figure 21: NRL en fonction du nombre de nœuds

Le nombre de paquet de contrôle (Overhead) tend par nature à augmenter lorsque le nombre de nœuds augmente (Figure 5). Il augmente cependant pour les deux protocoles de routage, il est moins élevé pour DSDV et très élevé pour OLSR.

La charge des paquets de contrôle (NRL) représente le rapport entre le nombre des paquets de contrôle envoyés dans le réseau, par rapport au nombre des paquets de données reçues par les nœuds destination. Cet indicateur reflète l'efficacité des protocoles de routage en termes de paquets de contrôle générés. Donc plus le nombre de paquets de contrôle générées par un protocole est élevé plus son NRL augmente.

### 5.1.2. Résultats de l'étude de l'effet de la mobilité (vitesse) des nœuds dans le réseau:

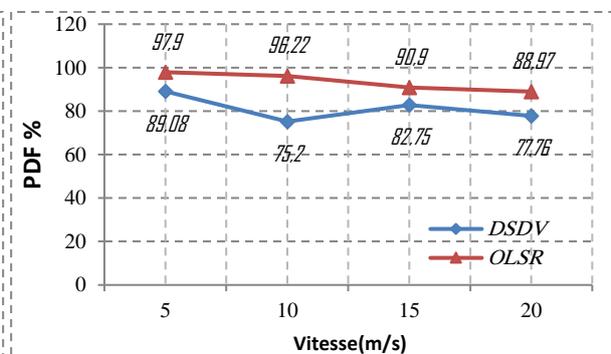
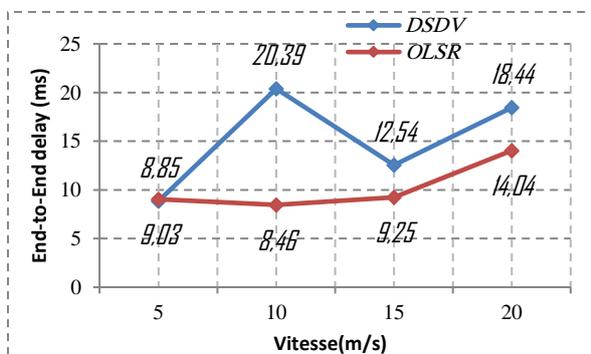


Figure 22: End to End Delay en fonction de la vitesse Figure 23: PDF en fonction de la vitesse

Sur la figure 22, à partir de 5m/s de vitesse OLSR donne un délai de 9.03 millisecondes pendant que DSDV donne 8.85. Ici on remarque qu'OLSR est toujours bien adapté au réseau

à forte mobilité par rapport à DSDV même si le délai s'accroît avec l'augmentation de la vitesse avec toujours un meilleur délai pour OLSR.

La figure 23 montre que le taux de paquets délivrés est influencé par l'augmentation de la vitesse des nœuds et tend à diminuer. Il reste cependant plus élevé pour OLSR que pour DSDV ce qui dû aux meilleurs performances de OLSR en terme de délai.

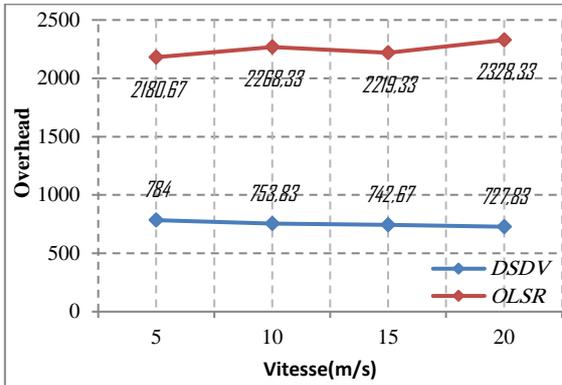


Figure 24: Overhead en fonction de la vitesse

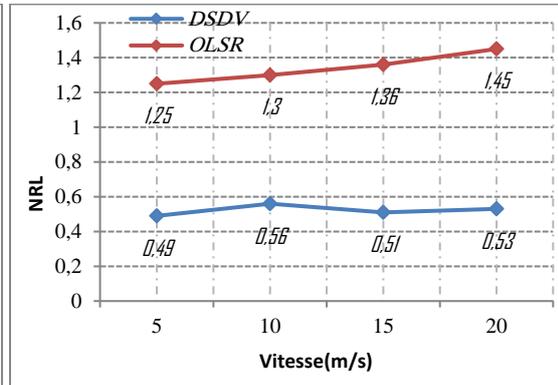


Figure 25: NRL en fonction de la vitesse

Les deux figures montrent qu'en termes de nombre de paquets de contrôle (Overhead) et de la charge des paquets de contrôle (NRL), OLSR a le plus nombre élevé par rapport à DSDV. Sauf pour celui du DSDV le nombre de paquets de contrôle tend à diminuer lorsque la vitesse augmente ce qui donne une variation topologique.

**5.1.3. Résultats de l'étude de l'effet de la mobilité (pause time) des nœuds dans le réseau:**

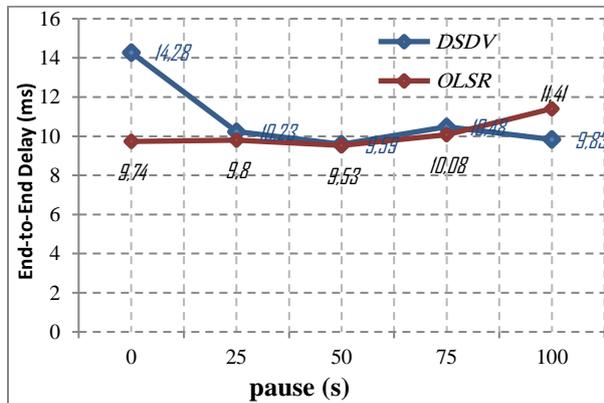


Figure 26: End to End Delay en fonction du temps de pause

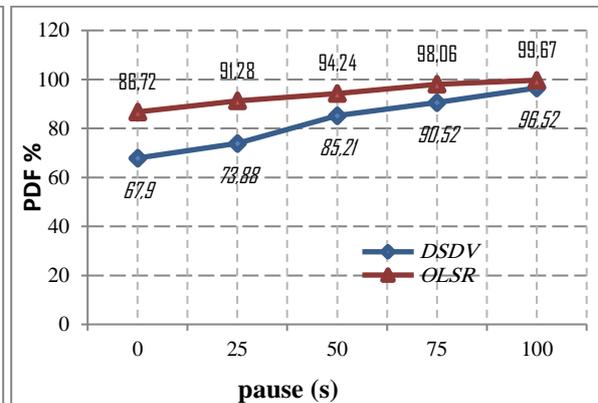


Figure 27: PDF en fonction du temps de pause

Sur la figure 26, si le temps de pause est égal à 0 (mobilité forte) ce qui implique un changement rapide de la topologie du réseau, le délai de DSDV (avec une diminution continue du délai moyen de transfert) est légèrement élevé par rapport à OLSR. On constate que le meilleur protocole en délai moyen de transfert est OLSR jusqu'au temps de pause 75.

Dans la figure 27, si la mobilité est plus ou moins forte, le taux de paquets délivrés par DSDV est inférieur à celui d'OLSR. Contrairement à DSDV, le protocole OLSR parvient à faire des

prises à jour pour les routes les plus récentes plus rapidement avec l'émission des paquets de contrôle tels TC (*topology control*), HELLO (*détection du voisinage*) d'où la livraison d'un meilleur taux de paquets de données par rapport à DSDV.

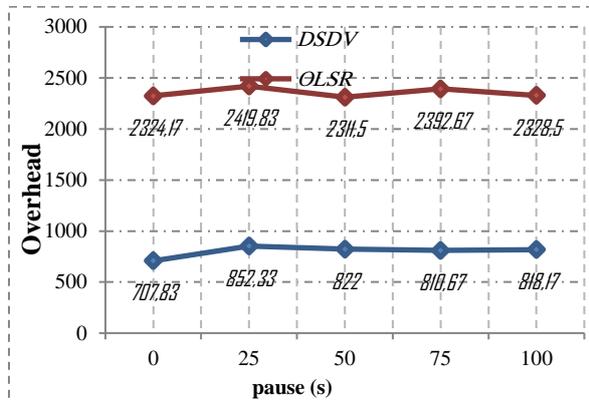


Figure 28: Overhead en fonction du temps de pause

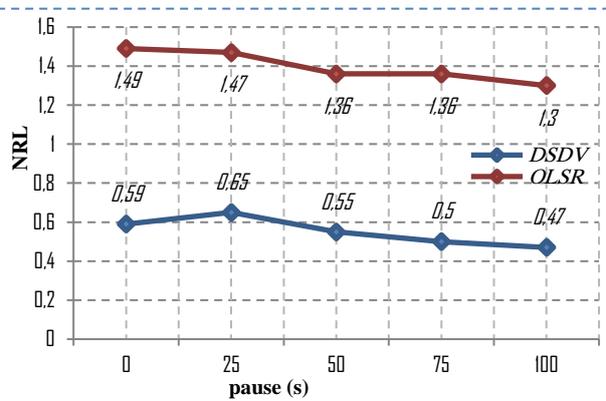


Figure 29: NRL en fonction du temps de pause

Sur la figure 28 le nombre de paquets de contrôle est plus élevé pour OLSR que pour DSDV et il augmente plus ou moins au fur et à mesure que le temps de pause est augmenté. Ce nombre élevé des paquets de contrôle s'explique par un changement topologique lors des déplacements des nœuds mobiles et les ruptures de liens entre ces nœuds du réseau.

Sur la figure 29, on remarque que la charge des paquets de contrôle (NRL) décroît avec la variation du temps de pause. Cela s'explique par la transmission d'un important taux de paquets de données avant la reprise des déplacements des nœuds mobiles dans le réseau.

**5.1.4. Résultats de l'étude de l'effet du trafic dans le réseau:**

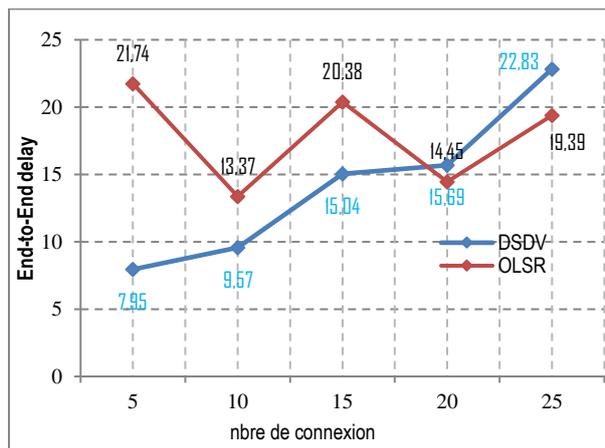


Figure 30: End to End Delay en fonction du Trafic

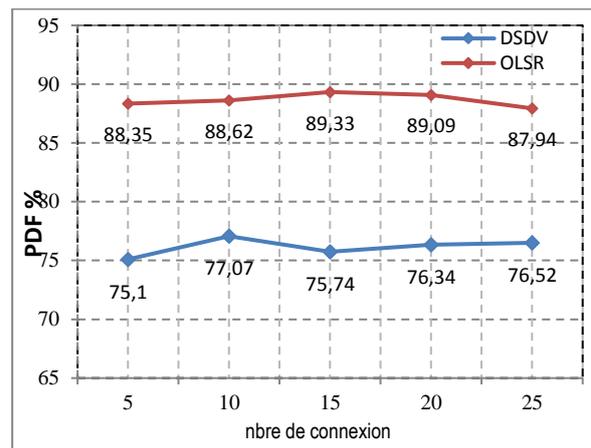


Figure 31: PDF en fonction du Trafic

Dans la figure 30, lorsqu'on varie le nombre de connexions (charge de trafic) entre 5, 10, 15, 20 et 25 connexions, le nombre de nœuds est fixé à 50 et le temps de pause des nœuds est fixé à 2s, nous remarquons, dans la figure que le délai est moins élevé pour DSDV (*mais il*

s'accroît avec l'augmentation de la charge de trafic) que pour OLSR. Par contre il est moins élevé pour OLSR que pour DSDV à partir de 20 et 25 connexions.

La figure 31 montre que le taux de paquets délivrés est influencé par l'augmentation de la charge de trafic. Le taux le plus élevé est celui produit par OLSR et le plus faible est celui de DSDV.

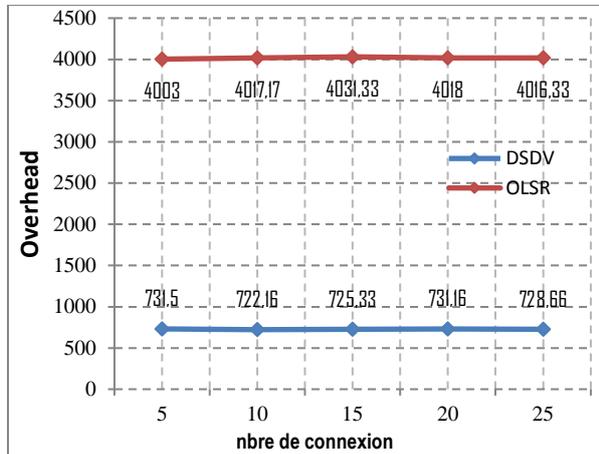


Figure 32: Overhead en fonction du Trafic

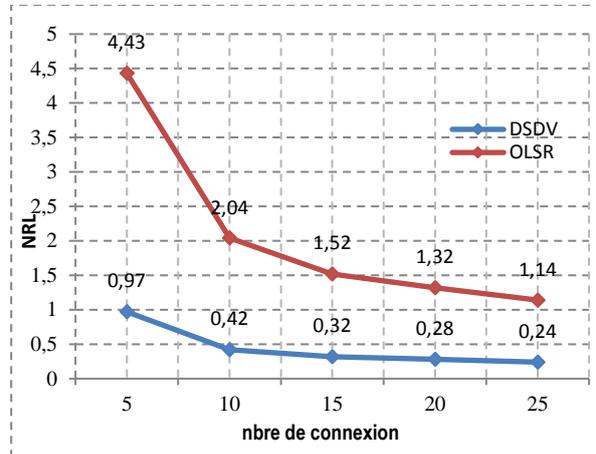


Figure 33: NRL en fonction du Trafic

Dans la figure 32, nous remarquons que le nombre de paquets de contrôle est élevé pour OLSR que pour DSDV et il s'accroît plus ou moins avec l'augmentation de la charge de trafic durant toute la simulation. Dans la figure 33, la charge des paquets de contrôle (NRL) tend à diminuer lorsque le nombre de connexions augmente. Elle reste cependant plus au moins stable, elle est moins importante pour DSDV et très importante pour OLSR.

### 5.1.5. Synthèse :

Dans cette section on a simulé nos deux protocoles de routage proactif (DSDV et OLSR), dans le but de faire une étude des performances de ces derniers.

Nous avons jugé utile de faire un aperçu sur les paramètres (métriques) utilisés dans la littérature du réseau mobile ad hoc. Nous en avons extrait les mieux appropriés pour mesurer les performances en terme du taux de livraison des paquets avec succès, délai moyen de bout en bout, overhead et pour décider du meilleur des protocoles sous des conditions particulières. Le travail que nous avons effectué (simulation sous NS-2), nous a permis de voir l'impact de la mobilité, du nombre de nœuds (ou densité), sur ces métriques pour les protocoles (DSDV et OLSR) et nous permet de dire que OLSR performe bien par rapport à DSDV dans la plupart des résultats obtenus. Enfin, il nous a permis de savoir que le choix de l'algorithme de routage dépend de plusieurs contraintes et qu'il est intéressant de considérer et de combiner le maximum d'entre elles pour tirer les meilleurs profits.

## 5.2. Routage réactif :

Cette partie est basée sur l'analyse comparative des comportements des deux protocoles de routage réactif AODV et DSR.

### 5.2.1. Résultats de l'étude de l'effet du nombre de nœuds mobiles dans le réseau:

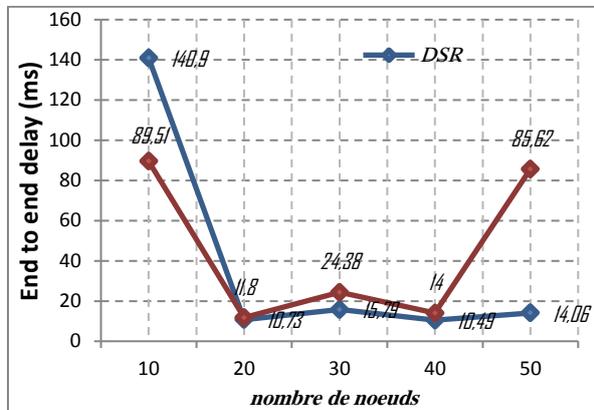


Figure 34: End to end Delay en fonction du nombre de nœuds

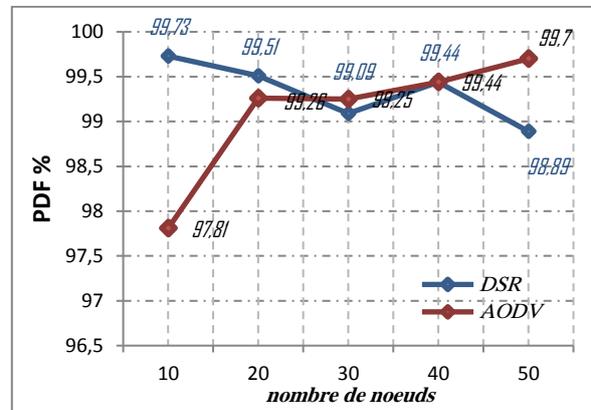


Figure 35: PDF en fonction du nombre de nœuds

Un premier aperçu sur la figure 34 donne une idée sur le temps moyen d'arrivée des paquets. Chacun des deux protocoles réactifs DSR et AODV sont plus lents en temps d'arrivée pour un nombre de nœuds égal à 10, avec respectivement un temps moyen d'arrivée de 140,9 ms et de 89,51 ms. Et à partir du nombre de nœuds allant de 20 jusqu'à 40 le temps moyen d'arrivée des paquets pour chacun des deux protocoles DSR et AODV diminue énormément. En prenant un avantage sur AODV, DSR voit une quasi-monotonie durant tout le reste de la simulation et seul AODV voit un énorme changement passant de 14 ms pour 40 nœuds à 85,65 ms pour 50 nœuds.

Sur la figure 35, pour un nombre de nœuds égal à 10, le protocole DSR donne un taux de délivrance de 99,73% supérieur à celui d'AODV avec 97,81%. On remarque que chacun des deux protocoles réactifs AODV et DSR sont meilleurs avec plus de 99% relativement au taux de délivrance concernant un nombre de nœuds allant de 20 jusqu'à 30 et convergent à partir de 40 nœuds avec un taux de 99,44%, et enfin avec une petite régression pour DSR (98,89%) pour 50 nœuds et une augmentation pour AODV jusqu'à 99,70%.

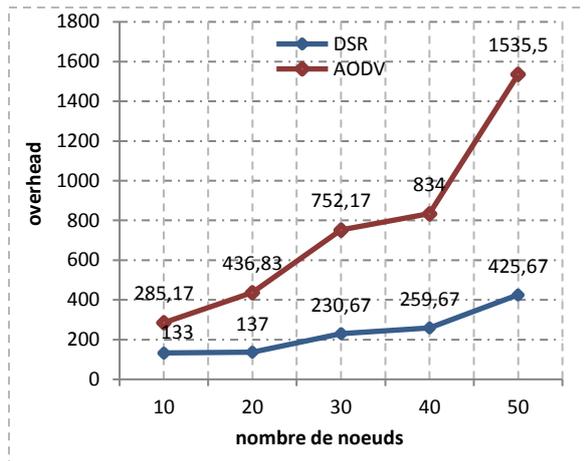


Figure 36: Overhead en fonction du nombre de nœuds

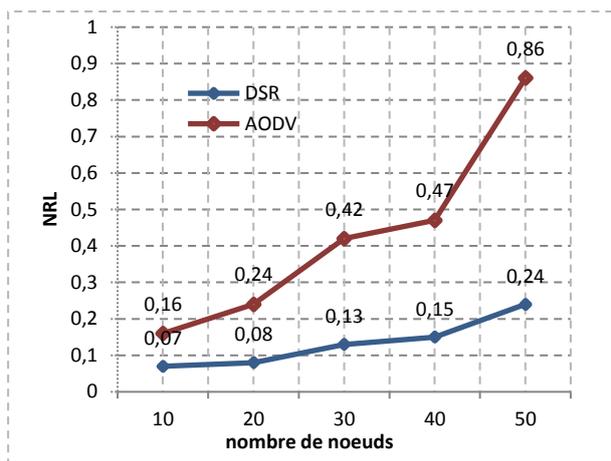


Figure 37: NRL en fonction du nombre de nœuds

Dans la figure 36 la première remarque est le nombre élevé de paquets de contrôle émis par AODV et qui augmente à chaque simulation (285.17; 436.83; 752.17; 834; 1535.5 respectivement pour 10, 20, 30, 40, 50 nœuds). On peut même dire que le nombre est presque doublé entre deux simulations successives.

Ce n'est pas le même cas pour DSR mais avec un nombre plus petit de paquets de contrôle (133; 137; 230.67 ; 259.67; 425.67) qui augmente proportionnellement avec le nombre de nœuds. Ces variations sont causées par la forte mobilité des nœuds du réseau.

La métrique NRL est fonction du nombre de paquets de contrôle et définit le coût de routage. Nous remarquons, dans la figure 37 que le coût de routage est plus élevé pour AODV que pour DSR et il s'accroît avec l'augmentation du nombre de nœuds.

### 5.2.2. Résultats de l'étude de l'effet de la mobilité (vitesse) des nœuds dans le réseau:

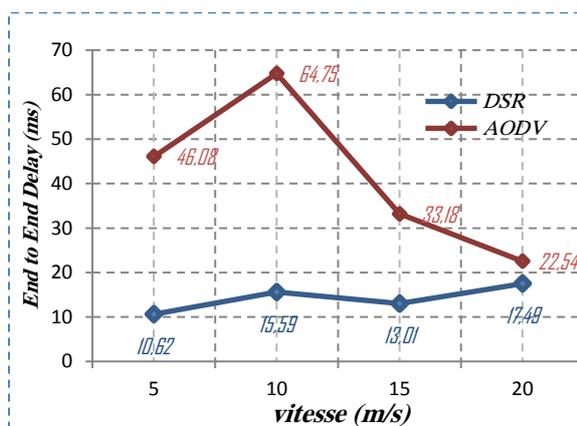


Figure 38: End to End Delay en fonction de la vitesse

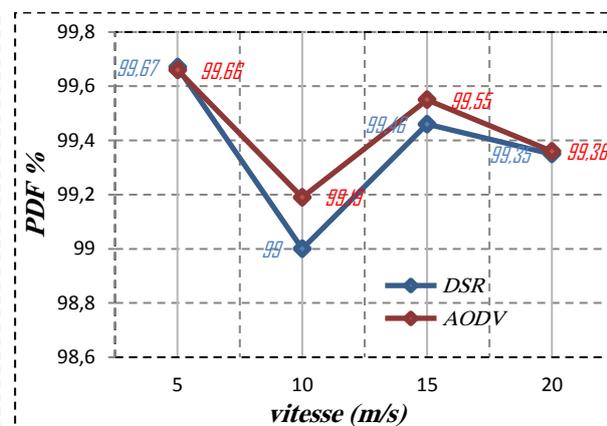


Figure 39: PDF en fonction de la vitesse

La figure 38 illustre le délai moyen de bout-en-bout par rapport à la vitesse des nœuds dans le réseau. Le délai de bout-en-bout d'un paquet dans le réseau est la somme des retards introduits par les nœuds intermédiaires entre la source et la destination. Il dépend du délai de

traitement dans un nœud intermédiaire, du délai de la mise en file d'attente, du délai lors de l'envoi sur le support physique et le délai de la propagation selon la distance.

Dans un environnement à haute mobilité, les liens entre les nœuds changent très rapidement et les connexions entre les nœuds existent pour des périodes de temps très courtes.

Par contre pour des réseaux avec faible mobilité ou mobilité nulle, les liens existent pour une période de temps très grande favorisant ainsi le trafic CBR à être transmis correctement à la destination. À une vitesse de mobilité de 5 ; 10 et 15 m/s, les deux protocoles présentent le même comportement. Par contre, à partir de 20 m/s, le protocole AODV (en diminuant de 33.18 à 22.54 millisecondes) prend l'avantage par rapport à DSR (en augmentant de 13.01 à 17.49 millisecondes).

Nous remarquons que les deux protocoles obtiennent le même taux 99.67% pour une vitesse maximale de 5 m/s sur la figure 39. Nous constatons aussi que le protocole AODV prend de l'avantage par rapport à DSR pour les réseaux à grandes mobilités à partir de 10 et de 15 m/s. Le nombre supérieur de messages de contrôle émis par AODV à cette vitesse de mobilité par rapport à DSR explique ce léger avantage. Les deux protocoles convergent vers 99.35% pour une vitesse de 20m/s.

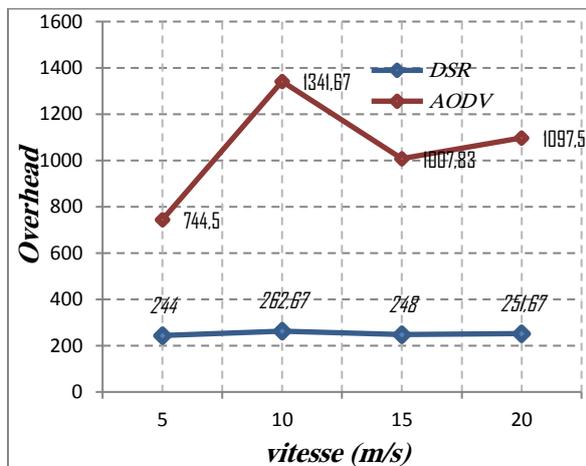


Figure 40: Overhead en fonction de la vitesse

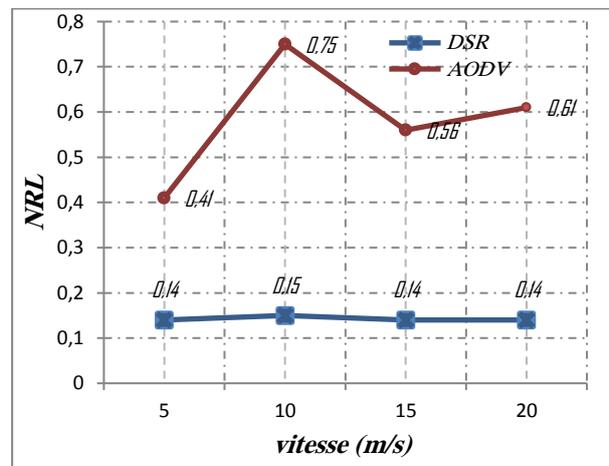


Figure 41: NRL en fonction de la vitesse

La figure 40 donne le nombre de messages de contrôle émis durant les simulations en fonction des vitesses de mobilité. On constate le nombre élevé de paquets de contrôle émis par AODV par rapport à DSR.

Sur la figure 41, on voit que le coût de routage est élevé pour AODV que pour DSR. Mais on constate une diminution du coût de routage de pour le protocole AODV pour une vitesse de 15m/s. Et enfin pour les valeurs du coût de routage est quasiment stables pour DSR.

5.2.3. Résultats de l'étude de l'effet de la mobilité (pause time) des nœuds dans le réseau:

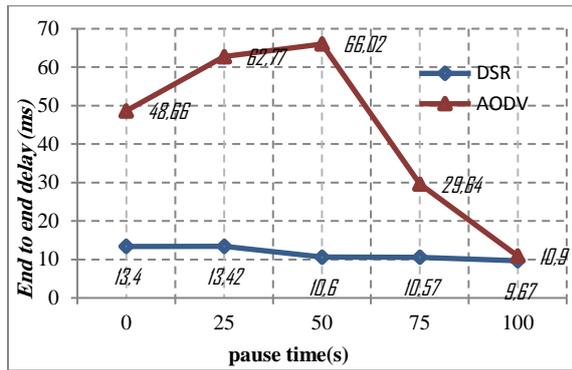


Figure 42: End to End Delay en fonction du temps de pause

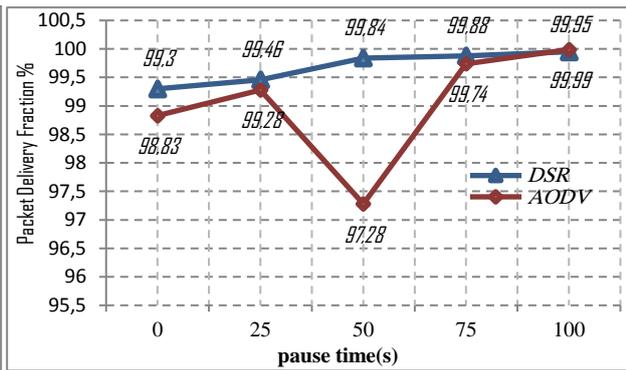


Figure 43: PDF en fonction du temps de pause

On remarque sur la figure 42 que lorsque le temps de pause est égal à 0 seconde (mobilité forte) ce qui implique un changement rapide de la topologie du réseau, le meilleur protocole en délai moyen de transfert est DSR avec une diminution continue du délai moyen de transfert durant toute la simulation. Alors qu'AODV voit une augmentation du délai moyen de transfert à partir de temps de pause allant de 0 ; 25 ; à 50 secondes et trouve une forte diminution du délai moyen de transfert au temps de pause égal à 75 et 100 secondes.

Dans la figure 43, au temps de pause 0, le protocole AODV avec un taux de 98.83% vient après DSR qui a un taux de 99.3%. Mais le protocole DSR offre un meilleur taux de délivrance proche de 100% durant toute la simulation alors que le protocole AODV donne une légère chute de 97.28% au temps de pause de 50 secondes (due aux temps de mise à jour table de routage), puis continue de tendre vers un taux proche de 100%.

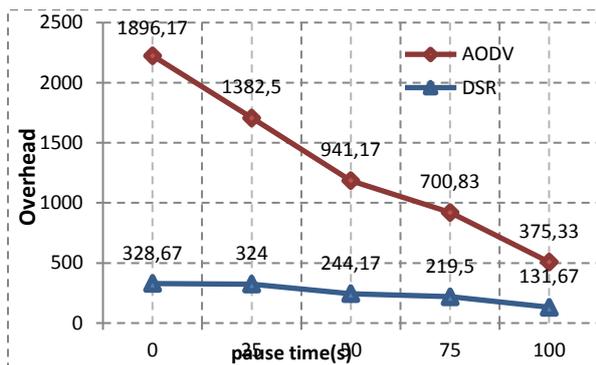


Figure 44: Overhead en fonction du temps de pause

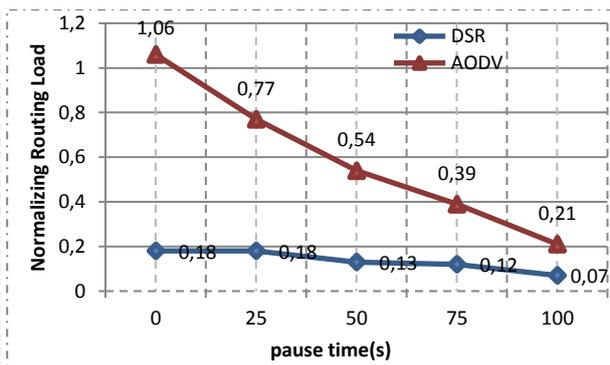


Figure 45: NRL en fonction du temps de pause

La figure 44 indique que les deux protocoles génèrent un grand nombre de paquets de contrôle (1896.17 pour AODV et 328.67 pour DSR) dès la première simulation avec un temps de pause 0 seconde (mobilité forte). On remarque aussi que le nombre de paquets de contrôle diminue entre deux simulations successivement de pause time de 0 ; 25 ; 50 ; 75 ; 100 secondes.

Le coût de routage est plus élevé pour AODV que pour DSR dans la figure 45 (respectivement 1.06 et 0.18) dès la première simulation avec un temps de pause 0 seconde (mobilité forte). Le coût de routage décroît avec l'augmentation de pause time.

**5.2.4. Résultats de l'étude de l'effet du trafic dans le réseau:**

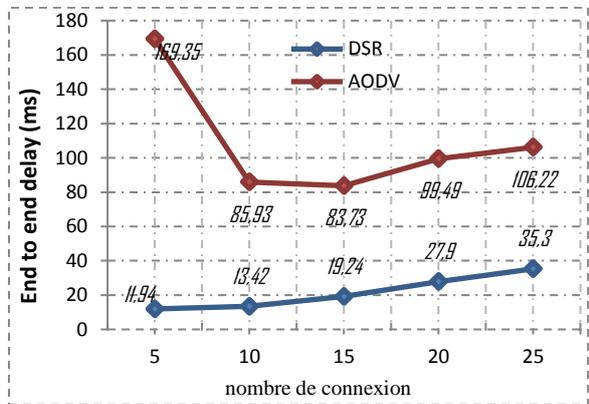


Figure 46: End to End Delay en fonction du Trafic

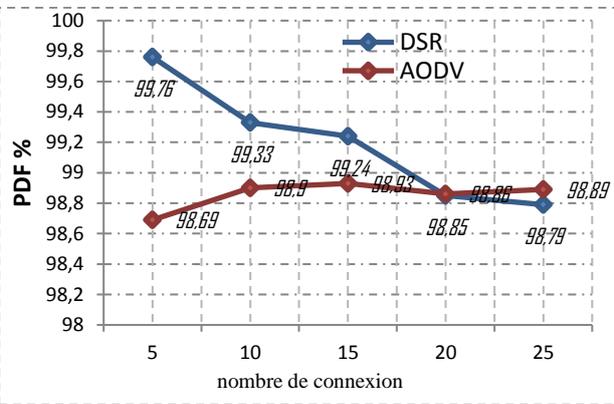


Figure 47: PDF en fonction du Trafic

La figure 46 montre qu'en termes de délai moyen de transfert, AODV est moins efficace par rapport à DSR. Sauf que pour DSR le délai moyen de transfert tend à augmenter lorsque la charge du trafic augmente. Par contre celui d'AODV continue de diminuer à partir de 10 à 15 connexions et voit une légère augmentation avec 20 à 25 connexions.

Dans la figure 47, le taux de paquets délivrés est influencé par l'augmentation de la charge de trafic. C'est ainsi que les valeurs pour DSR diminuent légèrement alors que celles pour AODV augmentent légèrement, avant de converger tous les deux vers un taux de plus de 98% pour un nombre de 20 connexions. Pendant que DSR continue de décroître avec 98.79%, AODV tente de croître légèrement avec 98.89% pour un nombre de 25 connexions

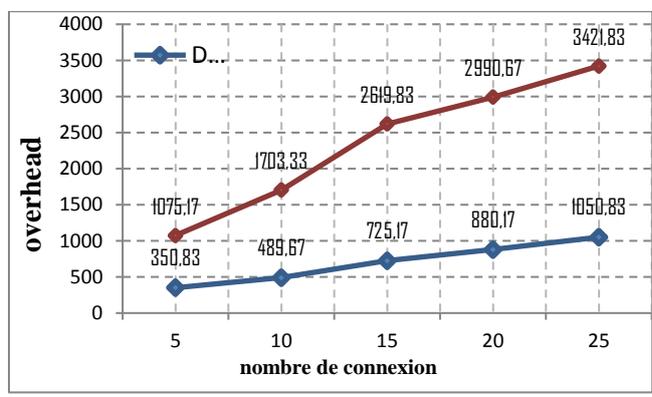


Figure 48: Overhead en fonction du Trafic

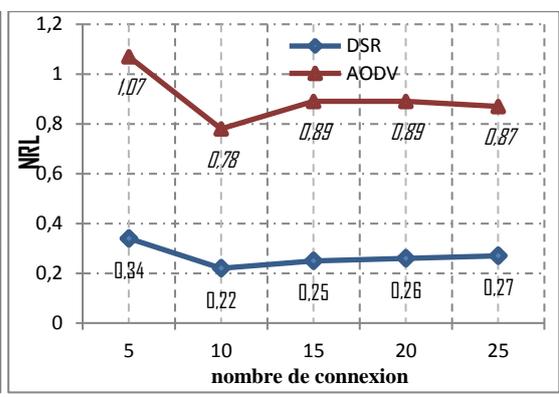


Figure 49: NRL en fonction du Trafic

Sur la figure 48, on constate un nombre élevé de paquets de contrôle émis par AODV et qui augmente à chaque simulation. C'est le même cas pour DSR mais avec un nombre plus petit

de paquets de contrôle c'est-à-dire à la simulation avec un nombre de *connexions* égal à 5, le nombre de paquets de contrôle de AODV est 1075.17 et la valeur pour DSR est 350.83 paquets puis pour les autres simulations de 10; 15; 20 jusqu'à 25 *connexions*, on remarque que le nombre de paquets de contrôle augmente proportionnellement avec le nombre de *connexions*.

La figure 49 montre que le coût de routage est élevé pour AODV que pour DSR. Mais on constate une diminution pour les deux protocoles pour un nombre de *connexions* égal à 10 (0.78 et 0.22 respectivement pour AODV et DSR). Et enfin pour les autres simulations de 15; 20 et 25 *connexions*, le coût de routage s'accroît avec l'augmentation de la charge de trafic.

### 5.2.5. Synthèse :

Dans cette partie on a simulé nos deux protocoles de routage réactif (AODV et DSR), dans le but de faire une étude comparative des performances.

Après analyse nous remarquons que DSR et AODV obtiennent de meilleures performances tous les deux dans les hypothèses initiales (forte mobilité, overhead, ...) avec un léger avantage pour DSR en termes de taux de paquets de données délivrés. Cela est expliqué par leurs caractéristiques réactives. Cependant, le passage à l'échelle est toujours un très gros problème pour DSR, ce dernier utilise le cache pour la découverte de route, ce mécanisme dégrade les performances de DSR en particulier en termes de taux de paquets de données délivrés dans le réseau. Par contre AODV s'adapte mieux au passage à l'échelle (densité) par rapport à DSR. Le choix du protocole de routage AODV résulte de plusieurs contraintes et qu'il est intéressant de combiner le maximum d'entre elles pour tirer les meilleurs profits.

### 5.3. Étude comparative d'AODV et OLSR :

Dans cette partie, on a gardé les mêmes paramètres de simulation utilisés dans la partie précédente, sauf dans ce cas, on a utilisé un temps de pause variables 0s, 10s, 20s, 40s, 100s entre chaque mouvement, dans une zone carrée de 1000x1000 m et avec un nombre de nœuds variables 25, 50, 75, 100 nœuds avec respectivement un trafic de 10, 15, 20, 25 connexions par réseau.

Vu le nombre important des résultats trouvés, nous présentons dans cette partie, brièvement les résultats de l'étude de l'impact de la charge et de la densité. Les résultats de l'étude de l'impact de la mobilité et de l'impact de la densité des nœuds mobiles sont brièvement cités dans la partie conclusion de ce chapitre.

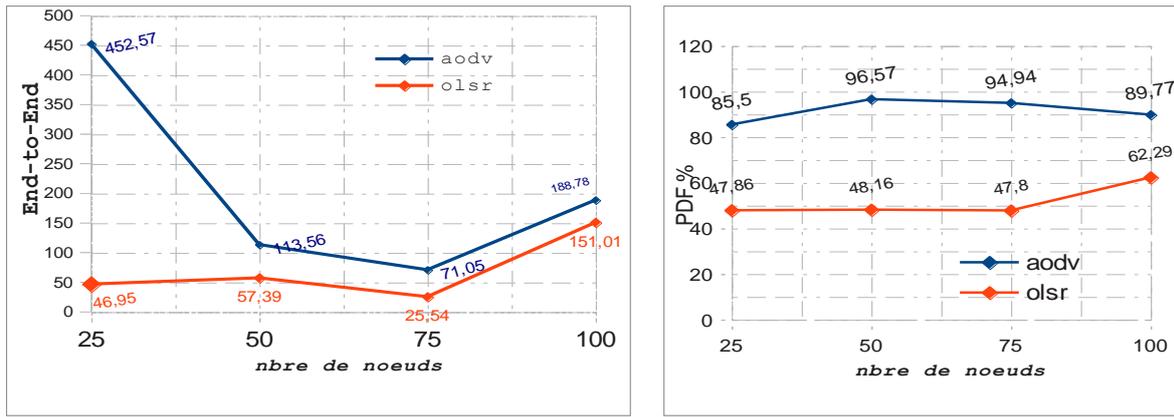


Figure 50: End to End Delay et PDF en fonction du nombre de nœuds

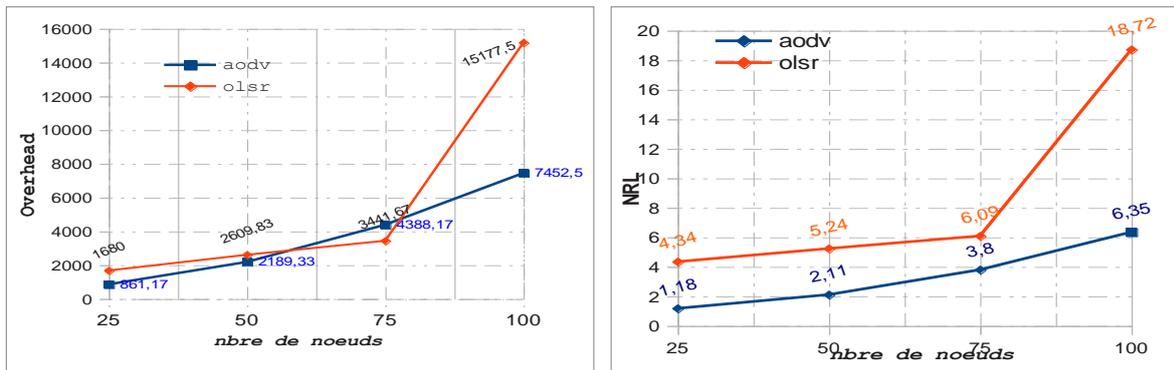


Figure 51: Overhead et NRL en fonction du nombre de nœuds

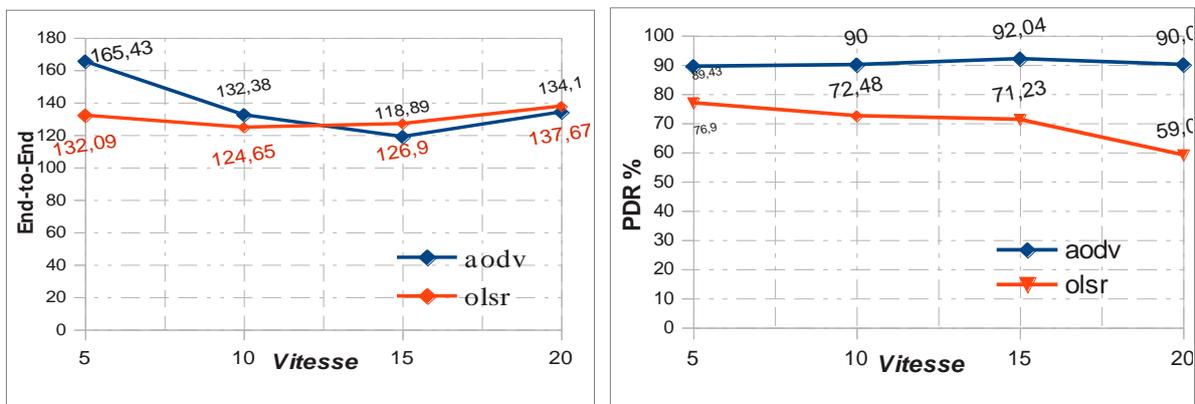


Figure 52: End to End Delay et PDF en fonction de la vitesse

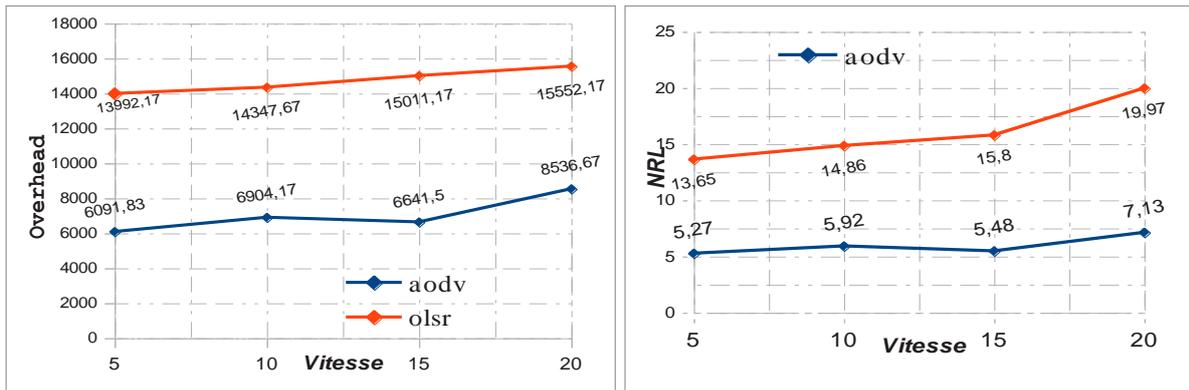


Figure 53: Overhead et NRL en fonction de la vitesse

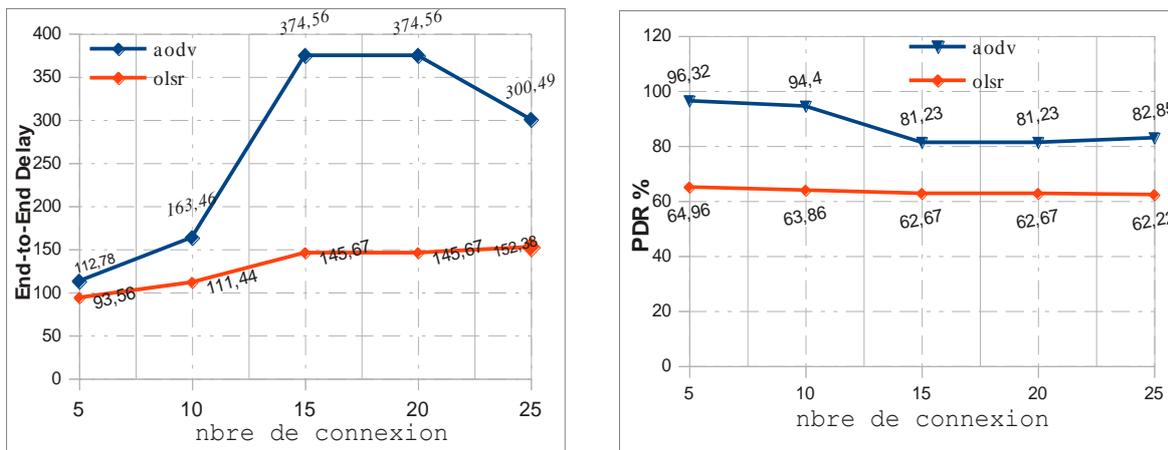


Figure 54: End to End Delay et PDF en fonction du Trafic

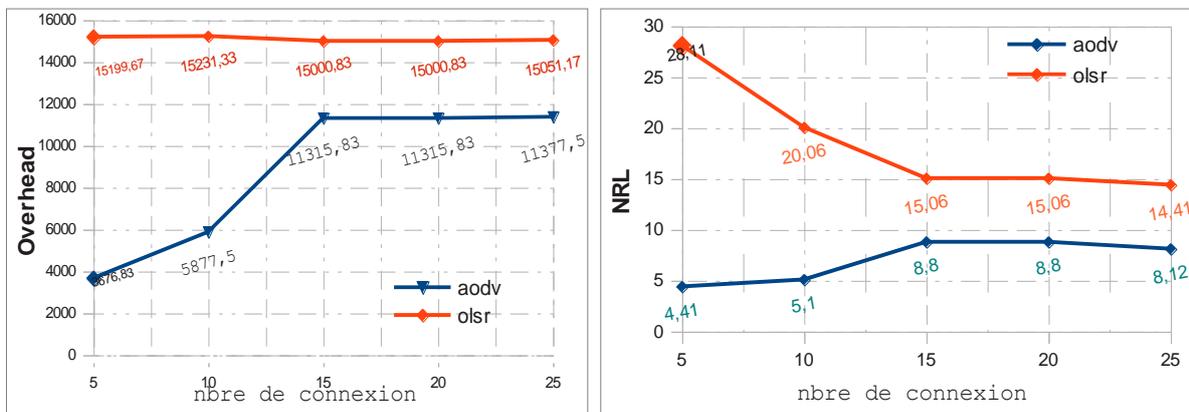


Figure 55: Overhead et NRL en fonction du Trafic

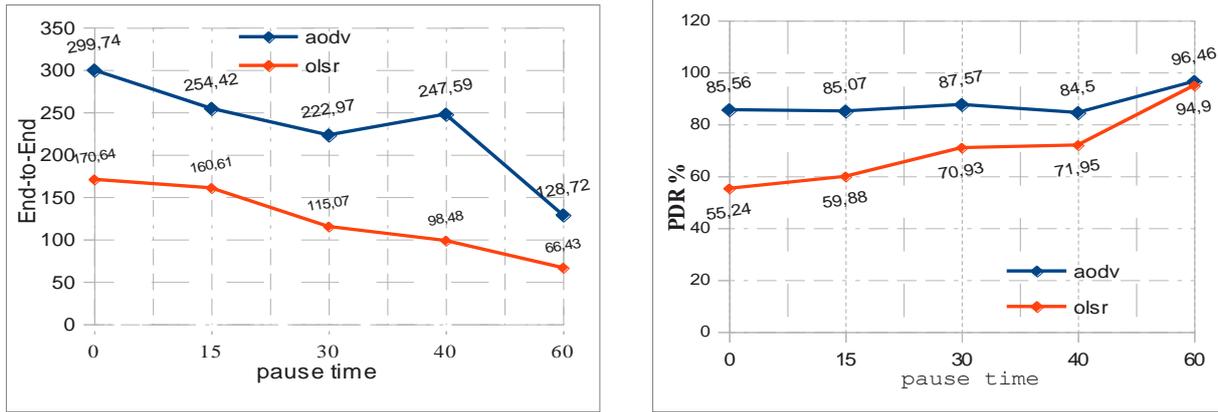


Figure 56: End to End Delay et PDF en fonction de temps de pause

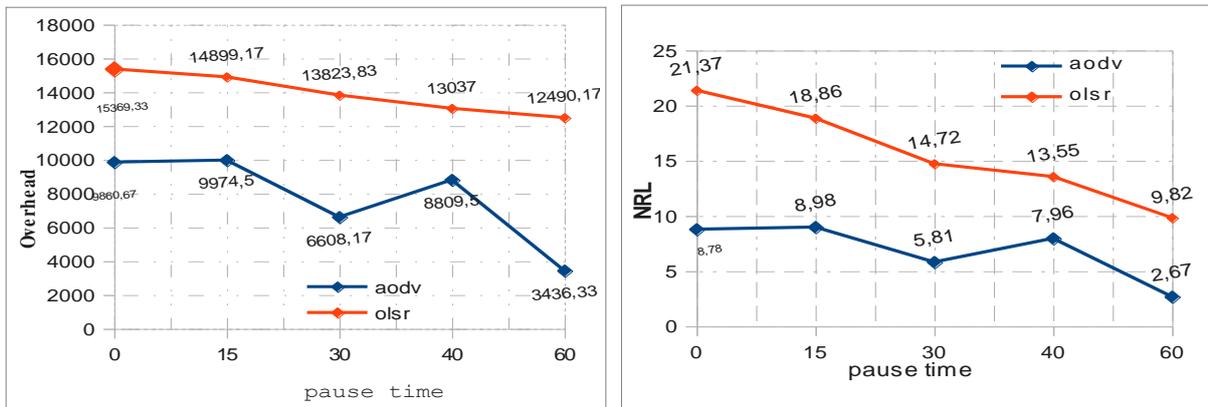


Figure 57: Overhead et NRL en fonction de temps de pause

D’après les résultats de la simulation, nous avons remarqué que :

En terme de la charge du réseau : OLSR génère plus de charge que le protocole AODV, à cause de l’aspect proactif du protocole, où les chemins entre les différents nœuds sont créés à l’avance. Cette charge influe grandement sur l’énergie des nœuds explicitement sur la durée de vie du réseau.

En terme du délai : à cause de l’aspect proactif, où les nœuds s’identifient d’échange entre eux, le protocole OLSR est meilleur que le protocole AODV.

Enfin, AODV représente le protocole le plus équilibré en termes de taux de livraison des paquets de données et de charge des paquets de contrôle selon les différents paramètres qu’on a choisis pour la simulation.

## 6. Conclusion :

Dans ce chapitre on a comparé les performances des protocoles de routage AODV, DSR, DSDV et OLSR pour les réseaux mobiles ad hoc en utilisant le simulateur ns-2. On a évalué différentes métriques tels que le taux d'arrivée des paquets de données, le temps moyen de transfert des paquets de données, et le nombre totales des paquets de contrôle dans le réseau. Le choix du protocole de routage à utiliser dans le réseau mobiles ad hoc reste ouvert, et dépend du critère à optimiser, chaque protocole à ses avantages et ses points faibles. Des évaluations comparatives des performances ont été réalisées pour mesurer le gain apporté par chaque protocole.

OLSR et AODV bien que de nature très différentes, sont très similaires en termes de performances. Dans un réseau très mobile, avec un changement fréquent de topologie, AODV a un petit avantage sur OLSR car les routes sont mises à jours plus rapidement. OLSR doit attendre plusieurs paquets Hello perdus avant de modifier l'état du lien et envoyer des informations de mises à jour. Par contre, dans un réseau plus statique, OLSR encombre moins le réseau qu'AODV qui émet beaucoup plus de messages à chaque découverte de route. En effet dans ce cas OLSR n'émet presque pas de message de mises à jour de la topologie. Dans un réseau très dense, OLSR charge moins le réseau qu'AODV.

Dans des réseaux moyens, OLSR et AODV sont équivalents. Lors de communications courtes, OLSR a un énorme avantage sur AODV car les routes sont disponibles immédiatement.

Dans la plupart des cas, les messages de contrôles d'AODV sont moins nombreux que ceux d'OLSR. OLSR émet d'autant plus de paquets de contrôle que le réseau est grand. OLSR est un peu supérieur à AODV car s'il est équivalent dans la plupart des réseaux, il est meilleur dans certains cas particuliers : comme des réseaux denses, ou des réseaux où le trafic est important et composé de nombreuses et courtes connexions.

## **Conclusion générale :**

Les réseaux mobiles ad hoc essayent d'étendre les notions de la mobilité à toutes les composantes de l'environnement, toutes les unités du réseau se déplacent librement et aucune administration centralisée n'est disponible.

La mobilité dans les réseaux mobiles a été un sujet d'actualité ces dernières années. Plusieurs solutions sont mises en place pour les réseaux avec infrastructure.

Lorsqu'il s'agit de réseaux mobiles ad hoc, il apparaît clairement que la mobilité doit être prise en charge par les nœuds.

Dans le but d'assurer la connectivité du réseau malgré l'absence d'infrastructure et la mobilité des stations, chaque nœud est susceptible d'être mis à contribution pour participer au routage et pour retransmettre les paquets d'un nœud qui n'est pas en mesure d'atteindre sa destination, tout nœud joue ainsi le rôle de station et de routeur. L'étude et la mise en œuvre de protocoles de routage pour assurer la connexion au sens classique du terme (tout sommet peut atteindre tout autre), est un problème très difficile. Les schémas de routage classique basé sur les localisations statiques des sites sont évidemment inadaptés dans un environnement mobile.

Ce travail entre dans le cadre de l'étude du problème de routage dans les environnements mobiles caractérisés par l'absence d'infrastructure. Différentes stratégies de routage conçues pour les réseaux mobiles ad hoc ont été étudiées. Selon la manière de construction de chemins entre les stations sources et les stations destination, les stratégies (ou les protocoles) de routage sont divisées en deux classes : les proactifs et les réactifs.

A travers notre étude des différents types de routage qui existent, nous avons vu que ces approches utilisent une variété de techniques afin de résoudre le problème de routage dans l'environnement des réseaux ad hoc. Cette variété de techniques fait que l'utilisation des solutions de routage qui existent dépend de la nature de l'application.

A partir des résultats trouvés de nos implémentations, nous constatons qu'il n'y a pas un protocole qui est favori par rapport aux autres dans tous les scénarios et les critères d'évaluation. Quelque soit la charge du réseau, la mobilité des nœuds et la densité du réseau Ad hoc, les protocoles DSDV et OLSR performant bien. Cela est expliqué par leurs caractéristiques proactives, cependant, ils s'adaptent mieux au passage à l'échelle (densité) par rapport aux protocoles réactifs. OLSR et AODV ont de meilleures performances dans les réseaux denses en termes de taux de paquets délivrés lorsqu'on les compare respectivement

avec DSDV et DSR, ce dernier utilise le cache pour la découverte de route, ce mécanisme dégrade les performances de DSR en particulier dans les réseaux dense.

Les résultats obtenus par la simulation doivent être pris comme une indication pertinente sur le comportement de ces quatre protocoles de routage Ad hoc et non comme une représentation exacte de son comportement en environnement réel, compte tenu de plusieurs contraintes de simulation à savoir la dimension du champs de mouvement des nœuds mobiles, le modèle de mobilité, le nombre de nœuds mobiles, le type de trafic, le temps de simulation etc.

## **Perspectives :**

Nous avons étudié les modèles de mobilité pour les réseaux mobiles ad hoc. Vu que ces réseaux deviennent de plus en plus connus et que beaucoup d'études existent dans ce domaine, l'utilisation de la simulation pour faire les mesures est indispensable puisqu'on est incapable de réaliser la plupart des scénarios dans la réalité. Nous avons fait des simulations en utilisant des métriques liées à la mobilité et à la connectivité des nœuds pour caractériser ces modèles de mobilité.

Donc puisque la plupart des modèles de mobilité sont le cas de scénarios à réaliser, l'intérêt sera de penser à de nouveaux scénarios et d'essayer de trouver de nouveaux modèles de mobilité intéressant avec des nouveaux protocoles de routage.

Donc plus on a des modèles de mobilité, plus on pourra simuler plus de scénarios et avoir des résultats plus variés notamment avec d'autres simulateurs, ce qui favorisera le passage à l'échelle ou la « scalabilité » (simuler des réseaux dont la taille peut atteindre 10.000 nœuds). Car actuellement, il existe plusieurs simulateurs de réseaux (en particulier les réseaux mobiles ad hoc) de popularité différente. Chaque simulateur se base sur une méthodologie et un modèle propre à lui pour pouvoir modéliser un réseau.

Dans les études futures nous essaieront d'évaluer les performances des protocoles de routage Ad hoc avec différents modèles de mobilité et différents types de trafic notamment le trafic de type TCP.

Une des perspectives de ce travail sera d'améliorer les stratégies (protocoles) de routage existantes, et de proposer éventuellement un nouveau protocole de routage qui sera capable de prendre en compte les limitations des réseaux mobiles ad hoc et ainsi d'offrir une meilleure adaptation à la mobilité de ces environnements.

## Procédure d'installation de NS-2 sous Linux (Kubuntu Version11.10) :

On installe tout d'abord les paquets de dépendances :

```
$ sudo apt-get update
$ sudo apt-get install build-essential autoconf automake libxmu-dev
```

Et on décompresse le Paquet «ns-allinone-2.35 »

```
$ tar -zxvf ns-allinone-2.35.tar.gz
```

Et on installe le programme après l'accès au dossier ns-allinone-2.35 :

```
$ cd ns-allinone-2.35
$ sudo ./install
```

A la fin de l'installation, il nous sera demandé d'installer nous même les variables d'environnements. Donc on édite le fichier bashrc:

```
$ kate ~/.bashrc &
```

On Ajoute les lignes suivantes à la fin de ce fichier

```
# LD LIBRARY PATH
OTCL_LIB=/home/dabo/Bureau/adhoc/ns-allinone-2.35/otcl-1.14
NS2_LIB=/home/dabo/Bureau/adhoc/ns-allinone-2.35/lib
X11_LIB=/usr/X11R6/lib
USR_LOCAL_LIB=/usr/local/lib
export LD LIBRARY PATH=$LD LIBRARY PATH:$OTCL_LIB:$NS2_LIB:$X11_LIB:$USR_LOCAL_LIB

# TCL LIBRARY
TCL_LIB=/home/dabo/Bureau/adhoc/ns-allinone-2.35/tcl8.5.10/library
USR_LIB=/usr/lib
export TCL LIBRARY=$TCL_LIB:$USR_LIB

# PATH
XGRAPH=/home/dabo/Bureau/adhoc/ns-allinone-2.35/bin:/home/dabo/Bureau/adhoc/ns-allinone-2.35/tcl8.5.10/unix:/home/dabo/Bureau/adhoc/ns-allinone-2.35/tk8.5.10/unix
NS=/home/dabo/Bureau/adhoc/ns-allinone-2.35/ns-2.35/
NAM=/home/dabo/Bureau/adhoc/ns-allinone-2.35/nam-1.15/
PATH=$PATH:$XGRAPH:$NS:$NAM
```

Pour que ces modifications prennent effet immédiatement on tape ceci :

```
$ source ~/.bashrc
```

Pour la validation de ns, on se rend dans le répertoire ns du pack allinone :

```
$ cd ns-2.35
$ ./validate
```

## ***Bibliographie :***

- [1] Agha, K. A., Pujolle, G., Vivier. Réseaux de mobiles et réseaux sans fil. Eyrolles, 2002
- [2] Mobile ad hoc network. [Http://www.ietf.org/html.charters/manetcharter.html](http://www.ietf.org/html.charters/manetcharter.html), 2012
- [3] Tayeb Lemlouma, Les réseaux ad hoc, Université USTHB, Houari Boumediene, 2000
- [4] Guy Pujolle – Les Réseaux édition 2005, Groupe Eyrolles 2004
- [5] Bruno, R., M. Conti et E. Gregori. 2005. « Mesh networks: commodity multihop ad hoc networks ». IEEE Communications Magazine, 2000
- [6] Leiner, B. M., R. J. Ruther et A. R. Sastry. 1996. « Goals and challenges of the DARPA GloMo program [global mobile information systems] ». IEEE Personal Communications
- [7] S. Corson, J. Macker, 'Mobile Ad hoc Networking (MANET), IETF, January 1999
- [8] Agha, K. A., Pujolle, G., Vivier. Réseaux de mobiles et sans fil. Eyrolles, 2008
- [9] B. Tavli, W. Heinzelman; "Mobile Ad Hoc Networks: Energy-Efficient Real-Time Data Communications"; Netherlands, Springer, 2006
- [10] Khennouche Dib.-Le routage dans les réseaux Ad Hoc, Univ Houari Boumediene(USTHB), 2004
- [11] M. Sedrati, L. Aouragh, L. Guettala & A. Bilami – « Etude des performances des protocoles de routage dans les réseaux mobiles ad-hoc », Publication, 2007
- [12] F. Ducatelle; "Adaptive Routing in Ad Hoc Wireless Multi-hop Networks", these, 2007
- [13] R. Meraihi ; "Qualité de service et topologie dans les réseaux ad hoc" ; Thèse, École nationale supérieure des télécommunications, Paris, 2004
- [14] Cormen, T. H., C. E., Rivest, R. L. Introduction à l'algorithmique: Cours Dunod, 2002
- [15] C.-K. Toh; "Mobile computing in wireless ad hoc networks"; IEEE Magazine, 2011
- [16] Charles E. Perkins, Elizabeth M. Royer: Auto-configuration for Ad Hoc Networks, Mobile Ad Hoc Networking Working Group, July 2000
- [17] S. Corson; "Routing protocol performance issues and evaluation Considerations"; 2010
- [18] H. Labiod, "Mobile Ad Hoc Networks: Unicast routing, QoS and multicast routing, 2006
- [19] C. Perkins, E. Belding-Royer, Routing, Network Working Group, 2003
- [20] P.Santi; "Topology Control in Ad Hoc and Sensor Networks"; England, 2005.
- [21] M. Abolhasan, T. Wysocki & E. Dutk. «A review of routing in ad hoc networks», 2008
- [22] David B. Johnson, « Routing in Ad Hoc Networks of Mobile Hosts », 1999
- [23] Chlamtac, M. Conti, and J. Liu. "Mobile ad hoc networking: imperatives and challenges". P64, 2003.
- [24] Paul Mühlethaler, 802.11 et les réseaux sans fil, Eyrolles, 2002.
- [25] P. Jacquet and T. Clausen, Optimized link state routing protocol, Project Hipercom, INRIA (2003).
- [26] G.Pujolle "les réseaux ", eyrolles édition 2008
- [27] C. Servin "RÉSEAUX ET TÉLÉCOMS" édition Dunod 2003
- [28] C.Perkins, E. Belding-Royer "Ad-hoc on-demand distance vector (AODV) routing". RFC 3561, IETF 2003, <http://rfc3561.x42.com>
- [29] D. Johnson, Y. Hu and D.Maltz. "The dynamic source routing for MANETS" RFC 4728, IETF 2007
- [30] C.E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance vector routing (DSDV) for mobile computers," in Proc. ACM SIGCOMM 94, London, UK, Oct. 1994, pp. 234-244
- [31] T. Clausen, P. Jacquet "Optimized Link State Routing Protocol (OLSR)" RFC 3626, IETF 2003, <http://rfc3626.x42.com>
- [32] N. Badache, D. Djenour & A. Derhab «Les réseaux ad hoc», Revue Scientifique (RIST), 2002
- [33] M.R. Pearlman, Z. J. Haas. "Determining the Optimal Configuration for the Zone Routing Protocol" journal IEEE 1999, page 1395-1414

- [34] Anis KOUBAA, Introduction à l'évaluation de Performance des Systèmes informatiques et de Communication, Janvier 2004, Nancy France
- [35] Programmation en Tcl/Otcl: <http://www.msen.com/~clif/TclTutor.html>  
<http://www.linbox.com/ucome.rvt/any/fr/Ressources/Cours/tcl/cour1.pdf>  
Programmation en Tcl/Otcl: <http://www.msen.com/~clif/TclTutor.html>
- [36] P. Anelli, Introduction à NS, Université de la Réunion, 2006,  
<http://www2.univreunion.fr/~panelli/enseignement/TP-NS-2/Doc/IntroNS.pdf>
- [37] P. Anelli & E. Horlait, NS-2: Principes de conception et d'utilisation,  
<http://www2.univreunion.fr/~panelli/enseignement/TP-NS-2/2-documentation.html>  
Tutorial NS: <http://www.isi.edu/nsnam/ns/tutorial/>
- [38] Arnold D. Robbins, GAWK: Effective AWK Programming-User's Guide, June, 2004.
- [39] Thomas Williams & Colin Kelley, gnuplot An Interactive Plotting Program, manual of gnuplot, 26 August 2007, <http://sourceforge.net/projects/gnuplot>.
- [40] Fall et K.Varadhan. 2008. « The ns Manual ». The VINT Project, UC Berkeley, USC/ISI, Radio Propagation Models: <http://www.isi.edu/nsnam/ns/doc/node216.html>

## Liste des publications

### 1 Revues Internationales:

- 1- K. Ouididi, A. Hajami, M. Elkoutbi "*QoS Routing For Mobile Adhoc Networks And Performance Analysis Using OLSR Protocol* " (IJCSIS) International Journal of Computer Science and Information Security April 2010, Vol. 8 No. 1 pp138-150.
- 2- Abdelmajid Hajami, Kamal Ouididi, Mohammed Elkoutbi "*A Distributed Key Management Scheme based on Multi hop Clustering Algorithm for MANETs*" (IJCSNS) International Journal of Computer Science and Network Security 2010 VOL.10 No.1, January 2010 pp. 39-49
- 3- K.Ouididi, A.Habbani and M.Elkoutbi , "*Using Mobility to Enhance the Routing process in the MIS System*": International Journal of Interactive Mobile Technologies (iJIM), Special Issue: Technical Basics ISSN: 1865-7923,vol 3 2009 pp24-32
- 4- N.Enneya, K.Ouididi, and M.El Koutbi "*Enhancing Delay in MANET Using OLSR Protocol*", Int. J. Communications, Network and System Sciences ijens- vol.2,no.4,May 2009 pp392-399.
- 5- K. Ouididi, A. Hajami, M. Elkoutbi "*QoS Routing Using OLSR Protocol*" proceedings and our ISI book for> the WSEAS Conferences in Corfu Island, Greece, July 2010 (to appear)

### 2 Conférences Internationales

- 6- K.oudidi, A.Hajami and M.Elkoutbi "*QoS Routing Using OLSR Protocol*" in proceedings of the IASTED International Conference on Wireless and Optical Communications (WC'2010, Alberta Banff, Canada, July 14-15, 2010. (ActaPress)
- 7- A.Hajami, K.oudidi and M.Elkoutbi "*An enhanced algorithm for MANET clustering based on Multi hops and Network Density*" In proceedings of NOTERE. May 2010 (to appear)
- 8- K.oudidi, A.Hajami and M.Elkoutbi "*A New Composite Metric For QoS Satisfying Both Mobility And Bandwidth Constraints In Manets*" In proceedings of African Conference on Research in Computer Science and Applied Mathematics CARI'10, Yamoussoukro- 2010, Rabat, Côte d'Ivoire. (to appear)
- 9- K.oudidi, A.Hajami and M.Elkoutbi "*Towards a QoS Aware Routing protocol For MANETs*" in Proceedings of the International Conference Next Generation Networks (NGNs'10) , Mararkech, Maroc, 2010 (to appear)
- 10- K.Ouididi, M.Elkoutbi "*A Mobility Based Metric For Qos In Mobile Ad Hoc Networks*" Proceedings of the IASTED International Conference on Wireless and Optical Communications (WOC'09), ISBN: 978-0-88986-793-2 Banff, Canada, July 6-8, 2009 pp11-18 (published at ActaPress)
- 11- K.Ouididi, M.ElKoutbi, "*Une métrique de QoS à Contraintes Multiples pour les MANETs*" in Proceedings of the International Conference Next Generation Networks (NGNs'09) June 2009 pp 74-82