

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE KASDI MERBAH OUARGLA

**Faculté des Nouvelles Technologies
de l'Information et de la
Communication**



**Département d'Informatique et
Technologie de l'Information**

Mémoire
En vue de l'obtention du diplôme de
Master en Informatique
Spécialité : Informatique Industriel

Présenté par :
Mohamed djihad BEN SALEM
Oussama BOUGOFFA

Thème

**Etude comparative de deux simulateurs pour les
réseaux ad-hoc sans fil**

Soutenu publiquement 14/06/2014

Dirigé par : M me. Wassila KORICHI

Fares KAHLESSENANE
Kamel BENKADDOUR

Examineur
Examineur

UKM Ouargla
UKM Ouargla

L'année universitaire : 2013/2014

Dédicace

*Je dédie cette
Mémoire ... ✍*



***MA TRÈS CHÈRE MÈRE
MON TRÈS CHER PÈRE***

*Autant de phrases et d'expressions aussi éloquents soit-elles ne
sauraient exprimer ma gratitude et ma reconnaissance.*

*Tu as su m'inculquer le sens de la responsabilité, de l'optimisme
et de la confiance en soi face aux difficultés de la vie.*

Tes conseils ont toujours guidé mes pas vers la réussite.

*Ta patience sans fin, ta compréhension et ton encouragement
sont pour moi le soutien indispensable que tu as toujours su m'apporter.*

ma grande famille

*je cite en particulier BENSALÈM,
mes tantes, mes oncles ainsi que mes cousins et cousines.*

ma grande frères

*je cite en particulier Hakima , Iman ,
mes tantes, mes oncles ainsi que mes cousins et cousines*

Merci de m'avoir accueilli parmi vous.

***Puisse ce travail témoigner de ma profonde affection et de ma
sincère estime.***

Remerciement

En premier lieu , nous remercions Dieu le tout puissant qui nous

A procuré ce succès .

Nous remercions nos parents qui nous ont soutenus tout au

Long de notre cursus .

Notre gratitude s'adresse particulièrement à notre promotricez

Mme Korichi qui nous a proposé ce sujet , encadré et pour sa
Disponibilité , ses conseils judicieux et ses encouragements qu' elle

Nous prodigués avec patience et gentillesse tout au long de la

Réalisation de ce travail,

Enfin, nous tenons à remercier en leur étant très

Reconnaissant tous ce qui nous ont aidé , encouragé et soutenue

De près ou de loin pendant cette année

Mohamed djihad BENSALÉM & oussama BOUGOFFA

Résumé :

Un réseau mobile ad hoc est un ensemble de nœuds mobiles connectés par des liens radio sans fil. Ces environnements présentent l'avantage d'un déploiement rapide et peu coûteux puisqu'ils ne nécessitent l'installation d'aucune infrastructure. Beaucoup de travaux de recherche se sont penchés dans le domaine des réseaux mobile ad hoc, pour développer de nouveaux protocoles et nouvelle application, qui utilisant les particularités sans fil sont proposés.

Afin de tester l'efficacité d'un nouveau protocole il est indispensable d'étudier son comportement dans un environnement proche de l'environnement réel. Cependant la reproduction très coûteuse d'un tel système a motivé la naissance des simulateurs de réseaux. La simulation permet non seulement de tester sans aucun coût de nouvelles technologies et de nouveaux protocoles, mais aussi d'anticiper les problèmes qui pourraient se poser dans le futur. Les environnements de simulation nous offrent un milieu d'exécution très proche de la réalité (dans la limite du possible), pour obtenir des résultats permettant d'envisager des corrections utiles et constructives [BTA00] [KAK 02]. Ce travail entre dans le cadre de l'étude des simulateurs dans les réseaux Ad Hoc, notre étude consiste à : comparer les deux simulateurs, « NS-2 » et « Glomosim » qui ont été testé par plusieurs scientifiques.

المخلص :

الشبكة اللاسلكية التي تدعى « AD HOC » هي عبارة عن مجموعة من العقد أو الوحدات و التي تمثل جهاز حاسوب أو هاتف نقال متصلة فيما بينها بواسطة الوصلات الراديو اللاسلكية . و تتميز بالانتشار السريع و غير مكلفة لأنها لا تتطلب تثبيت أي جهاز خاص بالبنية تحتية للشبكة . و قد ركزت الأبحاث في مجال الشبكات اللاسلكية « AD HOC » و المخصصة لوضع بروتوكولات جديدة و التطبيقات الجديدة . لأختبار أداء و فعالية بروتوكول معين ضمن الشبكة . و لا بد من الدراسة في بيئة قريبة من البيئة الحقيقية و هذا يتطلب أجهزة و موارد بشرية للدراسة و هذا مما يجعلها مكلفة . و من هذا ظهرت بما يسمى ببرامج محاكات الشبكة اللاسلكية.

بيئة المحاكات تقدم لنا بيئة شبه واقعية تمكننا من الحصول على نتائج تقريبية صحيحة و دقة النتائج تختلف من برنامج محاكي الى برنامج محاكي آخر , و لهذا في هذه المذكرة سنقوم بالمقارنة النظرية و المقارنة التطبيقية بين برنامجي المحاكات « NS-2 » و « GloMoSim ».

Table de matrice

Chapiter I. Les réseaux mobile ad hoc	15
I. Introduction.....	16
II. Les environnements mobiles.....	17
1. Classification des réseaux mobiles	17
1.1 Les réseaux avec infrastructure.....	17
1.2 Les réseaux sans infrastructure.....	18
2. L'utilisation des ondes radio dans la communication sans fil	19
3. La fiabilité de la communication sans fil	19
4. La communication cellulaire.....	20
5. Quelques éléments de l'infrastructure sans fil.....	21
III. Les réseaux mobile ad hoc	22
1. Définition.....	22
2. Les applications des réseaux mobiles ad hoc.....	23
3. Les caractéristiques des réseaux ad hoc	24
3.1 Une topologie dynamique :.....	24
3.2 Une bande passante limitée :.....	24
3.3 Des contraintes d'énergie :.....	24
3.4 L'absence d'infrastructure :.....	24
IV. Le protocole de routage de réseaux ad hoc.....	25
1. Les protocoles de routage pro-actifs.....	25
1.1 Présentation	25
1.2 <i>Le protocole de routage DSDV.....</i>	<i>25</i>
1.3 <i>Le protocole de routage WRP.....</i>	<i>26</i>
1.4 <i>Le protocole de routage GSR</i>	<i>26</i>
1.5 <i>Le protocole de routage FSR.....</i>	<i>26</i>
1.6 <i>Le protocole de routage HSR</i>	<i>27</i>
1.7 <i>Le protocole de routage ZHLS.....</i>	<i>27</i>
1.8 <i>Le protocole de routage CGSR</i>	<i>27</i>
1.9 <i>Le protocole de routage DREAM</i>	<i>27</i>
2. Les protocoles de routage réactifs (à la demande).....	28

2.1	Présentation	28
2.2	<i>Le protocole de routage CBRP</i>	28
2.3	<i>Le protocole de routage DSR</i>	28
2.4	<i>Le protocole de routage AODV</i>	29
2.5	<i>Le protocole de routage TORA</i>	29
2.6	<i>Le protocole de routage ABR</i>	29
2.7	<i>Le protocole de routage SSR</i>	29
3.	Avantages et inconvénients des réseaux mobiles Ad hoc :	29
3.1	<i>Avantage</i> :	30
3.2	<i>Inconvénients</i> :	31
V.	Conclusion	32
Capiter II. Présentation des deux simulateurs« NS-2 » et « GloMoSim »		33
I.	Introduction	34
II.	Environnement de simulation	35
1.	Introduction à la simulation	35
2.	Système réel et objectif de simulation	36
3.	Modèles de simulation	36
3.1	Gestion du temps	37
3.2	Simulation par événements discrets	37
3.3	Simulateur	38
4.	Avantages et inconvénients de la simulation	38
➤	Avantage :	38
➤	Inconvénients :	38
III.	NS-2 (Network simulateur)	39
1.	Présentation de network simulator NS-2	39
2.	Le langage TCL (Tool Command Language)	39
2.	Otcl	40
3.	NAM	40
4.	Le modèle de réseau sous ns-2	40
3	Caractéristiques d'une entité communicante sous NS-2	41
3.1	Le Classifier	41
3.2	Le lien	42
3.3	L'agent	42
4	Le processus de simulation	43

5.	Création d'un scénario	46
5.	L'avantage et inconvénients du simulateur NS-2.....	47
5.1	Avantage du simulateur NS-2 :.....	47
5.2	Les inconvénients :	48
IV.	GloMoSim (Global Mobile information system Simulator)	48
1.	Présentation générale	48
2.	PARSEC	48
2.1	Entité :	49
2.2	Message :.....	49
2.3	Événement :.....	50
2.4	L'exécution parallèle :.....	50
3.	Structure des répertoires de GloMoSim	50
4.	Architecteur GloMoSim [JNU 04]	51
5.	Utilisation de Simulateur GloMoSim	51
6.	L'outil de visualisation	52
7.	Configuration de un scénario	52
7.1	Fichier de configuration (Config.in).....	52
V.	Conclusion	56
Capitre III. Comparaison qualitative entre « NS-2 » &« GloMoSim »		57
I.	Introduction.....	58
1.	Placement des nœuds	58
1.1	Random :	58
1.2	Uniform :	58
1.3	Grid:.....	58
1.4	A partir d'un fichier :	58
1.5	Manuellement :	58
2.	Scénario de mobilité.....	59
2.1	Aucun :.....	59
2.2	Randomwaypoint :	59
2.3	Fichier de trace :	59
2.4	Randomdrunken :.....	59
3.	Terrain physique.....	59
	Flat grid :	59
	Un plan bitmap :.....	59

.4	Génération de trafic	60
	A partir d'un fichier :	60
	Manuellement :	60
5.	Modules implémentés.....	60
6.	Comparaison des paramètres de protocole entre les deux simulateurs	62
II.	Conclusion	62
Capitre IV. Comparaison quantitative entre « NS-2 » et « GloMoSim »		63
I.	Introduction.....	64
II.	Paramètres de simulation	64
III.	Portée de transmission.....	65
1.	Description	65
2.	Résultats de simulation	65
2.1	Appliqué le scénario dans le simulateur NS-2.....	66
2.2	Appliqué le scénario dans le simulateur GloMoSim.....	69
3.	Etude comparative	70
IV.	Méthode d'accès au médium.....	71
.1	Description	71
2.	Résultats de simulation :	71
2.1	Appliqué le scénario dans le simulateur NS-2 :	72
2.2	Appliqué le scénario dans le simulateur GloMoSim :.....	73
4.	Etude comparative	74
V.	Scalabilité, simulation et autres facteurs	75
.1	Description	75
2.	Résultats de simulation	75
2.1	Appliqué le scénario dans le simulateur NS-2 :	75
2.2	Appliqué le scénario dans le simulateur Glomosim :	75
3.	Etude comparative	75
VI.	Conclusion	78

Liste de figure

Figure I-1 : Le modèle des réseaux mobiles avec infrastructure.....	18
Figure I-2 : Le modèle des réseaux mobiles sans infrastructure.....	18
Figure I-3 : Le principe de réutilisation de fréquence.....	21
Figure I-4 : La modélisation d'un réseau ad hoc.....	23
Figure I-5 : Le changement de la topologie des réseaux ad hoc.....	23
Figure I-6 : Classification des protocoles.....	25
Figure II-1 : Cycle modélisation-simulation.....	34
Figure II-2 : L'architecture générale du NS2.....	41
Figure II-3 : Schéma d'un nœud dans NS2.....	43
Figure II-4 : Le processus de simulation NS-2.....	45
Figure II-5 : Architecteur GloMoSim.....	51
Figure II-6 : fichier NODES.INPUT.....	53
Figure II-7 : fichier config.in.....	55
Figure IV-1 : Scénario pour la vérification de la portée de transmission.....	66
Figure IV-2 : simulation_rés_aadv.tcl.....	66
Figure IV-3 : Exécution fichier .tcl.....	67
Figure IV-4 : NAM, visualisation représenter la topologie d'un réseau décrit avec NS-2.....	67
Figure IV-5 : NAM, visualisation représenter la topologie d'un réseau décrit avec NS-2.....	68
Figure IV-6 : Traces de la simulation.....	68
Figure IV-7 : Traces de la simulation.....	69
Figure IV-8 : app.CONFIG pour configure le flux CBR.....	69
Figure IV-9 : compiler le fichier config.in.....	69
Figure IV-10 : résultat de simulation.....	70
Figure IV-11 : Nombre de paquets reçus par le mobile N1 en fonction de la distance (N0, N1).....	70
Figure IV-12 : Scénario de test de la méthode d'accès.....	72
Figure IV-13: Le script fichier.tcl parti définition des variables.....	72
Figure IV-14 : fichier.tcl parti configuration paramètre les nœud.....	72
Figure IV-15 : fichier.tcl.....	73
Figure IV-16 : fichier.tr.....	73
Figure IV-17 : fichier app.config.....	74
Figure IV-18 : résultat de la simulation.....	74
Figure IV-19fichier.tcl définie les variable pour les paramètres.....	75
Figure IV-20 : Fichier APP . CONFIG paramètre de flux CBR.....	75
Figure IV-21 : Taux paquet perdu en fonction de la mobilité.....	76
Figure IV-22 : Taux paquets perdus en fonction du débit.....	77

Liste de Table

Table II-1 : Table de configuration de nœud.....	52
Table III-1 : Comparatif entre NS-2 et GloMoSim dans le placement des nœuds.....	58
Table III-2 : Comparatif entre NS-2 et GloMoSim dans le Scénario de mobilité.....	59
Table III-3 : Comparatif entre NS-2 et GloMoSim dans le modèle du terrain physique	60
Table III-4 : Comparatif entre NS-2 et GloMoSim La génération du trafic.....	60
Table III-5 : Ensemble des modules implémentés par les deux simulateurs GloMoSim et NS-2.....	61
Table III-6 : Comparaison des simulateurs NS-2 & GloMoSim de réseau	61
Table III-7 : Comparatif entre NS-2 et GloMoSim dans les paramètres de protocole implémenter	62
Table IV-1 : Paramètres communs aux scénarii.....	64

Introduction générale

Les réseaux de données sans fil ont connu une véritable explosion depuis la fin des années 90 aussi bien dans la vie de tous les jours pour se connecter à l'Internet que dans le monde de la recherche. L'environnement sans fil présente de nombreuses différences avec le monde des réseaux filaires en particulier au niveau des couches basses de communications que sont les couches physiques et liaisons de données. Les communications s'effectuant à l'aide d'un signal radio, les risques de voir ce signal brouillé ou atténué sont très importants. Un autre point important est que la topologie de ce type de réseau peut évoluer de manière très dynamique car les terminaux composant un réseau sans fil peuvent être mobiles.

Les constituants des réseaux 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.

Pour pouvoir tester et évaluer des travaux dans cet environnement, Il existe différentes techniques. Parmi elles, on peut citer la modélisation analytique, les mesures obtenues à partir d'expériences réelles, et la simulation.

➤ La modélisation analytique consiste à représenter les conditions réelles de façon formelle à l'aide d'outils mathématiques.

➤ La validation de protocoles et d'applications pour **MANets** (réseaux mobile ad hoc Net work) par des expériences réelles est complexe à mettre en œuvre. En effet, la mise en œuvre de MANets pour la validation est problématique pour plusieurs raisons :

- les terminaux étant mobiles, on a besoin d'un opérateur, humain ou robot, par terminal, capable de reproduire des schémas de mobilité et des actions.
- les expérimentations sont difficilement reproductibles. Les communications étant radio, elles peuvent être perturbées par des signaux extérieurs sur lesquels l'expérimentateur n'a pas de contrôle.
- L'étude du passage à l'échelle, de la variation de la vitesse et du modèle de mobilité des utilisateurs est difficile à réaliser.

➤ La simulation consiste à modéliser la globalité du système étudié et à le simuler numériquement à l'aide d'environnements provenant de mesures sur un système réel ou de modèles probabilistes. L'intérêt de la simulation est de pouvoir travailler sur des systèmes non disponibles. Par exemple, lors de l'étape de conception, il est beaucoup moins coûteux de réaliser une simulation préalable des alternatives envisagées. De plus, la simulation est un moyen très souple d'étudier un problème. Cette technique permet des réexecutions de programmes avec changement de paramètres et une prise de trace d'exécution sans perturbation.

L'évaluation des performances d'un système via une simulation consiste en: (i) le choix d'un modèle, (ii) l'évaluation par une technique de simulation et, (iii) l'interprétation des mesures recueillies. Un grand nombre de modèles de simulation ont été développés pour l'étude d'architectures et de protocoles sous divers scénarios réseaux (nombre de nœuds, mobilité, ...). Ils ont été largement utilisés pour l'évaluation des protocoles de routage.

Les réseaux mobiles peuvent être simulés à l'aide de langages de programmation à utilisation générale tels que C, C++ et JAVA ou de langages de simulation tels que MODSIM III, SIMSCRIPT II.5 et SLAM II. Dans la seconde moitié des années 90, avec l'élaboration de plusieurs normes pour les réseaux sans fil à portée limitée, un certain nombre de simulateurs ont été développés conjointement. On cite, par exemple, Network Simulator 2 [Site 2], et GloMoSim [Site 1]. « NS-2 » et « Glomosim » est certainement le simulateur de réseaux le plus utilisé par la communauté de chercheurs travaillant sur les environnements mobiles sans fil.

Cependant, choisir un simulateur parmi les autres est un problème, sur quelle base et comment un chercheur choisir ce simulateur est non pas l'autre ?

Dans ce travail nous allons faire une étude comparative entre les deux simulateurs de réseaux les plus utilisés par la communauté de chercheurs travaillant sur les environnements mobiles sans fil, qu'ils sont certainement « NS-2 » et « Glomosim »

Ce document est composé de quatre chapitres. Le premier chapitre présente les environnements mobiles et en particulier les environnements mobiles ad hoc. Il met en évidence les caractéristiques des communications sans fil et leurs conséquences sur le

fonctionnement de ces réseaux. Les problèmes que posent ces environnements pour l'implantation de nouveaux services, sont alors clairement définis. Le deuxième chapitre nous décrit les caractéristiques et les capacités des deux simulateurs « NS-2 » et « GloMoSim ». Le troisième chapitre consiste à comparer les deux simulateurs d'une manière préliminaire ou qualitative des deux simulateurs. Dans le dernier chapitre, nous procurions aux simulations pour pouvoir compares les deux simulateurs en terme de résultats et de temps de simulation.

Chapiter I. Les réseaux mobile ad hoc

Chapitre « 1 » : Les réseaux mobiles ad hoc

I. Introduction

L'essor des technologies sans fil offre aujourd'hui des perspectives intéressantes dans le domaine des télécommunications. 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 autonomie, puissance limitée, etc.) qui accèdent au réseau par le biais d'une interface de communication sans fil.

Les environnements mobiles offrent une grande flexibilité d'emploi. En particulier, ils permettent la mise en réseau de sites dont le câblage serait trop onéreux à réaliser dans leur totalité, voire même impossible (par exemple en présence d'une composante mobile). 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 ad hoc. Plusieurs systèmes utilisent déjà le modèle cellulaire et connaissent une très forte expansion à l'heure actuelle (les réseaux GSM par exemple) mais requièrent une importante infrastructure logistique et matérielle fixe.

L'alternative à ces réseaux cellulaires sont les réseaux mobiles ad hoc. Un réseau ad hoc peut être défini comme une collection d'entités mobiles interconnectées par une technologie sans fil formant un réseau temporaire sans l'aide de toute administration ou de tout support fixe.

Dans un réseau ad hoc les hôtes mobiles doivent former, d'une manière ad hoc, une sorte d'architecture globale qui peut être utilisées comme infrastructure du système. Les applications des réseaux ad hoc sont nombreuses, on cite l'exemple classique de leur application dans le domaine militaire et les autres applications de tactique comme les opérations de secours et les missions d'exploration.

II. Les environnements mobiles

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

1. Classification des réseaux mobiles

Les réseaux mobiles ou sans fil, peuvent être classés en deux classes : les réseaux avec infrastructure et les réseaux sans infrastructure.

1.1 Les réseaux avec infrastructure

Le modèle de système intégrant des sites mobiles et qui a tendance à se généraliser, est composé de deux ensembles d'entités distinctes : les "sites fixes" d'un réseau de communication filaire classique (wired network), et les "sites mobiles" (Wireless network) [IBR 94]. 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 (voir figure 1-1).

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 sévèrement le volume des informations échangées [DNR 92].

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. L'autonomie réduite de sa source d'énergie, lui occasionne de fréquentes déconnexions du réseau ; sa reconnexion peut alors se faire dans un environnement nouveau voire dans une nouvelle localisation.

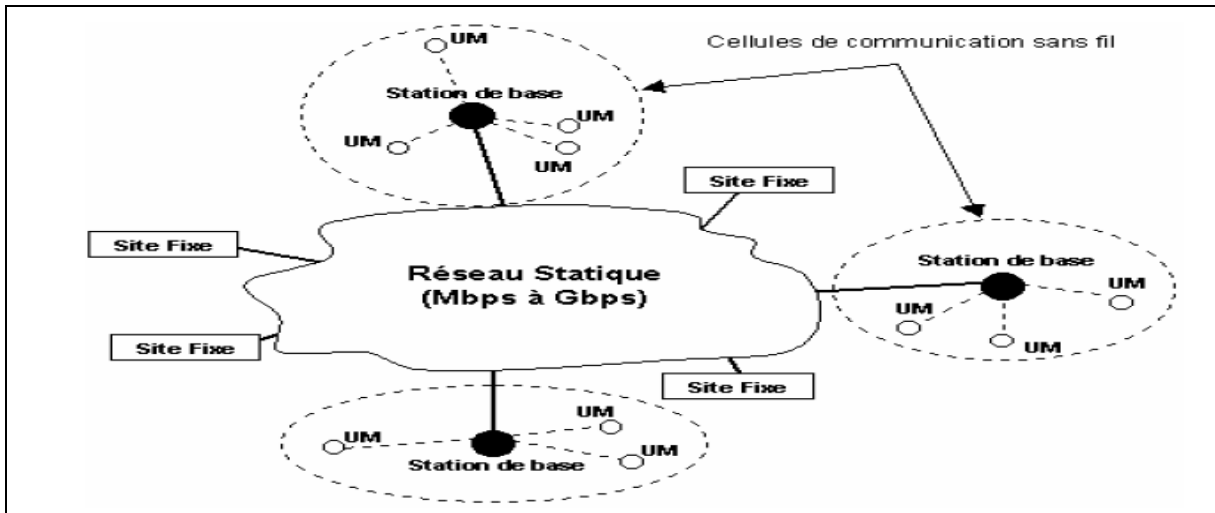


Figure I-1 : Le modèle des réseaux mobiles avec infrastructure.

1.2 Les réseaux sans infrastructure

Le modèle de réseau sans infrastructure préexistante ne comporte pas l'entité "site fixe", tous les sites du réseau sont mobiles et se communiquent d'une manière directe en utilisant leurs interfaces de communication sans fil (voir figure 1-2). 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.[NBA 98]

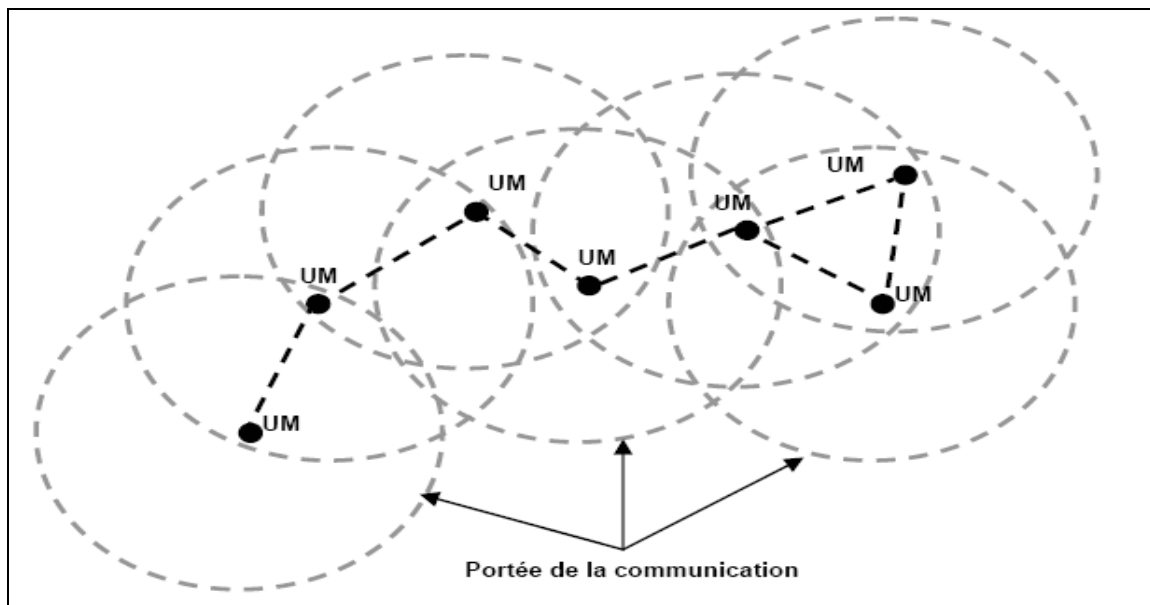


Figure I-2 : Le modèle des réseaux mobiles sans infrastructure.

2. L'utilisation des ondes radio dans la communication sans fil

La transmission radio utilisée dans la communication sans fil des unités mobiles, est basée sur le principe que l'accélération d'un électron crée un champ électromagnétique qui à son tour accélère d'autres électrons et ainsi de suite. Il est alors possible de provoquer le déplacement électromagnétique. Plus le nombre d'électrons déplacés est important, plus le signal est fort et plus sera grande sa portée, avec une vitesse proche de celle de la lumière [WYP 93].

Un déplacement coordonné d'électrons peut alors servir pour le transfert d'information et constitue la base de la communication sans fil. L'approche standard de la transmission radio est le déplacement des électrons à une fréquence donnée. Des techniques de modulation et de multiplexage permettent d'adapter les signaux transmis à la bande passante du support de communication et de rentabiliser son utilisation.

Deux signaux sur la même fréquence interfèrent et s'altèrent mutuellement. Pour y remédier le spectre de fréquence est divisé en plusieurs parties (bandes de fréquence), chaque partie est dédiée à une utilisation spécifique. La taille limitée du spectre de fréquence impose donc le regroupement d'utilisateurs dans des bandes étroites comme dans le cas de la radio cellulaire. Par exemple, la bande de 25 Mhz à 890 Mhz est réservée aux émissions de télévision et la bande supérieure à 890 Mhz pour la téléphonie cellulaire et la transmission par satellite. Au lieu d'allouer à chaque appel la totalité de la fréquence, la technologie cellulaire limite la puissance du signal au minimum nécessaire ; ce qui réduit les limites des interférences à une région de taille réduite autour de la station d'émission. Deux stations d'émission / réception situées dans des régions différentes suffisamment éloignées les unes des autres, peuvent utiliser la même fréquence sans risque d'interférence.

3. La fiabilité de la communication sans fil

La communication sans fil est moins fiable que la communication dans les réseaux filaires. La propagation du signal subit des perturbations (erreurs de transfert, micro- coupure, timeout) dues à l'environnement, qui altèrent l'information transférée. Il s'ensuit alors, un accroissement du délai de transit de messages à cause de l'augmentation du nombre de retransmissions. La connexion peut aussi être rompue ou altérée par la mobilité des sites.

Un usager peut sortir de la zone de réception ou entrer dans une zone de haute interférence. Le nombre d'unités mobiles dans une même cellule (dans le cas des réseaux cellulaires), par exemple lors d'un rassemblement populaire, peut entraîner une surcharge du réseau.

L'une aussi des limites de la communication sans fil vient de la relative faiblesse de la bande passante des technologies utilisées. On distingue les réseaux utilisant l'infrarouge avec un débit de 1Mbps, la communication radio avec 2Mbps et le téléphone cellulaire avec 9 à 14 Kbps. La bande passante est évidemment partagée entre les utilisateurs d'une même cellule. Pour augmenter la capacité de service d'un réseau, deux techniques sont utilisées : la technique de recouvrement des cellules sur différentes longueurs d'ondes et celle qui réduit la portée du signal pour avoir plus de cellules mais de rayon moindre couvrant une région donnée. Chaque cellule est généralement subdivisée en sept cellules dont le rayon r est égal au tiers de celui de la cellule de départ. Deux cellules peuvent utiliser la même fréquence f_i , si la distance d qui les sépare est au moins égale à trois fois le rayon r de la cellule (Figure 1-3). Cette dernière technique est généralement plus utilisée à cause de sa faible consommation d'énergie et une meilleure qualité du signal.[TLE 00]

4. La communication cellulaire

La configuration standard d'un système de communication cellulaire est un maillage (grid) de cellules hexagonales [SGH 95]. Initialement, une région peut être couverte uniquement par une seule cellule. Quand la compétition devient importante pour l'allocation des canaux, la cellule est généralement divisée en sept cellules plus petites, dont le rayon est égal à un tiers du rayon de la cellule de départ. Cette subdivision peut être répétée et l'on parle alors de systèmes micro-cellulaires. Les cellules adjacentes dans le maillage doivent utiliser des fréquences différentes, contrairement à celles qui sont situées sur les côtés opposés du maillage et qui peuvent utiliser la même fréquence sans risque d'interférence. La subdivision des cellules.

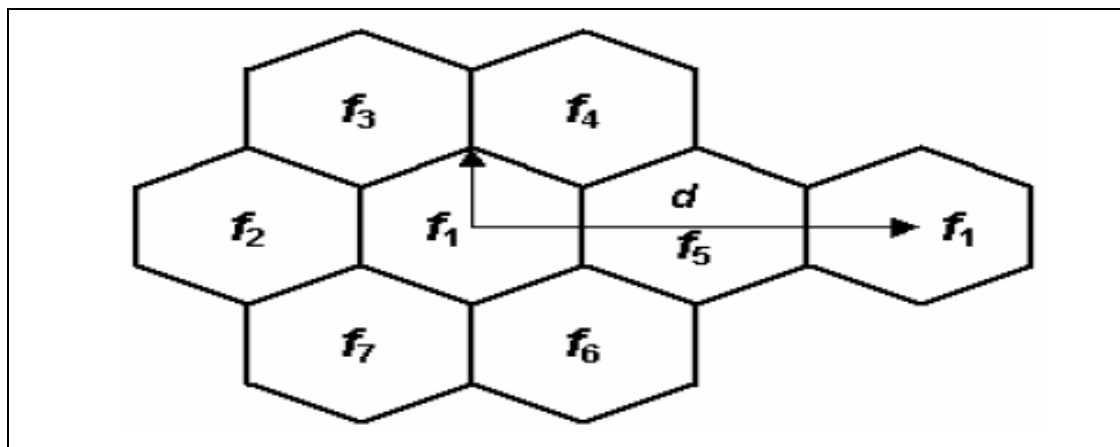


Figure I-3 : Le principe de réutilisation de fréquence.

5. Quelques éléments de l'infrastructure sans fil

Les réseaux informationnels de demain dits PCN (Personale Communication Network) intégreront une large variété de services (voix, données, multimédia ... etc.) offerts aux usagers indépendamment de leur position géographique. L'architecture générale de ces réseaux, bien qu'encore en débat, sera construite autour des infrastructures déjà existantes telles que :

Les réseaux téléphoniques cellulaires (à l'avenir micro-cellulaires) reliés au réseau téléphonique public. Les réseaux locaux traditionnels tels Ethernet, étendus à la communication sans fil, et reliés à des réseaux plus étendus de type LAN, WAN, Internet,...etc. Les architectures orientées vers des services spécialisés fournis par diffusion sur des portions d'ondes radio en modulation de fréquence ou par des satellites à des usagers munis de terminaux spéciaux.

[PEB 93],[ITB 94].

La même unité mobile peut, en principe, interagir avec les trois types d'infrastructures à différents moments, par exemple, en se déplaçant de l'intérieur d'un bâtiment où elle interagit avec un réseau local pourvu d'une interface de Communication sans fil, à l'extérieur du bâtiment où elle interagit avec le réseau téléphonique cellulaire.

III. Les réseaux mobile ad hoc

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 cellulaire, aucune administration centralisée n'est disponible, ce sont les hôtes mobiles elles-mêmes qui forment, d'une manière ad hoc, 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.

Les réseaux ad hoc sont idéals pour les applications caractérisées par une absence (ou la non-fiabilité) d'une infrastructure préexistante, tel que les applications militaires et les autres applications de tactique comme les opérations de secours (incendies, tremblement de terre..) et les missions d'exploration . [TLE 00]

1. Définition

Un réseau mobile ad hoc, appelé généralement MANET (Mobile Ad hoc NETwork), consiste en une grande population, relativement dense, d'unités mobiles qui se déplacent dans un territoire quelconque et dont le seul moyen de communication est l'utilisation des interfaces sans fil, sans l'aide d'une infrastructure préexistante ou administration centralisée. Un réseau ad hoc peut être modéliser par un graphe $G_t = (V_t, E_t)$ où V_t représente l'ensemble des nœuds (i.e. les unités ou les hôtes mobiles) du réseau et E_t modélise l'ensemble des connections qui existent entre ces nœuds (voir figure 1-4).

Si $e = (u, v) \in E_t$, cela veut dire que les nœuds u et v sont en mesure de communiquer directement à l'instant t . [TLE 00]

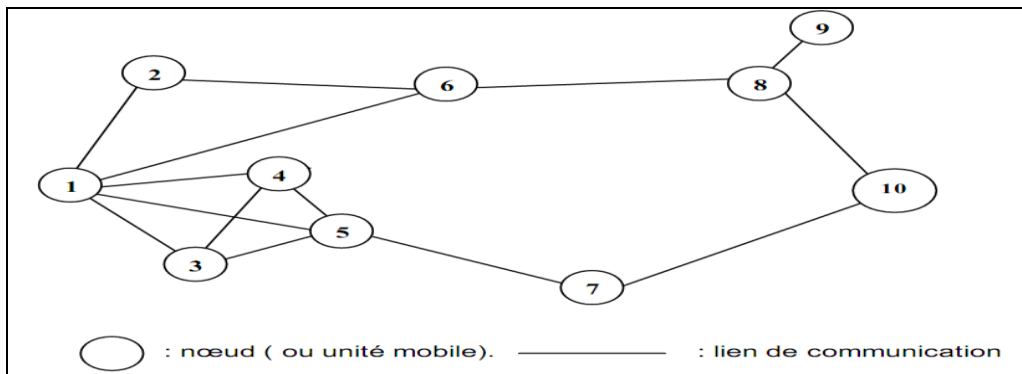


Figure I-4 : La modélisation d'un réseau ad hoc.

La topologie du réseau peut changer à tout moment (voir figure 1-5).Elle est donc dynamique et imprévisible ce qui fait que la déconnexion des unités soit très fréquente [TLE 00].

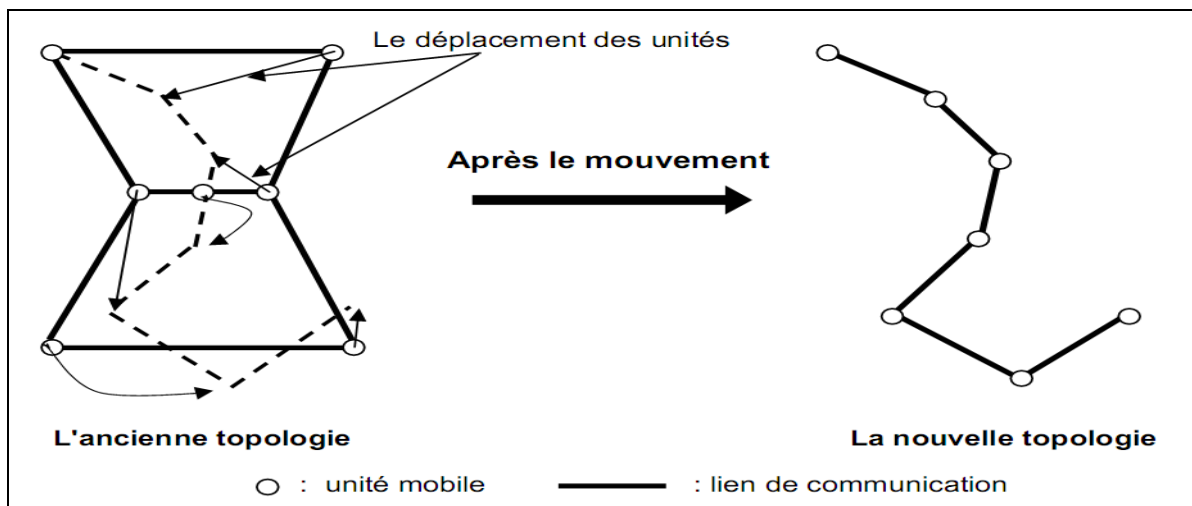


Figure I-5 : Le changement de la topologie des réseaux ad hoc.

Un exemple d'un réseau ad hoc : un groupe d'unités portables reliées par des cartes HIPERLAN. Les réseaux appelés GSM (Sont de réseau cellulaire) ne représentent pas des réseaux ad hoc, car la communication entre les unités passe obligatoirement par des stations de base du réseau filaire.

2. Les applications des réseaux mobiles ad hoc

Les applications ayant recours aux réseaux ad hoc couvrent un très large spectre, incluant les applications militaires et de tactique, les bases de données parallèles, l'enseignement à distance, les systèmes de fichiers répartis, la simulation distribuée interactive et plus simplement les applications de calcul distribué. d'une façon générale, les réseaux ad hoc sont utilisés dans

toute application où le déploiement d'une infrastructure réseau filaire est trop contraignant, soit parce que difficile à mettre en place, soit parce que la durée d'installation du réseau ne justifie pas de câblage à demeure.[TLE 00]

3. Les caractéristiques des réseaux ad hoc

Les réseaux mobiles ad hoc sont caractérisés par ce qui suit :[TLE 00]

3.1 Une topologie dynamique :

Les unités mobiles du réseau, se déplacent d'une façon libre et arbitraire. Par conséquent la topologie du réseau peut changer, à des instants imprévisibles, d'une manière rapide et aléatoire. Les liens de la topologie peuvent être unis ou bidirectionnels.

3.2 Une bande passante limitée :

Un 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.

3.3 Des contraintes d'énergie :

Les hôtes mobiles sont alimentés par des sources d'énergie autonomes comme les batteries ou les autres sources consommables. Le paramètre d'énergie doit être pris en considération dans tout contrôle fait par le système.

3.4 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.

IV. Le protocole de routage de réseaux ad hoc

Suivant la manière de création et de maintenance de routes lors de l'acheminement des données, les protocoles de routage peuvent être séparés en deux catégories, les protocoles pro-actifs et les protocoles réactifs. Les protocoles pro-actifs établissent les routes à l'avance en se basant sur l'échange périodique des tables de routage, alors que les protocoles réactifs cherchent les routes à la demande.

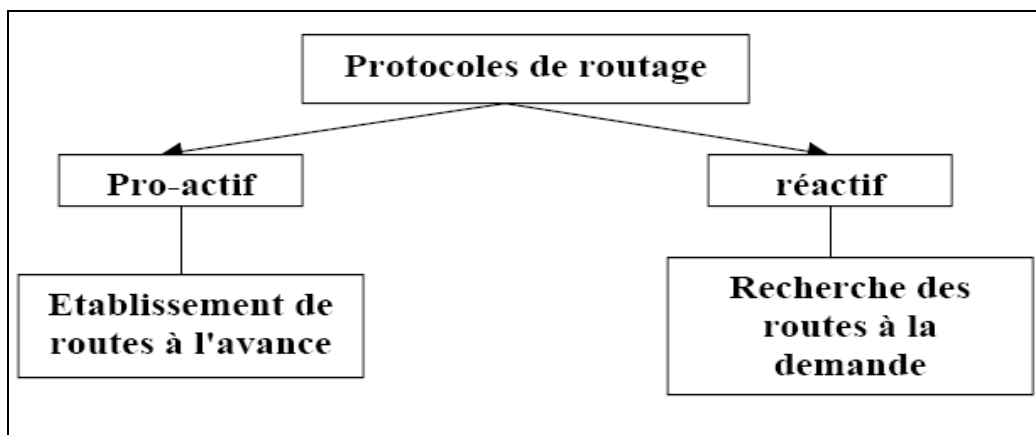


Figure I-6 : Classification des protocoles

1. Les protocoles de routage pro-actifs

1.1 Présentation

Les protocoles de routage pro-actifs pour les réseaux mobiles ad-hoc, sont basés sur la même philosophie que les protocoles de routage utilisés dans les réseaux filaires conventionnels. avant de présenter quelques protocoles de cette classe. Les deux principales méthodes utilisées dans le routage des réseaux filaires sont : la méthode Etat de Lien (« Link State ») et la méthode du Vecteur de Distance (« Distance Vector »).[JTL 99]

1.2 Le protocole de routage DSDV

L'algorithme de routage de Perkins appelé "Vecteur de Distance à Destination Dynamique Séquencée" ou **DSDV** (Dynamic Destination-Sequenced Distance- Vector) a été conçu spécialement pour les réseaux mobiles. Il est basé sur l'idée classique de l'algorithme distribué de Bellman-Ford en ajoutant 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 : séquence numéro) qui correspond à un nœud destination.

Pour chaque nœud i , le numéro de séquence (NS) de la destination j est associé à chaque entrée de distance Dijk, pour chaque voisin k . Le NS est utilisé pour faire la distinction entre les anciennes et les nouvelles routes, ce qui évite la formation des boucles de routage.

[CPB 94]

1.3 Le protocole de routage WRP

Le protocole de routage sans fil **WRP** (Wireless Routing Protocol) est basé sur l'utilisation de la classe des algorithmes de recherche de chemins **PFA** (Path-FindingAlgorithm). Beaucoup d'algorithmes PFA existent dans la littérature : ils utilisent des données concernant la longueur et le nœud prédécesseur du chemin le plus court correspondant à chaque destination, ceci afin d'éviter le problème de "counting to infinity" du DBF. Le problème des PFAs est la présence des boucles de routage temporaires dans le chemin spécifié par le prédécesseur, avant qu'ils convergent.[SMG 95]

1.4 Le protocole de routage GSR

Le protocole appelé "Routage à Etat Global" ou GSR (Global State Routing), est similaire au protocole DSDV décrit précédemment. Ce protocole utilise les idées du routage basé sur l'état des liens (LS), et les améliore en évitant le mécanisme inefficace de l'inondation des messages de routage. Le GSR utilise une vue globale de la topologie du réseau, comme c'est le cas dans les protocoles LS.[SMG 95]

1.5 Le protocole de routage FSR

Le protocole "Routage à Etat de l'œil du Poisson" FSR, (Fisheye State Routing) peut être vu comme une amélioration du protocole GSR précédent. Le nombre élevé des messages de mise à jour échangés implique une grande consommation de la bande passante, ce qui a un effet négatif dans les réseaux ad hoc caractérisés par une bande passante limitée. Le protocole FSR est basé sur l'utilisation de la technique "œil de poisson" (fisheye), proposée par Kleinrock et Stevens, qui l'ont utilisé dans le but de réduire le volume d'information nécessaire pour représenter les données graphiques.[ICG 99]

1.6 Le protocole de routage HSR

Les notions de partitionnement et de groupes sont très répandues dans les réseaux mobiles ad hoc. La formation de groupes peut considérablement améliorer les performances des réseaux. Par exemple, les interférences des signaux peuvent être réduites en utilisant différents codes étendus à l'aide des groupes. Le partitionnement, quant à lui, peut être exploité dans les réseaux de grande taille afin de réaliser un routage hiérarchique, ce qui réduit le contrôle des données de routage. Le problème principal du routage hiérarchique dans les réseaux sans fil est la mobilité et la gestion de la localisation. [GTS 95]

1.7 Le protocole de routage ZHLS

Le protocole "Routage à Etat de Liens Hiérarchique basé sur les Zones", appelé ZHLS (Zone-Based Hierarchical Link State Routing), est basé sur la décomposition du réseau en un ensemble de zones. Dans ce protocole, les membres d'une zone n'élisent pas de représentants, contrairement à ce qui se fait dans les autres protocoles hiérarchiques. Avec cette décomposition, on a deux niveaux de topologies : le niveau nœud, et le niveau zone. La topologie basée sur le premier niveau donne la façon selon laquelle les nœuds d'une zone donnée sont connectés physiquement. [JTL 99]

1.8 Le protocole de routage CGSR

Le protocole appelé CGST (Clusterhead Gateway Switch Routing) utilise principalement l'algorithme de routage DSDV : l'ensemble des unités mobiles du réseau est décomposé en groupes, et chaque groupe élit un représentant. Les nœuds appartenant à la portée de communication d'un représentant de groupe appartiennent au groupe représenté par ce dernier. [CWM 97]

1.9 Le protocole de routage DREAM

Le protocole appelé "Algorithme d'Effet de Routage basé sur la Distance pour la Mobilité" ou DREAM (Distance Routing Effect Algorithm for Mobility) est un protocole pro-actif basé sur les informations de localisation des unités mobiles. [BSW 98]

2. Les protocoles de routage réactifs (à la demande)

2.1 Présentation

Les protocoles de routage réactifs (aussi appelés protocoles de routage à la demande) sont les protocoles les plus récents proposés dans le but d'assurer le service du routage dans les réseaux sans fil. La majorité des solutions proposées pour résoudre le problème de routage dans les réseaux ad hoc, et qui sont évaluées actuellement par le groupe de travail MANET (Mobile Ad Hoc Networking Working Groupe) de l'IETF (Internet Engineering Task Force), appartiennent à cette classe de protocoles de routage.[VPC 99]

2.2 Le protocole de routage CBRP

Dans le "Protocole de Routage Basé sur les Groupes" appelé CBRP (Cluster Based Routing Protocol), l'ensemble des nœuds du réseau est décomposé en groupes. Le principe de formation des groupes est le suivant : Un nœud p qui n'a pas de statut (i.e. qui n'est ni membre ni représentant de groupe), active un timer et diffuse un message "Hello".

Lorsqu'un représentant de groupe reçoit ce message, il envoie immédiatement une réponse à l'émetteur. Lors de la réception de réponse, le nœud p change son état « indécié » à l'état "membre". Si p dépasse un certain timeout en attendant la réponse et dans le cas où il possède un lien bidirectionnel vers au moins un nœud voisin, il se considère lui-même comme un représentant de groupe. Dans le cas contraire, p reste dans l'état in décié et il répète la même procédure. A cause des changements rapides de la topologie des réseaux ad hoc, l'attente des nœuds « indéciés » est très courte.[MJT 99]

2.3 Le protocole de routage DSR

Le protocole "Routage à Source Dynamique" (DSR : Dynamic Source Routing) est basé sur l'utilisation de la technique "routage source". Dans cette technique, la source des données détermine la séquence complète des nœuds à travers lesquels les paquets de données seront envoyés. Pour envoyer un paquet de donnée à un autre nœud, l'émetteur construit une route source et l'inclut en tête du paquet. La construction se fait en spécifiant l'adresse de chaque nœud à travers lequel le paquet va passer pour atteindre la destination.[CPR 00]

2.4 Le protocole de routage AODV

Le protocole "Routage avec Vecteur de Distance à la Demande" (**AODV** : Ad hoc On demand Distance Vector), Il réduit le nombre de diffusions de messages en créant les routes au besoin, contrairement au **DSDV** qui maintient la totalité des routes. L'AODV est basé sur l'utilisation des deux mécanismes "Découverte de route" et "Maintenance de route" (utilisés par le **DSR**), en plus du routage nœud-par-nœud, du principe des numéros de séquence et de l'échange périodique du **DSDV**. [**CPR 00**]

2.5 Le protocole de routage TORA

L'Algorithme de Routage Ordonné Temporairement ou TORA (Temporary Ordering Routing Algorithme) a été conçu principalement pour minimiser l'effet des changements de la topologie.[**VDS 99**]

2.6 Le protocole de routage ABR

Le protocole "Routage Basé sur l'Associativité" (ABR : Associativity Based Routing) représente une nouvelle approche de routage pour les réseaux mobiles ad hoc. Le protocole définit une nouvelle métrique de routage appelée degré de stabilité d'association, et évite les problèmes de formation des boucles de routage, ainsi que ceux de blocage et de duplication de paquets. Dans l'ABR, le choix des routes est basé sur ce qu'on appelle les états d'associativité des nœuds.[**CKT 96**]

2.7 Le protocole de routage SSR

Le protocole "Routage basé sur la Stabilité du Signal" (SSR : Signal Stability-based Routing), est un protocole de routage réactif dont le choix des routes est basé sur la puissance du signal entre les nœuds, en plus de leur stabilité de localisation. Ce critère de sélection de routes fait que les chemins utilisés durant le routage des données ont une forte interconnexion.

[**RDU 97**]

3. Avantages et inconvénients des réseaux mobiles Ad hoc :

Les réseaux mobile Ad Hoc , appelés aussi MANET pour ad hoc Network , utilisation aucune connexion filaire n'est disponible , par exemple lors d'une opération militaire , et plus généralement quand le déploiement rapide d'un réseaux est nécessaire , dans ce cas , les nœuds communiquent en acheminant les message par routage "multi-saut" Indépendamment du fait de disposer ou non d'un infrastructure . Le mode Ad hoc multi-saut a de nombreux avantages en comparaison avec le mode de communication avec stations de base.[**YMZ 11**]

3.1 Avantage :

✓ **Pas de câblage :**

L'une des caractéristiques des réseaux Ad Hoc est l'absence d'uncâblage. et ce en éliminant toutes les connexions filaires qui sont remplacées par des connexions radio

✓ **Déploiement facile :**

L'absence du câblage donne plus de souplesse, et permet de déployer un réseau Ad hoc facilement et rapidement .cette facilité peut être justifiée par l'absence d'une infrastructure préexistante permettant ; ainsi, d'économiser tout le temps de déploiement et d'installation du matériel nécessaire.

✓ **Consommation énergétique :**

Un mobile émet plus de messages en mode Ad Hoc qu'en mode infrastructure, puisqu'il doit à la fois transmettre ses propres paquets mais également les paquets des autres mobiles pour lesquels il fait office de routeur. On pourrait donc penser que ce mode est plus gourmand en énergie. Néanmoins, les portées de communication peuvent être largement réduites en mode Ad Hoc. Cette diminution de portée de communication permet d'économiser beaucoup d'énergie, puisque la consommation énergétique varie au moins proportionnellement au carré de la distance de communication. Ainsi, même si l'on transmet dix fois plus de messages en mode Ad Hoc, diviser par dix la portée de transmission permet de ne consommer qu'un dixième de l'énergie nécessaire au mode station de base.

✓ **Permet la mobilité :**

Comme l'indique leur nom, et à l'image des réseaux sans fil avec infrastructure les réseaux mobile ad hoc permettent une certaine mobilité à leur nœuds, De ce fait, ces derniers peuvent se déplacer librement à condition de ne pas s'éloigner trop les uns des autres pour grader leur connectivité.

✓ **Extensible :**

L'une des propriétés les plus importantes d'un réseau Ad Hoc est la possibilité de l'étendre, et d'augmenter sa taille très facilement et sans nécessiter trop de moyens. Pour expliquer cet aspect, il suffit uniquement d'imaginer l'arrivée d'un nouveau nœud mobile à un réseau Ad Hoc déjà installé et mis en place. Pour que ce nœud

fonctionne au sein du réseau, il suffit de procéder à quelques configurations au niveau du nœud lui-même.

✓ **la souplesse d'utilisation :**

est un paramètre très important puisque les seuls éléments pouvant tomber en panne sont les terminaux eux-mêmes. Autrement dit, il n'y a pas de panne "pénalisante" de manière globale (une station qui sert au routage peut être remplacée par une autre si elle tombe en panne). [TAB 12]

✓ **Coût :**

Le déploiement d'un réseau Ad Hoc ne nécessite pas d'installer des stations de base, les mobiles sont les seules entités physiques nécessaires pour déployer un tel réseau. Ce qui conduit à la réduction de son coût d'une manière significative.

3.2 Inconvénients :

✓ **Topologie non prédictible :**

L'activité permanente et les déplacements fréquents des nœuds d'un réseau Ad Hoc rendent son étude très difficile. La raison est bien connue, le changement rapide de sa topologie dû aux déplacements des nœuds.

✓ **Capacités limitées (puissance de calcul, mémoire, énergie) :**

Dans un tel réseau, la configuration de la portée de communication des nœuds (ce qui revient à paramétrer la puissance d'émission) est importante. En effet, il faut qu'elle soit suffisante pour assurer la connectivité du réseau. Mais plus on accroît la portée des mobiles, plus les communications demandent de l'énergie. Il faut donc trouver un compromis entre la connectivité du réseau et la consommation énergétique.[YMZ 11]

✓ **Taux d'erreur important :**

Les risques de collisions augmentent avec le nombre de nœuds qui partagent le même médium. Par conséquent, plus la portée augmente, plus le risque de collisions n'est important.

✓ **Sécurité :**

Un autre dilemme des réseaux Ad Hoc, et qui attire la curiosité des chercheurs et des spécialistes de ce domaine est la notion de sécurité. Un réseau Ad Hoc tel que

défini précédemment ne permet pas d'assurer la confidentialité de l'information échangée entre les nœuds. Contrairement aux réseaux filaires, les réseaux sans fil sans infrastructure ne peuvent utiliser un matériel spécifique (firewale par exemple) pour empêcher les accès non autorisés au réseau.

V. Conclusion

Dans ce chapitre nous avons vu que les réseaux informatiques basés sur la communication sans fil peuvent être classés en deux catégories : les réseaux avec infrastructure fixe préexistante, et les réseaux sans infrastructure. Dans la première catégorie, le modèle de la communication utilisé est généralement le modèle de la communication cellulaire. Dans ce modèle les unités mobiles sont couvertes par un ensemble de stations de base reliées par un réseau filaire, et qui assurent la connectivité du système. La deuxième catégorie essaie 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. Les réseaux de cette catégorie sont appelés : les réseaux ad hoc.

Les réseaux ad hoc doivent s'organiser automatiquement de façon à être déployables 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 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. Chaque nœud participe donc à une stratégie ou protocole de routage qui lui permet de découvrir les chemins existants, afin d'atteindre les autres nœuds du réseau.

Capiter II. Présentation des deux simulateurs« NS-2 » et « GloMoSim »

Chapitre « 2 » : Présentation des deux
simulateurs de deux « NS-2 » & GloMoSim »

I. Introduction

Pour tester les performances d'une solution apportée à un problème de communication dans un réseau, il n'est pas toujours possible d'accéder aux infrastructures nécessaires en raison de leurs coûts élevés. De plus, les expérimentations réelles n'offre souvent pas une grande souplesse. Rappelons que les réseaux ad hoc sont des réseaux qui englobent plusieurs unités mobiles qui se déplacent dans un territoire quelconque et dont le seul moyen de communication est l'utilisation des interfaces radio. En effet, [KAY 12] il serait très coûteux voire impossible de mettre en place un réseau à des fins de tests de certains critères. Pour remédier à ce problème et afin de tester les performances d'un nouveau protocole, on a recours à la simulation qui met à la disposition de l'utilisateur un environnement d'expérimentation.

Qu'est-ce que la simulation ?

Simuler, c'est modéliser un système complexe, afin de prévoir son comportement dans le monde réel. Il s'agit d'une approche permettant de représenter le fonctionnement d'un système réel constitué de plusieurs entités, de modéliser les différentes interactions entre elles, et enfin évaluer le comportement global du système et son évolution dans le temps. Le recours à la simulation permet de contourner les limites de la complexité des modèles analytiques. Toutefois, il est nécessaire de bien identifier les caractéristiques du système afin de le représenter, le plus finement possible, par des modèles abstraits.[HYZ 03]

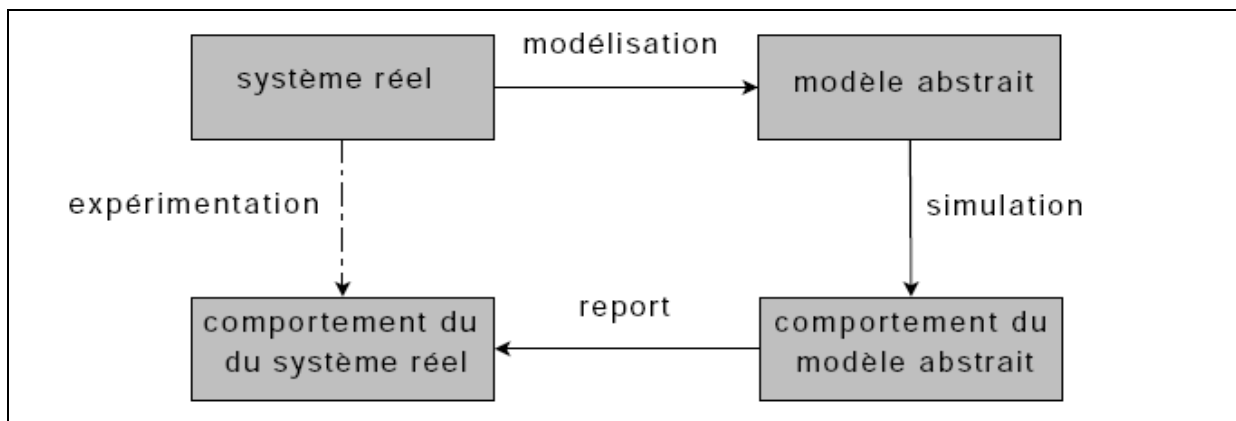


Figure II-1 : Cycle modélisation-simulation.

La simulation d'un système réel devient nécessaire dès lors que les modèles analytiques deviennent, soit trop complexes en termes de calcul et de temps de résolution, soit trop simplifiés vis-à-vis de la réalité rendant, par ce fait, les résultats obtenus non représentatifs du comportement du système dans un environnement réel. Ainsi, la simulation peut s'avérer nécessaire dans les cas suivants :[HYZ 03]

- Les expériences sur système réel sont trop coûteuses en termes de ressources matérielles et Humaines.
- Les expériences sur système réel ne sont pas reproductibles ni représentatives de tous les environnements possibles. Dans ce cas, la simulation permet de caractériser le comportement global du système pour différents environnements.

II. Environnement de simulation

1. Introduction à la simulation

La simulation connaît de nos jours un essor considérable. Ceci est dû aussi bien à l'intérêt théorique que présente la modélisation des systèmes simulés, que par les besoins croissants de simuler par ordinateur des réalisations de plus en plus complexe.

On constate en effet que les conditions d'expérimentation sont aujourd'hui toujours plus coûteuses, notamment aux techniques dites virtuelles. Celle-ci consiste à représenter la réalité simulée sur ordinateur, à donner à celle-ci des moyens de perceptions réalisés artificiellement, puis à simuler son fonctionnement. C'est sur ce dernier point que la simulation joue un rôle important en mettant à la disposition de l'utilisateur un environnement d'expérimentation dont on peut faire varier les paramètres.

Les conceptions et les techniques de programmation on fait de leur côté d'important progrès, on particulier dans les domaines de l'abstraction et du parallélisme. C'est ainsi que les langages de programmations d'aujourd'hui offrent des moyens performants, ce qui rend possible la réalisation d'un simulateur dans un environnement de programmation existant [ERA 96].

2. Système réel et objectif de simulation

La simulation est une technique de modélisation du monde réel. Elle permet de représenter le fonctionnement d'un système composé de différents centres d'activité, de mettre en évidence les caractéristiques de ceci et les interactions entre eux, de décrire la circulation de différents objets traités par ces processus et en fin d'observer le comportement du système dans son ensemble et dans son évolution dans le temps.

On appelle communément système réel le phénomène que l'on veut observer. On y distingue d'une part les centres d'activités ou tâches d'activités, c-à-d les endroits où il se passe quelque chose ou un objet est créé, traité, transformé ou détruit, d'autre part les objets qui circulent dans le système et qui sont manipulés par différentes tâches actives.

Nous considérons que le but de l'opération est d'étudier un tel système et pour comprendre les relations entre ces composantes ou pour prédire ces performances dans un environnement donné [ERA 96].

3. Modèles de simulation

On distingue plusieurs modèles de simulation selon qui sont statiques ou dynamiques, déterministes ou aléatoires, continues ou discrets.

- Un modèle de simulation est dynamique ou statique selon que les valeurs de ses variables
- Un modèle de simulation est déterministe s'il ne contient aucune variable aléatoire, il est aléatoire dans le cas contraire.

Dans le cas d'un système aléatoire, les résultats de la simulation sont eux même aléatoires et ne donnent qu'une estimation du comportement du système simulé.

Enfin, un modèle de simulation est continu si l'ensemble des instants considéré forme un intervalle compact sur l'axe des temps, il est discret si celui-ci se compose d'un nombre fini ou dénombrable de valeurs isolées.

3.1 Gestion du temps

Une simulation consiste à gérer le temps ainsi que les actions qui sont liés aux différents instants du système réel et à faire fonctionner abstraitement le modèle qui représente ce système.

Dans un modèle discret, le temps est géré par un échéancier et par une horloge centrale ou une horloge du système. L'échéancier est une liste d'événement ordonné chronologiquement selon l'heure à laquelle ils doivent être activés. L'horloge du système donne à chaque instant l'heure du système de simulation.

Elle est réglée au départ sur l'heure d'activation du premier événement. Chaque événement inséré dans l'échéancier en est retiré lorsque son heure d'activation est égale à celle du système de simulation. Des actions associées à cet événement sont alors exécutées, ensuite l'horloge du système est avancée à l'heure d'activation de l'événement suivant.

Dans un modèle continu le temps est discrétisé selon un pas donné et, chaque avance de temps, les valeurs des variables du système sont mises à jour [ERA 96].

3.2 Simulation par événements discrets

La simulation par événements discrets désigne la modélisation d'un système réel tel qu'il évolue dans le temps, par une représentation dans laquelle les grandeurs caractérisant le système (variable) ne changent qu'en un nombre fini de ou dénombrable de points isolés dans le temps. Ces points sont les instants où se passent les événements, c-à-d le phénomène capable de modifier l'état de système et nous appelons comme événement tout changement d'état du système réel se produisant à un instant donné, ainsi que les actions qui accompagnent ou caractérisent ce changement.

La simulation par événement discret consiste alors à prendre en compte dans la modélisation des tâches actives, les seuls instants où un événement se produit et à concentrer l'activité des tâches simulées sur ces instants-là. Nous considérons comme système de simulation par événements discrets tous les systèmes basés sur cette abstraction [ERA 96].

3.3 Simulateur

Nous appelons simulateur un programme qui met en œuvre un modèle de simulation par événements discrets. La tâche première d'un simulateur est d'assurer que la chronologie des événements soit respectée. A chaque occurrence d'un événement, les actions qui sont associées à celui-ci sont exécutées [ERA 96].

Les applications de la simulation sont innombrables. Parmi les domaines dans lesquels elle est le plus utilisée, on s'intéresse à la simulation des réseaux mobiles (réseaux ad hoc).L'un des simulateurs de ce type est « Glomosim » développé en utilisant le langage PARSEC.

4. Avantages et inconvénients de la simulation

Nous trouvons aussi dans la simulation et les avantages et les inconvénients : [BAK 14].

➤ **Avantage :**

- ✓ Observations des états du système.
- ✓ Etudes des points de fonctionnement d'un système.
- ✓ Etudes de l'impact des variables sur les performances du système.
- ✓ Etude d'un système sans les contraintes matérielle

➤ **Inconvénients :**

- ✓ La conception de modèles peut nécessiter des compétences spéciales.
- ✓ Résultats pas forcément généralisable.

III. NS-2 (Network simulateur)

1. Présentation de network simulator NS-2

Le simulateur du réseau **NS-2** est un outil logiciel de simulation de réseaux informatiques. Il est principalement bâti avec les idées de la conception par objets, de réutilisation du code et de modularité.

NS-2 est écrit en C++ et utilise le langage **OTCL** (Object Tools Command Language) dérivé de **TCL**. A travers **OTCL**, l'utilisateur décrit les conditions de la simulation : la topologie du réseau, les caractéristiques des liens physiques, les protocoles utilisés, les communications qui ont lieu. La simulation doit d'abord être saisie sous forme de fichier que NS va utiliser pour produire un fichier contenant les résultats. Mais l'utilisation de l'Otcl permet aussi à l'utilisateur de créer ses propres procédures (par exemple s'il souhaite enregistrer dans un fichier l'évolution d'une variable caractéristique du réseau au cours du temps).

Il contient les fonctionnalités nécessaires à l'étude des algorithmes de routage unicast ou multicast, des protocoles de transport, de réservation, des services intégrés, des protocoles d'application. De plus le simulateur possède déjà une palette de systèmes de transmission, d'ordonnanceurs et de politiques de gestion de files d'attente pour effectuer des études de contrôle de congestion. [PEH 99]

Le Simulateur se compose d'une interface de programmation en TCL et d'un noyau écrit en C++ dans lequel la plupart des protocoles réseaux ont été implémentés :

- **Traffic** parreto, ON/OFF, CBR, FTP, telnet, etc.
- **Couche Transport** TCP, UDP
- **Couche Réseaux** IP, routage dans les réseaux ad-hoc (**aodv**, **dsr**,**dsv**, **tora**, **amodv**), routage dans les réseaux filaire (Link state, Distance vector), les réseaux multicast, IntServ, DiffServ .
- **Couche MAC** CSMA, CDMA, 802,X, Token ring, MPLS, liens satellite, etc.[Site 4]

2. Le langage TCL (Tool Command Language)

TCL (Tool Command Language) Ce langage est à typage dynamique, multi-plateforme et extensible. Il est facile à apprendre et repose sur 11 règles syntaxiques [ELB 07].TCL s'interface très facilement avec le langage C, de plus il est économe en mémoire vive.est un

langage de programmation dont le but est de passer des commandes à des programmes interactifs tels que des éditeurs de texte, des débogueurs et des interpréteurs shell. Il possède une syntaxe simple et il est lui-même programmable : les utilisateurs de Tcl peuvent en effet écrire des procédures pour créer des commandes plus puissantes que celles fournies par l'ensemble préconstruit.

La bibliothèque Tcl est constituée d'un analyseur syntaxique du langage Tcl, de routines implémentant les commandes prédéfinies de Tcl, et de procédures permettant à chaque application d'ajouter à Tcl des commandes additionnelles qui lui sont spécifiques. Le programme applicatif génère des commandes Tcl et les passe à l'analyseur syntaxique de Tcl pour l'exécution. [Site 6]

2. Otc

OTcl est une extension orientée objet de Tcl. Les commandes Tcl sont appelées pour un objet. OTcl est une extension du langage de commande TCL, qui utilise une programmation structurée (boucles, procédures, notions de variables)[ELB 07]. Le moteur OTcl transforme les instructions TCL en instructions C++.. En OTcl, les classes sont également des objets avec des possibilités d'héritage. La définition d'une classe commence par la directive **Class**. Les fonctions et les attributs d'une classe s'installent par la suite par les commandes « instvar » et « instproc ». L'utilisation « instproc » définit les méthodes de la classe de manière assez semblable à C++. Lors d'une instantiation en OTc. [KFV 02]

3. NAM

NAM est un outil de visualisation qui présente deux intérêts principaux : représenter la topologie d'un réseau décrit avec NS-2, et afficher temporellement les résultats d'une trace d'exécution NS-2. Par exemple, il est capable de représenter des paquets TCP ou UDP, la rupture d'un lien entre nœuds, ou encore de représenter les paquets rejetés d'une file d'attente pleine. Ce logiciel est souvent appelé directement depuis les scripts TCL pour NS-2, afin de visualiser directement le résultat de la simulation.[PDA 04]

4. Le modèle de réseau sous ns-2

Un modèle de réseau sous NS est constitué :

- De nœuds de réseau : endroits où est généré le trafic, ou nœuds de routage ;
- De liens de communication entre les réseaux.

- D'agents de communication, représentant les protocoles de niveau transport (TCP, UDP) ; ces agents sont attachés aux nœuds et connectés l'un à l'autre, ce qui représente un échange de données (connexion TCP, flux UDP).
- D'applications qui génèrent le trafic de données selon certaines lois (CBR, VBR), et se servent des agents de transport. [PEH 99]

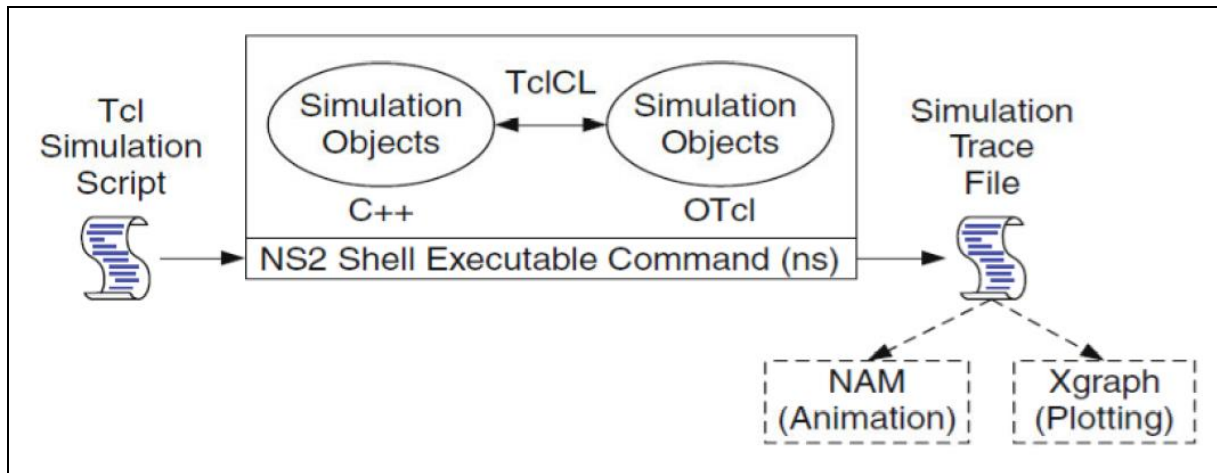


Figure II-2 : L'architecture générale du NS2.

L'architecture générale du NS2 est présentée sur la (figure 2-2), il consiste en deux type de Langage de programmation : le C++ et l'OTcl (Object-orientedTool Command Language). Le C++ est utilisé pour programmer les entités internes des systèmes simules, alors que l'OTcl est utilisé pour définir les scenarios des simulations et les paramètres de configuration. Ces deux types de langages sont, ensuite, lies via le TcICL qui permet le passage des codes C++ vers les codes en OTcl et « viceversa ». [KBO 12]

Une fois la simulation terminée, NS produit deux fichiers de traces qui visualisent la dynamique des systèmes simules, et qui peuvent être interpréter en utilisant les outils: NAM (Network AniMator) et Xgraph.[KBO 12]

3 Caractéristiques d'une entité communicante sous NS-2

Le nœud (entité communicante) constitue l'élément de base de notre modèle. Un nœud dans NS-2 est une classe définie dans OTCL, qui a une adresse et qui contient trois entités : le « classifier », le lien et l'agent. [Site 5]

3.1 Le Classifier

La fonction d'un nœud est d'examiner des champs du paquet reçu, et plus précisément, l'adresse source et l'adresse destination. Selon ces valeurs, le nœud envoie ce paquet sur ses

interfaces de sortie (F). En NS-2, ceci est effectué par un objet qui s'appelle «classifier ». Il existe plusieurs types de classifier qui sont utilisés pour différents buts :

- « adresse classifier » : il est utilisé pour traiter les paquets unicast, son rôle est de sélectionner les paquets adressés directement au nœud, et de choisir le lien vers le prochain nœud ;
- « port classifier » : son rôle est de sélectionner l'agent auquel le paquet est destiné ;
- « multicast classifier » : il est utilisé pour classifier les paquets multicast.

3.2 Le lien

Il est utilisé pour relier les nœuds (voir figure 2-4). Un lien est défini par plusieurs paramètres comme : sa bande passante, le point d'entrée, la durée de vie de chaque paquet, etc. NS2 présente plusieurs types de liens, ainsi on peut distinguer des liens unidirectionnels ou bidirectionnels, des liens filaires et des liens non filaires pour modéliser les réseaux sans fils.

3.3 L'agent

Les agents représentent des points terminaux, là où des paquets de couche réseau sont construits ou consommés. Ces agents constituent le troisième composant du nœud. Dans NS2, le rôle de l'agent est de fournir l'adresse de destination, les fonctions pour générer les paquets et l'interface à la classe application (voir figure 2-4).

Dans NS2 il existe plusieurs types d'agents, chacun a un rôle spécifique :

- agent TCP : pour émettre un trafic TCP ;
- agent UDP : pour émettre un trafic UDP ;
- agent TCPSink : pour la réception du trafic TCP ;
- agent NULL : pour la réception des paquets UDP.

La figure 2-3 représente les entités existantes dans un nœud et les liens entre ses entités.

[KAY 12]

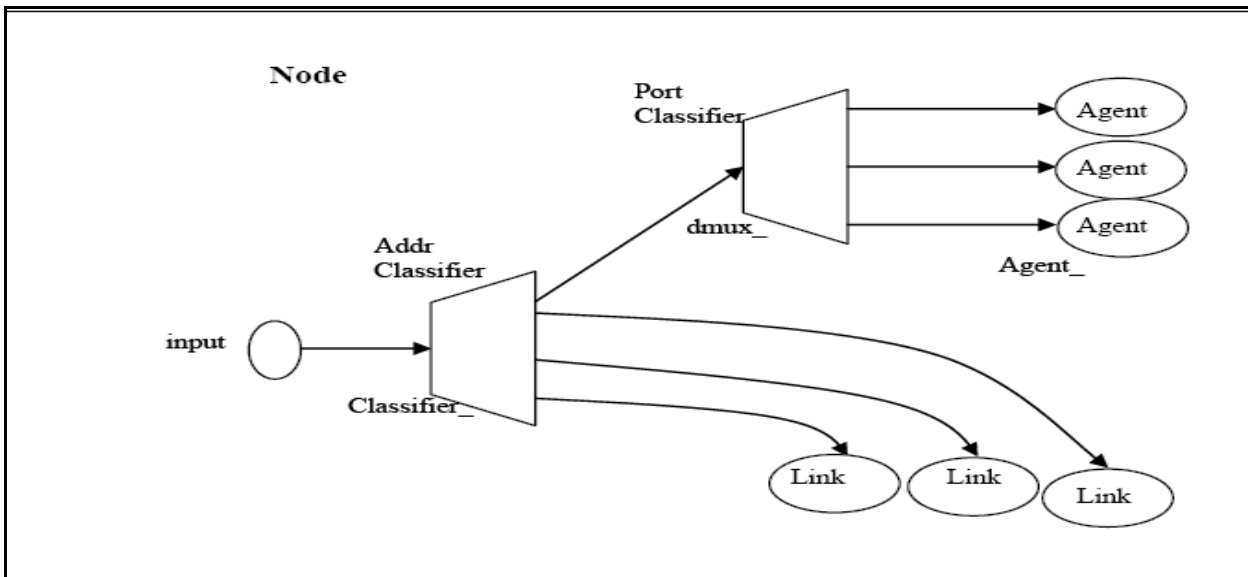


Figure II-3 : Schéma d'un nœud dans NS2

L'analyse des résultats est en général peu aisée, le résultat de la simulation étant essentiellement composé d'un fichier retraçant l'ensemble des envois, réceptions et suppressions de paquets. Un certain nombre de scripts ont été développés (ou sont en cours de développement) pour faciliter cette analyse. [KAY 12]

4 Le processus de simulation

Le processus de simulation en utilisant NS-2 est composé de trois phases principales :

[KAY 12]

- ✓ **Phase de préparation** : s'occupe de la génération des fichiers d'entrées. A cette étape, on introduit des fichiers de scripts **Otcl** qui décrivent l'environnement avec tous ses nœuds, leurs déplacements et leur trafic de données. Ces fichiers sont classés en deux catégories :
 - Fichiers de scénario qui décrivent les nœuds, leurs positions ainsi que leurs mouvements.
 - Fichiers de communication qui décrivent le trafic dans le réseau.
- ✓ **Phase de simulation** : pour lancer les simulations et générer les traces. Les deux fichiers obtenus de la phase de préparation sont introduits dans un script de lancement OTcl. Le script de simulation consiste à indiquer la topologie du réseau, à activer des traces aux endroits pertinents, à engendrer des événements particuliers à des instants donnés. A la fin de cette étape on obtient deux fichiers (journaux) appelé aussi «

fichiers traces ». Le premier fichier sera traité par l'outil de visualisation NAM. Et le deuxième doit être filtré par un script awk afin d'afficher le résultat en utilisant l'outil Xgraph.

- ✓ **Phase d'analyse :** pour analyser les traces et générer les courbes. L'outil de visualisation NAM s'occupe du premier fichier trace. Deux éléments intéressants sont proposés à la visualisation : un dessin de la topologie du réseau étudié, et une visualisation dynamique du déroulement du programme dans le temps. Le deuxième fichier de trace sauvegarde tous les échanges de paquets effectués. Afin de dessiner les courbes en utilisant Xgraph, le fichier doit être filtré par un script awk pour ne garder que les informations pertinentes.

Le schéma des étapes de simulation est illustré dans la figure suivante : [KAY 12]

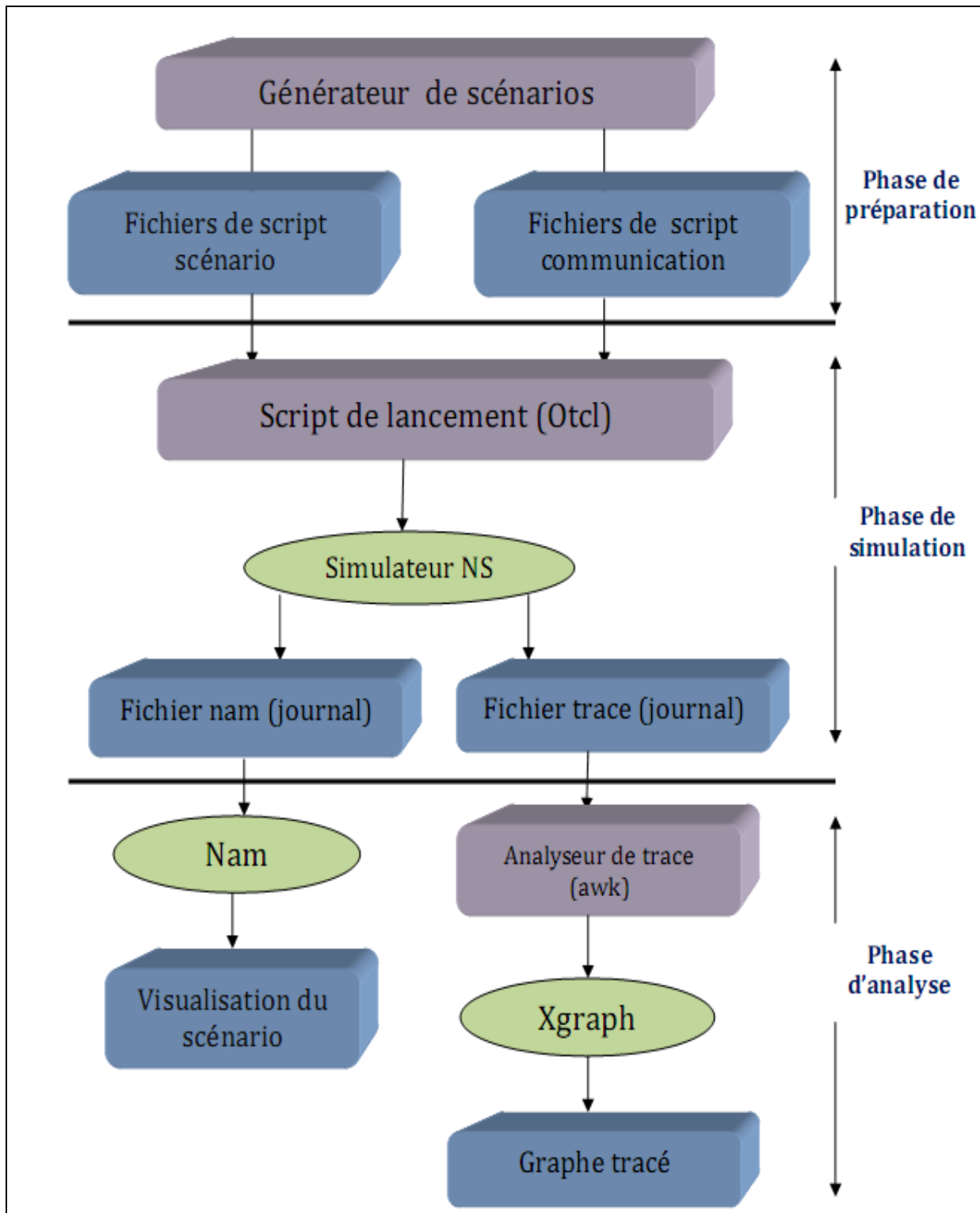


Figure II-4 : Le processus de simulation NS-2

5. Création d'un scénario

Pour décrire un réseau et son trafic, il faut définir dans l'ordre : [TAL 09]

1. la topologie du réseau : les nœuds et les arcs
2. la couche transport (UDP, TCP, ...) entre des paires de nœuds
3. la couche application qui va fournir les données
4. des temporisateurs précisant les instants auxquels les transferts vont démarrer

- ✓ Dans un fichier otcl, on commence par créer un simulateur

```
set ns [new Simulator]
```

- ✓ Pour conserver l'historique de la simulation (notamment pour la visualiser avec NAM), on peut indiquer les deux fichiers vers lesquels seront dirigés les deux historiques :

```
set tracefd [open simulation_tras.tr w]
set namtrace [open simulation_tras.nam w]
```

- ✓ **Topologie** : Ensuite, on crée les nœuds du réseau, puis les liens qui les relie avec leur débit et leur latence, et la politique de rejet des paquets en trop (ici **DropTail** mais il y en a d'autres). On peut aussi préciser le nombre limite de paquets dans la file d'attente.

```
set n1 [$ns node]
set n2 [$ns node]
$ns duplex-link $n1 $n2 1Mb 50ms DropTail
$ns queue-limit $n1 $n2 100
```

- ✓ Les nœuds de réglage de position et de mouvement

```
# Provide initial location of mobilenodes
$node_0 set X_ 50.0
$node_0 set Y_ 200.0
$node_0 set Z_ 0.0

$node_1 set X_ 100.0
$node_1 set Y_ 200.0
$node_1 set Z_ 0.0

# Generation of movements node_1 . vitesse Node(1) 1m/s
$ns at 1 "$node_1 setdest 400.0 200.0 1"
```

Au moment de 1,0 secondes, le nœud commence à se déplacer à partir de son emplacement actuel vers Nœud1 (50, 200, 0.0) à la vitesse de 1m / s. et Nœud 2 (100, 200, 0.0) fixé

- ✓ **Transport** : Il faut créer les agents de transport, les attacher aux nœuds et les connecter entre eux. Parmi les agents disponibles, on trouve **Agent/Null**, **Agent/UDP**, **Agent/TCP** et **Agent/TCPSink**. Le code est par exemple :

```
set udp [new Agent/UDP] # crée une source UDP
set trappe [new Agent/Null] # crée un puits d'information
$ns attach-agent $n1 $udp # attache la source à un noeud
$ns attach-agent $n2 $trappe # attache la destination à un autre
$ns connect $udp $trappe # associe la paire origine/destination
```

- ✓ Il faut enfin attacher l'application à l'agent de transport :

```
set app [new Application/Traffic/CBR]
$app attach-agent $udp
```

- ✓ Pour démarrer et arrêter une source *src*, on a :

```
$src start et $src stop
```

- ✓ **Temporisateurs** : Il est possible de programmer des événements qui seront exécutés à des dates précises de la simulation. Par exemple :

```
$ns at 1.0 "$truc start" # démarre l'application $truc à t=1.0
$ns at 5.0 "$truc stop" # l'arrête à t=5.0
$ns at 10.0 "finish" # invoque la procédure finish à t=10.0
```

- ✓ Typiquement, la fin de la simulation se fera par l'invocation d'une procédure chargée du traitement final des statistiques. Exemple :

```
proc finish { } {
global ns f nf stat
puts "C'est fini"
close $f # fermeture du fichier de trace (s'il y a)
close $nf # fermeture du fichier de trace animation
puts "Débit = $stat(debit)" # impression d'une statistique
```

5. L'avantage et inconvénients du simulateur NS-2

L'avantage et inconvénients du simulateur NS-2 : [KAY 12]

5.1 Avantage du simulateur NS-2 :

- ✓ Un logiciel de simulation multicouche.
- ✓ Un outil complètement libre pour plusieurs plateformes.

- ✓ Possibilité d'ajouter des composants à la demande.
- ✓ Développement orienté objet.
- ✓ Du fait de sa popularité, de nombreux protocoles sont à priori disponibles pour NS-2.

5.2 Les inconvénients :

- ✓ la modélisation dans NS-2 reste une tâche complexe : il n'y a pas d'interface graphique .
- ✓ Une forte technicité est requise pour utiliser ce simulateur.

IV. GloMoSim (Global Mobile information system Simulator)

1. Présentation générale

GloMoSim (Global Mobile information system Simulator) [site 1] est un environnement de simulation pour les réseaux sans-fil de grande taille. Il a été développé par le PCL (Laboratoire de Calcul Parallèle) de l'UCLA (Université de Californie à Los Angeles) dans le cadre du projet DARPA GLOMO (Global Mobil information system). GloMoSim a donné suite à Qualnet, le simulateur commercial de l'entreprise « Scalable Network » affiliée à l'UCLA. GloMoSim s'appuie sur l'environnement PARSEC (Parallèle Simulation Environnent for complexe systèmes) qui est un environnement de simulation pour l'exécution séquentielle ou parallèle de modèles de simulation à événements discrets.

Pour exécuter GloMoSim, le compilateur PARSEC est nécessaire. Pour développer de nouveaux protocoles dans GloMoSim, l'utilisateur doit avoir une certaine familiarité avec PARSEC, mais il n'est pas requis d'une connaissance de niveau expert. Pour la plupart des protocoles, il suffit d'écrire purement code C avec l'ajout de quelques fonctions PARSEC.

[Site 1]

2. PARSEC

PARSEC (pour parallèle environnement de simulation pour les systèmes complexes) est un langage de simulation C développé par le Laboratoire d'Informatique du Parallélisme à l'UCLA, pour une exécution séquentielle et parallèle des modèles de simulation à événements discrets [BRM 98]. Il peut également être poursuivi comme un langage de programmation parallèle.

PARSEC adopté l'approche de l'interaction des processus de la simulation à événements discrets. Un objet (également désignée tant que processus physique) ou un ensemble d'objets dans le système physique est représenté par un processus logique. Un de ses avantages les plus importants est sa possibilité d'exécuter un modèle de simulation à événement discret par plusieurs protocoles de simulation asynchrone parallèle dans différentes architectures parallèle, il sépare entre le modèle de simulation et les algorithmes (séquentielle ou parallèle) utilisés pour l'exécution du modèle.

Les objets dans un système physique (processus physique) sont modélisés par des processus logiques nommés 'entity'. Les interactions entre les processus physiques sont modélisées par l'échange de messages entre ses entités correspondantes.

2.1 Entité :

Une entité représente un objet spécifique dans le système physique, chaque programme dans PARSEC doit avoir une entité principale « driver » qui initialise l'exécution de la simulation qui a le rôle de la fonction « main » dans le langage « C ». La déclaration d'une entité est similaire à la déclaration d'une fonction en « C », mais l'entité ne retourne pas de valeur, cette déclaration définit un type d'entité « enum » sur lequel on peut créer plusieurs instances, en exécutons l'instruction 'new'. [JNU 04]

2.2 Message :

PARSEC utilise des objets de type « message » composés de nom et une liste de paramètres, la définition d'un message est similaire à la définition de structure en « C », les paramètres de message peut inclure des tableaux de taille fixe et qui passent par valeur.

Il est possible d'utiliser les pointeurs comme des paramètres de message, mais c'est dangereux dans une architecture de mémoire distribuée.

Les entités communiquent entre elles en utilisant les messages, chaque entité a un buffer unique, des primitives d'envoi et de réception asynchrone de messages sont fournies pour respectivement le dépôt et l'enlèvement des messages du buffer. [JNU 04]

Ces primitives sont :

send () : Une entité envoie un message par l'instruction send, chaque message sera estampillé et déposé dans le buffer d'entité destinataire.

receive () : Une entité accepte les messages de son buffer en exécutant l'instruction receive.

[JNU 04]

2.3 Événement :

Chaque événement dans le modèle de simulation à événement discret simule certaines activités dans le système physique qui entraîne d'autres objets, chaque événement est associé à un estampille qui indique le temps auquel l'événement correspond apparaît dans le système.

Le temps de la simulation peut seulement avancer quand l'entité reçoit un message ou quand elle exécute l'instruction « hold ». [JNU 04]

2.4 L'exécution parallèle :

En plus de l'algorithme séquentiel, parsec supporte deux types d'algorithme de synchronisations parallèles : conservatrice et optimiste, qu'on va les voir par la suite.

Pour une exécution parallèle certaines restrictions doivent être suivies :

1. variables globales : la simulation parallèle en PARSEC ne doit pas utiliser les variables globales, il est interdit pour deux entités d'avoir un pointeur sur la même donnée.
2. L'état d'une entité doit être seulement déterminé par des variables déclarés dans cette entité, et ces variables ne doivent pas être partagées avec d'autres entités, ces restrictions sont pour les raisons suivantes :
 - ✓ PARSEC assume que tout changement dans l'état de l'entité est le résultat de message, cette règle est violée avec un état d'entité globale.
 - ✓ La simulation optimiste a besoin que l'état d'entité sauvegardé ne fonctionne pas avec l'état partagé, les variables globales ne sont pas contrôlées par le système.
 - ✓ La mémoire partagée a une copie des variables globales quant au mémoire distribuée elle a N copies non consistantes.
3. Les pointeurs ne peuvent pas être passés dans des paramètres de message ou les paramètres d'entité.

3. Structure des répertoires de GloMoSim

Après avoir téléchargé et simulateur GloMoSim, il devrait contenir les répertoires suivants:

application : contient le code pour la couche d'application

bin : pour l'exécutable et les fichiers d'entrée/sortie.

doc : contient une documentation pour le simulateur.

include: contient les fichiers d'inclusion (.h).

mac : contient le code pour la couche mac.

main : contient les fichiers principaux tel que driver.pc (le point d'entrée)

Network : contient le code pour la couche réseau.

radio : contient le code pour la couche radio.

transport : contient le code pour la couche transport.

scenario : contient les fichiers d'entré et d'initialisation pour les différents scénarios.

Cette structure facilite la tâche de modification et d'amélioration telle que l'ajout d'un nouveau protocole dans une couche particulière, et l'ajout d'un nouveau calcul statistique.

4. Architecteur GloMoSim [JNU 04]

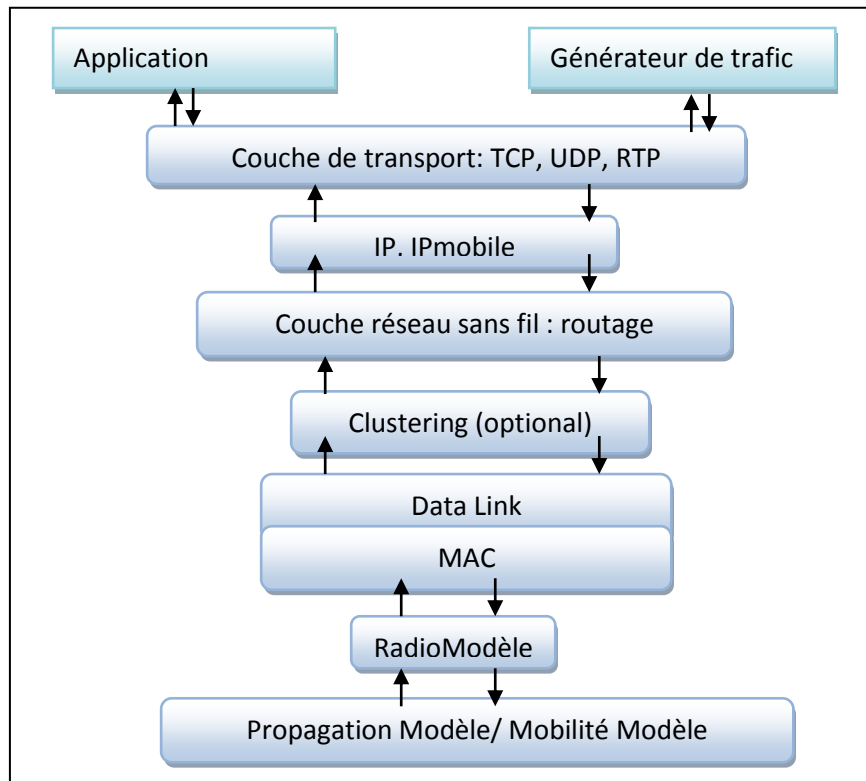


Figure II-5 : Architecteur GloMoSim

5. Utilisation de Simulateur GloMoSim

Après avoir installé avec succès GloMoSim ,une simulation peut être démarrée en exécutant la commande suivante dans le répertoire « ../ BIN »

./glomosim config.in

Fichier « confih.in » contient les paramètres de configuration de la simulation (un exemple d'un tel fichier est../ CONFIG.IN). Un fichier appelé GLOMO.STAT est produit à la fin de la simulation et contient toutes les statistiques générées. [JNU 04]

6. L'outil de visualisation

GloMoSim dispose d'un outil de visualisation qui est indépendant de la plateforme, car il est codé en Java. Pour initialiser l'outil de visualisation, nous devons exécuter à partir du répertoire java gui suit :GlomoMain java. Permet de déboguer et vérifier les modèles et scénarios, reprendre et l'exécution de l'étape ; transmissions spectacle de paquets, les groupes spectacle de mobilité dans différentes couleurs et montrent les statistiques.

La couche radio est affichée dans l'outil de visualisation comme suit : Quand un nœud transmet un paquet, un lien jaune est tiré de ce nœud à tous les nœuds dans sa gamme de puissance.

Comme chaque nœud reçoit le paquet, le lien est effacé et une ligne vert est dessinée pour une réception réussie et une ligne rouge est tracée pour la réception échouée. Aucune distinction n'est faite entre les différents types de paquets (ex : paquets de contrôle vs paquets régulières, etc..) [KXU 04].

7. Configuration de un scénario

7.1 Fichier de configuration (Config.in)

Les principaux paramètres de configuration pour la mise en place d'un scénario sont définis dans le fichier de CONFIG.IN. Ces paramètres sont les suivants :[RBA 96]

paramètre	Description
SIMULATION TEMPS	Durée maximale de simulation. La partie numérique peut être suivie par lettres en option pour modifier le temps de simulation. Par exemple, 100 ns (100 nano-secondes), 100MS (100 milli-secondes), 100S ou 100 (100 secondes), 100M (100 minutes), 100H (100 heures) et 100D (100 jours).
SEED	Est un nombre aléatoire utilisé pour initialiser une partie de différents nombres générés aléatoirement dans la simulation.
TERRAIN-DIMENSIONS	Zone de terrain simulé en mètres.
NOMBRE DE-nœuds	Nombre de nœuds simulés

Table II-1 : Table de configration de nœud

Les valeurs par défaut de ces paramètres dans le fichier de CONFIG.IN sont : [RBA 96]

SIMULATION-TIME 15M

SEED 1

TERRAIN-DIMENSIONS (700, 700)

NUMBER-OF-NODES 30

paramètre	Description
NODE-PLACEMENT	Représente la stratégie de placement de nœud.
MOBILITY	Représente le modèle de mobilité.

Le paramètre **NODE-PLACEMENT** peut être attribué les valeurs suivantes :

[RBA 96]

RANDOM (nœuds sont placés au hasard dans le terrain physique),

UNIFORME (sur la base du nombre de nœuds dans la simulation, l' relief physique est divisé en un certain nombre de cellules. Dans chaque cellule, un nœud est placé de manière aléatoire),

GRID (Nœud placement commence à 0,0 et sont placés sous forme de grille à chaque nœud GRID-UNIT loin de ses voisins, le nombre de nœuds doit être carré d'un entier) et **FILE** (Position de nœuds est lu à partir NODE-PLACEMENT-FILE) .

Les valeurs par défaut de ces paramètres dans le fichier de CONFIG.IN et un exemple fichier NODES.INPUT sont : [RBA 96]

```
# NODE-PLACEMENT FILE
# NODE-PLACEMENT-FILE ./nodes.input
# NODE-PLACEMENT GRID
# GRID-UNIT 30
# NODE-PLACEMENT RANDOM
NODE-PLACEMENT UNIFORM
```

Figure II-6 : fichier NODES.INPUT

paramètre	Description
PROPAGATION-LIMIT	Les signaux en dessous de ce paramètre (en dBm) ne sont pas livrés. Cette valeur doit être inférieure à RADIO-RX-SENSIBILITE + RADIO-ANTENNE-GAIN d'un nœud dans le modèle. Dans le cas contraire, résultats de la simulation peuvent être incorrectes. Valeur inférieure devrait faire la simulation plus précise, mais elle aussi faire l'temps d'exécution plus long.
TEMPERATURE	température de l'environnement (en K).
RADIO-TYPE	Modèle de radio pour transmettre et recevoir des paquets.
RADIO-FREQUENCY	Fréquence en Hertz.
RADIO-BANDWIDTH	La bande passante en bits par seconde.

[RBA 96]

Paramètre	Description
RADIO-RX-TYPE	Spécifie le modèle de réception de paquets.
RADIO-TX-POWER	Puissance de transmission radio (en dBm).
RADIO-ANTENNA-GAIN	Gain de l'antenne (en dB).
RADIO-RX-SENSITIVITY	Sensibilité de la radio (en dBm).
RADIO-RX-THRESHOLD	Puissance minimum pour le paquet reçu (en dBm).

Dans le paramètre RADIO-RX-TYPE, lorsque le paramètre SNR-bornée est utilisé, si le signal de Bruit (SNR) est plus de RADIO-RX-SNR-SEUIL (en dB), il reçoit le signal sans erreur. Sinon, le paquet est abandonné. RADIO-RX-SNR seuil doit être spécifié. Le BER-Paramètre BASE regarde taux d'erreur binaire (BER) dans le tableau SNR-BER spécifié par BER-TABLE-FILE. [RBA 96]

Paramètre	Description
MAC-PROTOCOL	Définition de Medium Access Protocol.
PROMISCUOUS-MODE	Il est mis à OUI si nœuds veulent entendre les paquets destinés au nœud voisin. Actuellement, cette option doit être réglée sur OUI pour que DSR. Réglage à NO peut économiser une quantité négligeable de temps pour d'autres protocoles.
NETWORK-PROTOCOL	Définition du protocole réseau.
ROUTING-PROTOCOL	Définition du protocole de routage.
APP-CONFIG-FILE	Spécifie le fichier qui configure des applications telles que FTP, Telnet et CBR.

Exemple fichier confige .in

```

**** GloMoSim Configuration File ****
SIMULATION-TIME 15M
SEED 1
TERRAIN-DIMENSIONS (700, 700)
NUMBER-OF-NODES 2
NODE-PLACEMENT FILE
NODE-PLACEMENT-FILE ./nodes.input
# NODE-PLACEMENT GRID
# GRID-UNIT 30
# NODE-PLACEMENT RANDOM
# NODE-PLACEMENT UNIFORM
MOBILITY NONE
# RandomWaypoint and itsrequired parameters.
#MOBILITY RANDOM-WAYPOINT
#MOBILITY-WP-PAUSE 30S
#MOBILITY-WP-MIN-SPEED 0
#MOBILITY-WP-MAX-SPEED 10
#MOBILITY TRACE
#MOBILITY-TRACE-FILE ./mobility.in

#MOBILITY PATHLOSS-MATRIX

# The following parameters are necessary for all
the mobility models

MOBILITY-POSITION-GRANULARITY 0.5
PROPAGATION-LIMIT -111.0
#PROPAGATION-PATHLOSS FREE-SPACE
PROPAGATION-PATHLOSS TWO-RAY
#PROPAGATION-PATHLOSS PATHLOSS-
MATRIX
NOISE-FIGURE 10.0
TEMPERATURE 290.0
RADIO-TYPE RADIO-ACCNOISE
#RADIO-TYPE RADIO-NONOISE
RADIO-FREQUENCY 2.4e9
RADIO-BANDWIDTH 2000000
RADIO-RX-TYPE SNR-BOUNDED
RADIO-RX-SNR-THRESHOLD 10.0
#RADIO-RX-SNR-THRESHOLD 8.49583
#RADIO-TX-POWER 15.0
RADIO-ANTENNA-GAIN 0.0
RADIO-RX-SENSITIVITY -81.0
RADIO-RX-THRESHOLD -81.0

MAC-PROTOCOL 802.11
#MAC-PROTOCOL CSMA
#MAC-PROTOCOL MACA
#MAC-PROTOCOL TSMA
#TSMA-MAX-NODE-DEGREE 8
#MAC-PROPAGATION-DELAY 1000NS
#PROMISCUOUS-MODE NO
#####
#
# Currently the only choice.
NETWORK-PROTOCOL IP
NETWORK-OUTPUT-QUEUE-SIZE-PER-PRIORITY 100
#RED-MIN-QUEUE-THRESHOLD 150
#RED-MAX-QUEUE-THRESHOLD 200
#RED-MAX-MARKING-PROBABILITY 0.1
#RED-QUEUE-WEIGHT .0001
#RED-TYPICAL-PACKET-TRANSMISSION-TIME
64000NS
#ROUTING-PROTOCOL BELLMANFORD
ROUTING-PROTOCOL AODV
#ROUTING-PROTOCOL DSR
#ROUTING-PROTOCOL LAR1
#ROUTING-PROTOCOL WRP
#ROUTING-PROTOCOL FISHEYE
#ROUTING-PROTOCOL ZRP
#ZONE-RADIUS 2
#ROUTING-PROTOCOL STATIC
#STATIC-ROUTE-FILE ROUTES.IN
APP-CONFIG-FILE ./app.conf
APPLICATION-STATISTICS NO
TCP-STATISTICS NO
UDP-STATISTICS YES
ROUTING-STATISTICS NO
NETWORK-LAYER-STATISTICS NO
MAC-LAYER-STATISTICS NO
RADIO-LAYER-STATISTICS NO
CHANNEL-LAYER-STATISTICS NO
MOBILITY-STATISTICS YES
GUI-OPTION NO
GUI-RADIO NO
GUI-ROUTING NO

```

Figure II-7 : fichier config.in

V. Conclusion

Dans ce chapitre, nous avons tout d'abord représenté les capacités et le caractère des deux simulateurs NS-2 et GloMoSim et environnements de simulation existant sur les deux simulateurs.

Dans le chapitre suivant, nous allons parler de la différence qualitative entre les deux simulateurs Ns-2 et GloMoSim.

Capitre III. Comparaison qualitative entre « NS-2 » & « GloMoSim »

Chapitre « 3 » : Comparaison qualitative
entre « NS-2 » & « GloMoSim »

I. Introduction

Dans ce chapitre, nous allons mener une comparaison entre les deux simulateurs en ce qui concerne le paramétrage des simulations et les modèles implémentés dans :

- ✓ L'emplacement des nœuds,
- ✓ Le scénario de mobilité,
- ✓ Terrain physique,
- ✓ Génération de trafic,
- ✓ Modules de protocoles implémentés.

1. Placement des nœuds

1.1 Random : les nœuds sont placés aléatoirement sur le terrain physique (les dimensions du terrain sont précisées par l'utilisateur).

1.2 Uniform : en se basant sur le nombre de nœuds, le terrain est divisé en cellules. Dans chaque cellule, un nœud est placé aléatoirement.

1.3 Grid: le premier nœud se place à une position bien déterminée. Les autres nœuds sont placés en ordre à des distances égales pour former une sorte de grille. L'utilisateur précise toutes les informations nécessaires pour pouvoir construire cette grille.

1.4 A partir d'un fichier : un fichier contient l'emplacement de chaque nœud.

1.5 Manuellement : à l'aide d'un environnement graphique, en utilisant le « glisser poser ».

	NS-2	GloMoSim
Random	X	X
Uniform		X
Grid		X
A partir d'un fichier	X	X
Manuellement		

Table III-1 : Comparatif entre NS-2 et GloMoSim dans le placement des nœuds

2. Scénario de mobilité

- 2.1 Aucun** : les nœuds sont fixes
- 2.2 Randomwaypoint** : Le nœud choisit aléatoirement une destination sur le terrain de et s'y rend avec une vitesse moyenne choisie. Lorsqu'il atteint sa destination, le nœud fait une pause. Cette procédure se répète jusqu'à la fin de la simulation.
- 2.3 Fichier de trace** : la trajectoire des mobiles est extraite d'un fichier de trace. Cette méthode permet ainsi de décentraliser l'implémentation des modèles à l'extérieur du simulateur.
- 2.4 Randomdrunken** : si un nœud se trouve à la position (x,y) à l'instant t , sa position à l'instant suivant est tirée dans $\{(x-1, y), (x+1, y), (x, y-1), (x, y+1)\}$. Le déplacement est conditionné par le non-dépassement du terrain physique.

	NS-2	GloMoSim
Aucun	X	X
Randomwaypoint		X
Fichier de trace	X	X
Randomdrunken		X

Table III-2 : Comparatif entre NS-2 et GloMoSim dans le Scénario de mobilité

Avec NS-2 on peut préciser pour un nœud donné, un point de départ, un point de d'arrivée, le temps du commencement du déplacement et la vitesse (en m/s) pour atteindre cette destination. Avec GloMoSim ceci n'est pas possible car pour définir un chemin de mobilité d'un nœud, on définit un point de départ, un point de d'arrivée et le temps de l'atteinte de la destination. C'est GloMoSim qui spécifie une vitesse constante pour atteindre la destination au temps spécifié.

3. Terrain physique

Flat grid : une grille plate donnant les dimensions du terrain.

Un plan bitmap : le terrain peut indiquer la présence d'obstacles en plus il peut intégrer une image comme arrière-plan tel qu'une carte.

	NS-2	GloMoSim
Flat grid	X	X
Un plan bitmap	X	

Table III-3 : Comparatif entre NS-2 et GloMoSim dans le modèle du terrain physique

4. Génération de trafic

A partir d'un fichier : détermine les sources, les puits, le type du flux qui transite entre eux avec les paramètres nécessaires pour décrire ce flux comme par exemple le temps du commencement du trafic, la taille des paquets à envoyer etc.

Manuellement : a l'aide de l'outil graphique on accède à chaque nœud individuellement pour le configurer.

	NS-2	GloMoSim
A partir d'un fichier	X	X
Manuellement		

Table III-4 : Comparatif entre NS-2 et GloMoSim La génération du trafic

5. Modules implémentés

Chacun des simulateurs « NS-2 » et « GloMoSim » implémente un ensemble de modules. Chaque module présente un protocole ou un modèle qui caractérise une couche donnée. Comme le montre le (tableau 3-5), les deux simulateurs ne présentent pas des similitudes dans leurs modules implémentés. Cette diversité rend la comparaison plus difficile. En effet pour pouvoir comparer le comportement des deux simulateurs, il faut les paramétrer de la même façon (même protocoles, même modèles) pour pouvoir par la suite tirer des conclusions. Mais ce qui rassure dans cette situation c'est le fait qu'on peut la plupart des cas trouvé des intersections entre les deux simulateurs. Par exemple, on peut utiliser l'application « stg/rtg » de NS-2 pour obtenir l'application CBR implémenté par GloMoSim.

	NS-2	GloMoSim
Physique	PHY, WPHY, AWPBY	Two ray et Free space
Liaison de donnés (Mac)	Ether, WLANAD, WLANIN, LBP, STP ARP, 2F-BLSR	IP with AODV, Bellman- Ford, DSR, Fisheye, LAR scheme 1, ODMRP, WRP
Réseau (routage)	routage dans les réseaux ad-hoc (aodv, dsr,dsdv, tora, amodv),	IP with AODV, Bellman-Ford, DSR, Fisheye, LAR1 ,ZRP,WRP
Transport	TCP, UDP RTP/RTCP, SDP	TCP et UDP
Application	générateur de trafic (CBR, FTP), http et Telnet	CBR , FTP , http et Telnet

Table III-5 : Ensemble des modules implémentés par les deux simulateurs GloMoSim et NS-2

Ce tableau indique la différence entre les deux simulateurs.

Stream principales	Simulateurs réseau	
	NS-2	GloMoSim
licence	open Source	open Source
GUI	No	limité
interface	C++ ,OTcl	C
Modules disponibles	filaire, Wireless, Ad-Hoc et réseaux sans fil Capteur réseaux	filaire, sans fil et Ad-Hoc réseaux
Documentation et Support Utilisateurs	excellent	pauvres
évolutivité	petit	grand
extensible	oui	oui
simulation Technique	Simulation à événements discrets	Simulation à événements discrets

Table III-6 : Comparaison des simulateurs NS-2 & GloMoSim de réseau

6. Comparaison des paramètres de protocole entre les deux simulateurs

Simulateur / Protocol	NS2	GloMoSim
IEEE 802.11 CSMA/CA MAC,	X	X
UDP	X	X
FTP	X	X
CBR	X	X
Fisheye		X
ODMRP		X
WRP		X
Bellman-Ford		X
DSR	X	X
MACA		X
AODV	X	X

Table III-7 : Comparatif entre NS-2 et GloMoSim dans les paramètres de protocole implémenter

II. Conclusion

Nous avons déjà mentionné format du logiciel de simulation GloMoSim et Ns-2 des avantages et des inconvénients et la plupart des réglages sont disponibles pour chaque simulateur. Ne soyez pas et aucun d'entre eux fournissent un bon réseau de soutien et simule parfaitement. Et si nous avons cette comparaison entre simulation et Ns-2 et GloMoSim et qui fournira une bonne expérience pour les utilisateurs savants réseau ad hoc ou autre réseaux.

Capitre IV. Comparaison quantitative entre « NS-2 » et « GloMoSim »

Chapitre « 4 » : Comparaison quantitative entre
« NS-2 » & « GloMoSim »

I. Introduction

Dans le chapitre précédent, la comparaison entre NS-2 et GloMoSim est théorique, Cette comparaison ne suffit pas. En effet, il est nécessaire de passer à l'expérimentation en réalisant un ensemble de scénario avec les deux simulateurs. Ensuite, la comparaison est faite en termes de résultats et de temps de simulation.

II. Paramètres de simulation

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

Paramètre	Valeur
Modèle de propagation Radio	Tow Ray Ground
Couche MAC	IEEE 802.11
Protocole de routage	AODV
Couche transport	UDP
Type du trafic généré	CBR
Débit du trafic	200 paquets/seconde
Taille paquet	512 octets
Débit du lien radio	11 Mbits/s
Portée de transmission	250 mètres

Table IV-1 : Paramètres communs aux scénarii

La couche MAC est implémentée selon la norme IEEE 802.11b. En particulier, le mécanisme d'accès utilisé dans cette couche est la méthode DFC (Distributed Coordination Function) qui est typiquement le mécanisme CSMA/CA (Carrier Sense Multiple Access With Collision Avoidance). La norme IEEE 802.11b opère à des débits compris entre 1 et 11 Mbits/s.

La variation de débits dépend de la qualité de l'environnement radio (interférences, obstacles, distances entre équipements,...).

AODV est choisi comme protocole de routage puisqu'il est implémenté par les deux simulateurs.

Le débit du trafic ainsi que la taille des paquets ont été sélectionnés, car d'après des études antérieures comme [JNU 04], nous avons de meilleurs résultats avec ces choix.

Le flux CBR est choisi vu qu'il est simple. Il est caractérisé par un débit constant et typiquement utilisé par les applications vidéo et audio. Ce type d'application exige des délais stricts. Un flux CBR est généré en fixant la taille du paquet (dans notre cas 512 octets) et le temps d'inter-arrivée entre deux paquets consécutifs (dans notre cas 0.005s).

III. Portée de transmission

1. Description

Avec NS-2, la portée de transmission peut être déterminée en affectant la valeur désirée en mètres au paramètre « Transmission Range » qui se trouve dans le module « WPHY » (Wireless PHYsical-layer module). Avec GloMoSim[JNU 04], la portée de transmission peut être déterminée en effectuant trois étapes :

Détermination du modèle de propagation, dans notre cas c'est le modèle « Tow Ray Ground»,

- ✓ Détermination de la puissance minimale pour recevoir un signal,
- ✓ Détermination de la puissance de transmission.

Dans ce qui suit, nous allons appliquer ces conditions pour la configuration de la portée de transmission à 250 mètres.

2. Résultats de simulation

Pour vérifier qu'effectivement la portée de transmission vaut 250 mètres, nous procédons à la simulation suivante : Une topologie contenant deux nœuds N_0 et N_1 . Le nœud N_0 envoie au nœud N_1 un flux CBR. Le trafic commence à 1s et s'achève à 250s. Le temps de simulation étant 300s et le protocole de routage étant AODV. Le nœud N_1 est fixe, alors que le nœud N_0 se déplace vers la droite avec une vitesse égale à 1m/s (figure 4.1). La distance initiale entre les deux nœuds vaut 50 mètres.

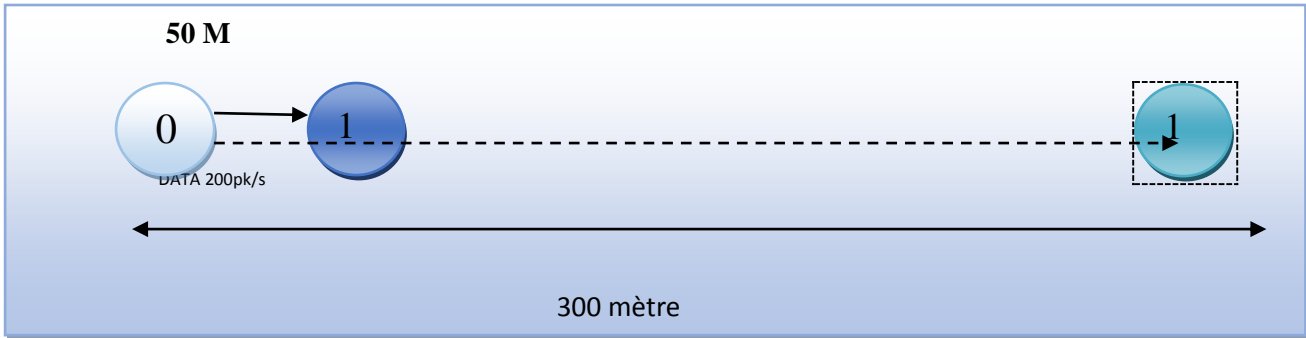


Figure IV-1 : Scénario pour la vérification de la portée de transmission

En se déplaçant, la distance (N1, N2) augmente, ainsi peu à peu, le mobile N2 quitte la zone de couverture de N1. Cet événement est marqué par le fait que N2 ne reçoit plus les paquets envoyés par N1. Par conséquent, le nombre de paquets reçus devient constant. Si nous mesurons à cet instant la distance (N1, N2), nous devons s’attendre à une valeur égale ou proche des 250 mètres

2.1 Appliqué le scénario dans le simulateur NS-2

Cette figure représente fichier scripte.tcl"simulation_rés_aodv.tcl" Il contient la position initial des Nœud 0 et Nœud 1 et génération de mouvements nœud 1 et setup UDP connexion

```

*simulation_rés_AODV.tcl (-/Desktop/TCL2/1) - gedit
simulation_rés_AODV.tcl
for {set i 0} {$i < $val(nn)} {incr i} {
    set node_($i) [$ns node]
}

# Provide initial location of mobilenodes
$node_0 set X_ 50.0
$node_0 set Y_ 200.0
$node_0 set Z_ 0.0

$node_1 set X_ 100.0
$node_1 set Y_ 200.0
$node_1 set Z_ 0.0

# Generation of movements node_1 . vitess Node(1) 1m/s
$ns at 0 "$node_1 setdest 400.0 200.0 1"

#Setup a UDP connection
set udp [new Agent/UDP]
$ns attach-agent $node_0 $udp
set null [new Agent/Null]
$ns attach-agent $node_1 $null
$ns connect $udp $null
$udp set fid_ 2

#Setup a CBR over UDP connection
set cbr [new Application/Traffic/CBR]
$cbr attach-agent $udp
$cbr set type_CBR
$cbr set packet_size_ 512
$cbr set interval_ 0.005
$cbr set random_ false
    
```

Figure IV-2 : simulation_rés_aodv.tcl

L'exécution de ce script se fait, grâce à ns, en utilisant simplement la commande :

« ns simulation_rés_aodv.tcl » .

```
hp@ubuntu:~/Desktop/TCL2/1$ ns simulation_rés_AODV.tcl
num_nodes is set 2
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
CBR packet size = 512
CBR interval = 0.0050000000000000001
CBR rat = 819200
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5, distCST_ = 550.0
SORTING LISTS ...DONE!
hp@ubuntu:~/Desktop/TCL2/1$
```

Figure IV-3 : Exécution fichier .tcl

Cette(figure 4-4) illustre l'envoi des paquets entre Nœud 0 et Nœud 1 dans visualisation Ns-2 « nam »

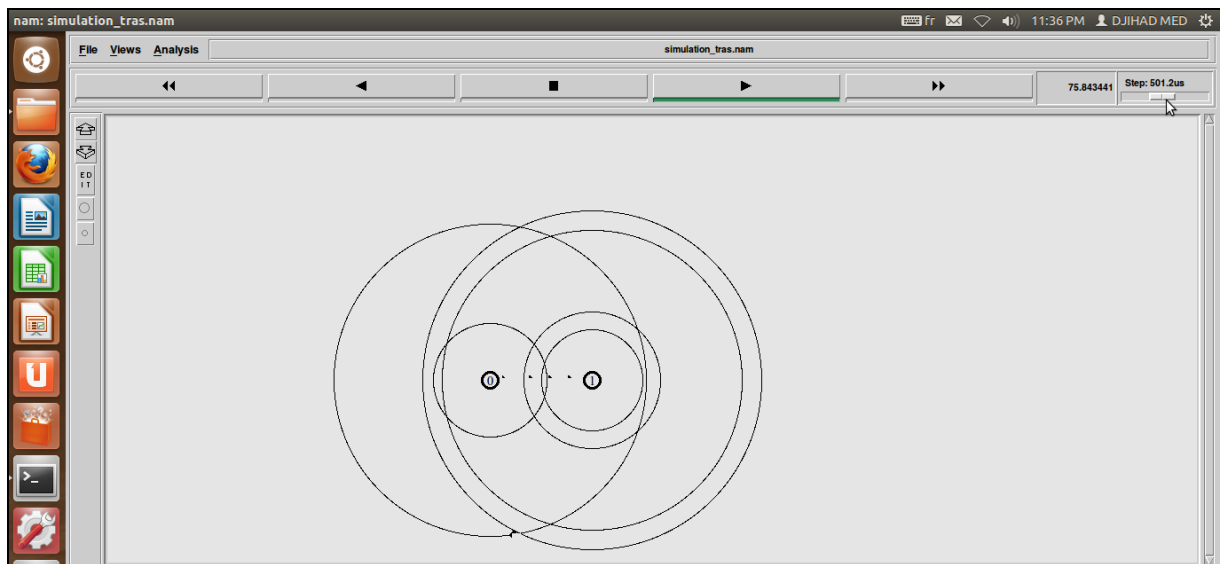


Figure IV-4 : NAM, visualisation représenter la topologie d'un réseau décrit avec NS-2

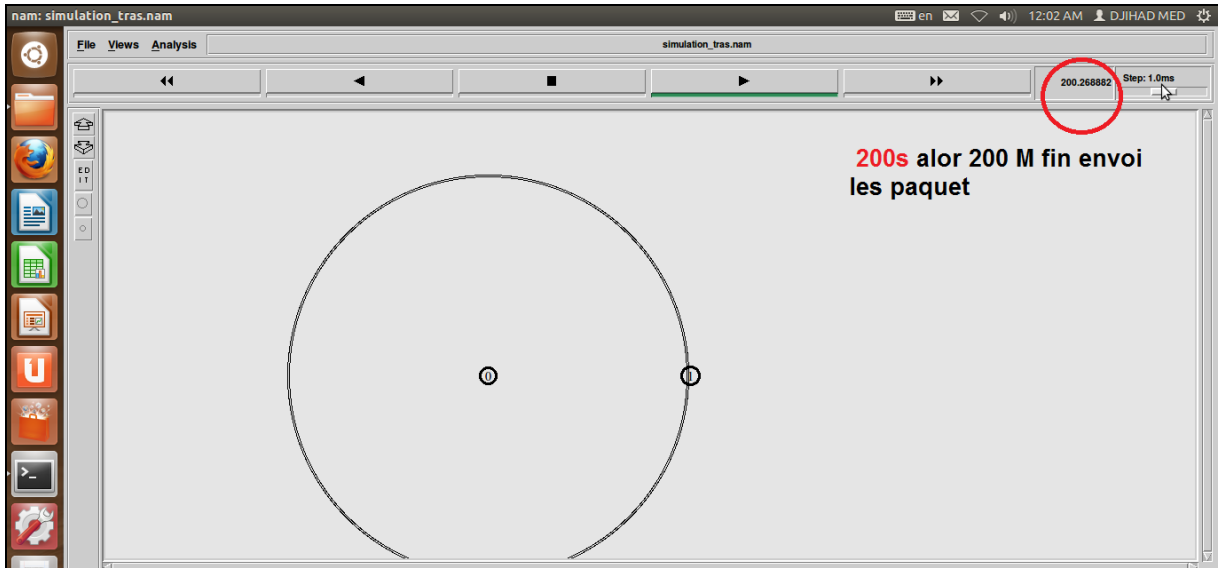


Figure IV-5 : NAM, visualisation représenter la topologie d'un réseau décrit avec NS-2

Lire le fichier de trace fichier.tr voir(figure 4-6)à l'aide de la commande more. À chaque évènement arrivant au niveau d'un lien, une ligne est écrite dans ce fichier :

- **r** signifie que le paquet est reçu par le nœud destinataire ;[ART 01]
- **d** signifie que le paquet est rejeté (perdu).[ART 01]
- **s** Une nœud envoie le paquet « send »

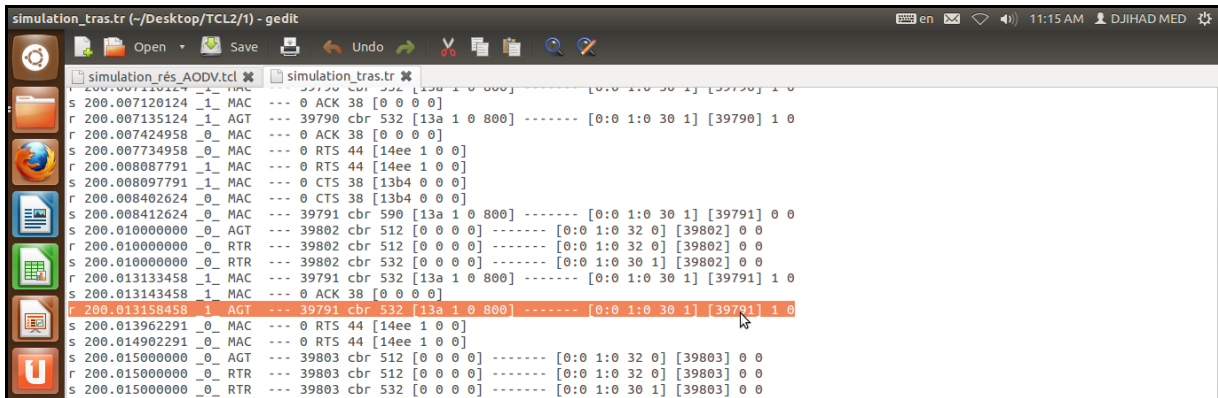


Figure IV-6 : Traces de la simulation

Dans cette (figure 4-6), nous trouvons View Terrace contient des données résultant des simulations, et indique le nombre de paquets envoyés et le moment où ceux-ci sont envoyés les paquets.

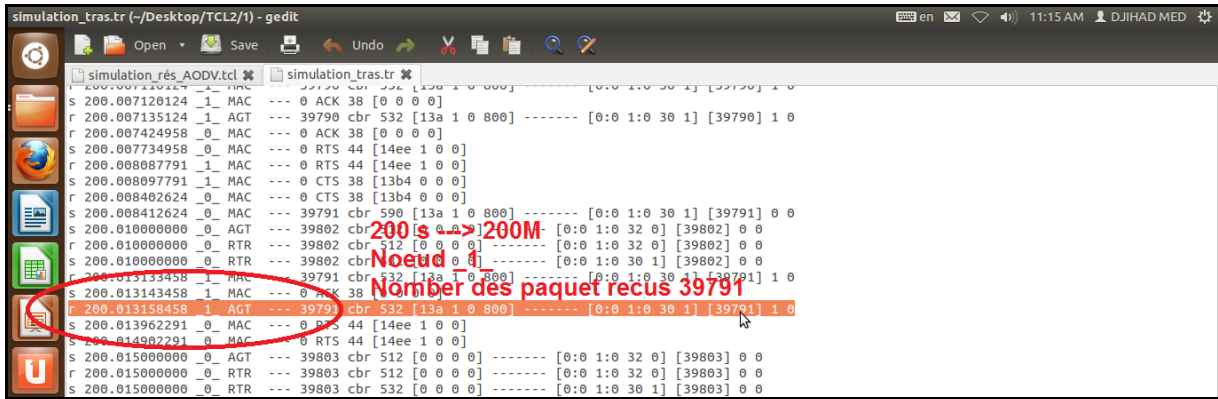


Figure IV-7 : Traces de la simulation

Cette (figure 4-7) montre le temps d'arrêt envoyer et recevoir des paquets sont présentés dans la ligne rouge. Et indique le nombre de paquets reçus.

2.2 Appliqué le scénario dans le simulateur GloMoSim

- Nœud 0 envoie des paquets (génère un trafic CBR) vers Nœud 1
- Toutes les 0,005S secondes envoie un paquet
- Le début des temps d'envoyer des paquets 1S
- Temps arrêt l'envoi des paquets 181S

```
CBR 0 1 0 512 0.005s 1s 181s
```

Figure IV-8 : app.CONFIG pour configure le flux CBR

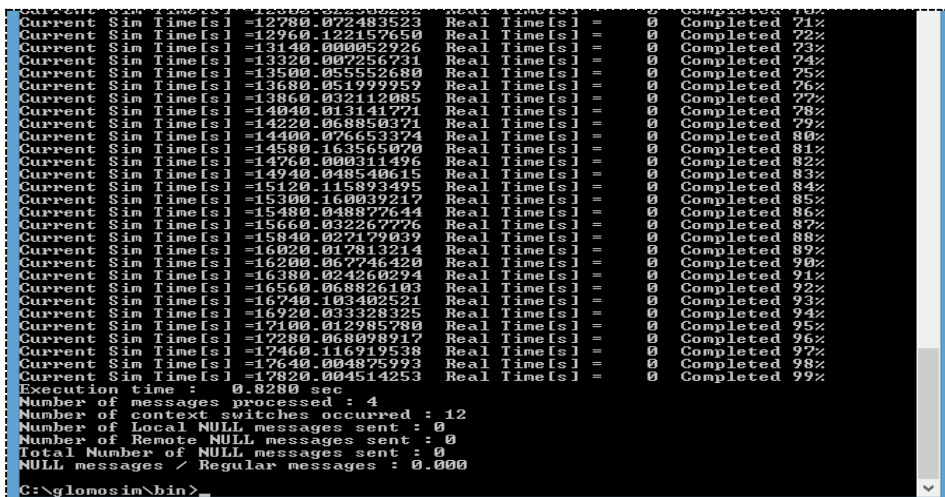


Figure IV-9 : compiler le fichier confige.in

Après compilation de fichier config.in nous trouvons les résultats tels que la réception des paquets et les paquets envoyés dans ce fichier glomo.stat

```

Node: 0, Layer: AppChrClient, (1) Last packet sent at [s]: 180.9900000
Node: 0, Layer: AppChrClient, (1) Session status: Closed
Node: 0, Layer: AppChrClient, (1) Total number of bytes sent: 18432000
Node: 0, Layer: AppChrClient, (1) Total number of packets sent: 36000
Node: 0, Layer: AppChrClient, (1) Throughput (bits per second): 819222
Node: 0, Layer: AppFtpClient, from 0 to 1 (cid = 1), start = 150006498030, end = 150460288444 ns (closed), bytes sent = 7908 B, bytes rcv = 279 B, throughput = 139412 bps
Node: 1, Layer: 802.11, pkts from network: 0
Node: 1, Layer: 802.11, UCAST (non-frag) pkts sent to chanl: 34
Node: 1, Layer: 802.11, BCASF pkts sent to chanl: 0
Node: 1, Layer: 802.11, UCAST pkts rcvd clearly: 36038
Node: 1, Layer: 802.11, BCASF pkts rcvd clearly: 1
Node: 1, Layer: 802.11, retx pkts due to CTS timeout: 1
Node: 1, Layer: 802.11, retx pkts due to ACK timeout: 0
Node: 1, Layer: 802.11, pkt drops due to retx limit: 0
Node: 1, Layer: 802.11, RTS Packets ignored due to Busy Channel 0
Node: 1, Layer: 802.11, RTS Packets ignored due to NAV 0
Node: 1, Layer: TransportUdp, Number of pkts from application 0.
Node: 1, Layer: TransportUdp, Number of pkts to application 36000.
Node: 1, Layer: AppFtpServer, from 0 to 1 (cid = 2), start = 150015415663, end = 150470509489 ns (closed) bytes sent = 279 B, bytes rcv = 7908 B, throughput = 139013 bps
Node: 1, Layer: AppChrServer, (1) Client address: 0
Node: 1, Layer: AppChrServer, (1) First packet received at [s]: 1.005662221
Node: 1, Layer: AppChrServer, (1) Last packet received at [s]: 180.998057809
Node: 1, Layer: AppChrServer, (1) Average end-to-end delay [s]: 0.003062891
Node: 1, Layer: AppChrServer, (1) Session status: Closed
Node: 1, Layer: AppChrServer, (1) Total number of bytes received: 18432000
Node: 1, Layer: AppChrServer, (1) Total number of packets received: 36000
Node: 1, Layer: AppChrServer, (1) Throughput (bits per second): 819234
    
```

Figure IV-10 : résultat de simulation

3. Etude comparative

La figure suivante présente le nombre de paquets reçus par le mobile N1 en fonction de la distance séparant les nœuds N0 et N1.

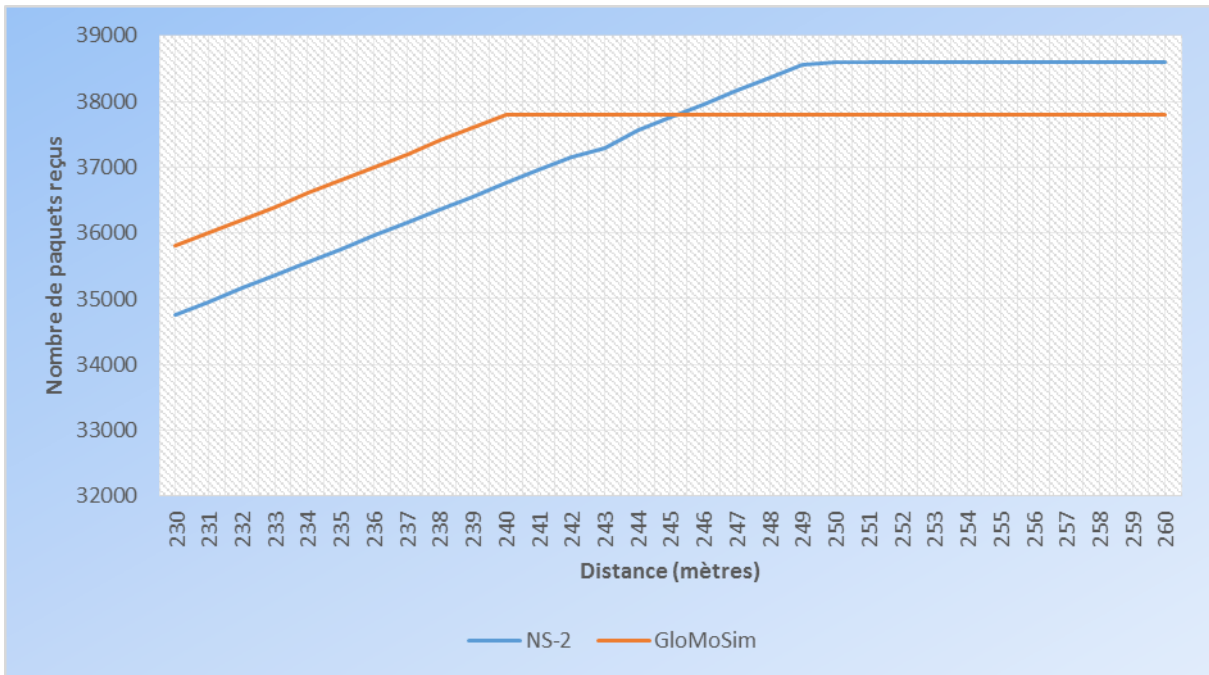


Figure IV-11 : Nombre de paquets reçus par le mobile N1 en fonction de la distance (N0, N1)

Cette (figure 4-11) montre bien que la portée de transmission de NS-2 vaut 250 mètres. En revanche, la portée de transmission de GloMoSim est différente de la valeur prévue. Par conséquent le taux de paquets reçus avec NS-2 sera supérieur à celui avec GloMoSim (37859 paquets pour GloMoSim et 38577 paquets pour NS-2). Ceci est dû à des facteurs (comme l'effet du bruit et des interférences) qui interviennent dans la portée de transmission.. En effet, si nous fixons cette distance, deux mobiles ne peuvent communiquer que si la distance qui les sépare est inférieure ou égale à 250 mètres. Alors qu'en réalité le nœud source peut réussir à envoyer des trames même si la distance qui le sépare du nœud destinataire excède les 250 mètres, car dans ce cas, le calcul des interférences et du bruit décidera si la communication est possible ou pas.

IV. Méthode d'accès au médium

1. Description

La méthode d'accès au médium que nous adoptons pour les simulateurs NS-2 et GloMoSim est : « CSMA/CA » avec le mécanisme « RTS/CTS ». En bref, cette méthode permet l'accès au média en mode avec contention. Son principe de fonctionnement est le suivant : Une station voulant transmettre, écoute le support de transmission, si le support est occupé (une autre station est en train de transmettre), alors la station remet sa transmission à plus tard. Si le support est libre pour un temps « DIFS » (Distributed Inter Frame Space), alors la station va transmettre une trame « RTS » qui a pour but de réserver le canal pendant la durée nécessaire à la transmission.

Une fois la trame « RTS » reçue par la station destinataire, celle-ci va attendre un temps inférieur à « DIFS » (temps « SIFS ») pour ensuite transmettre à son tour une trame « CTS » qui a pour but d'avertir toutes les stations à sa portée, que le canal sera occupé pendant le temps recalculé de la trame « RTS » précédente. Ceci permet en effet d'éviter d'éventuelles collisions pendant la transmission des flux de données. Ce mécanisme peut être considéré comme une tentative d'introduire la qualité de service sur le lien WLAN afin d'être apte à transférer des flux multimédia (le cas du trafic CBR). [Site 3]

2. Résultats de simulation :

Le but du scénario suivant est de créer une compétition sur l'obtention du canal de transmission. Ce scénario consiste à placer quatre nœuds équidistants de 50 mètres. (figure 4-12) Les nœuds N0 et N3 envoient simultanément un flux CBR (de 0.1s à 30s) aux nœuds N1

et N3. Le temps de simulation étant 50s. La distance (N0, N3) est égale à 150 mètres. Dans cette situation, les quatre nœuds se situent dans la zone de couverture de N1 et la probabilité de collision sera élevée.

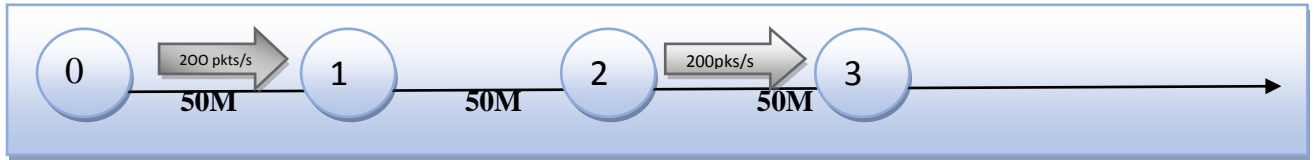


Figure IV-12 : Scénario de test de la méthode d'accès

2.1 Appliqué le scénario dans le simulateur NS-2 :

Paramètre défini

```
# Define options
set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# radio-propagation model
set val(netif) Phy/WirelessPhy ;# network interface type
set val(mac) Mac/802_11 ;# MAC type
set val(ifq) Queue/DropTail/PriQueue ;# interface queue type
set val(ll) LL ;# link layer type
set val(ant) Antenna/OmniAntenna ;# antenna model
set val(ifqlen) 1000 ;# max packet in ifq
set val(nn) 4 ;# number of mobilenodes
set val(rp) AODV ;# routing protocol
set val(x) 700 ;# X dimension of topography
set val(y) 700 ;# Y dimension of topography
set val(stop) 10 ;# time of simulation end
```

Figure IV-13: Le script fichier.tcl parti définition des variables

```
# configure the nodes
$ns node-config -adhocRouting $val(rp) \
    -llType $val(ll) \
    -macType $val(mac) \
    -ifqType $val(ifq) \
    -ifqLen $val(ifqlen) \
    -antType $val(ant) \
    -propType $val(prop) \
    -phyType $val(netif) \
    -channelType $val(chan) \
    -topoInstance $topo \
    -agentTrace ON \
    -routerTrace ON \
    -macTrace ON \
    -movementTrace OFF
```

Figure IV-14 : fichier.tcl parti configuration paramètre les nœud

Après le changement la plupart des paramètres dans le fichier voir figure 4-15

- Nœud 0 envoie des paquets vers Nœud 1
- Nœud 2 envoie des paquets vers Nœud 3
- Toutes les **0,005s** secondes envoie un paquet
- Le début des temps d'envoyer des paquets **0.1s**
- Temps arrêt l'envoi des paquets **30s**

L'émettions des paquets par les Nœud Noeud1 et Noeud3 ce fait de manier parallèle .

```
#le trafic issu de l'agent udp0 est envoyé vers null1
$ns connect $udp0 $null1
#le trafic issu de l'agent udp1 est envoyé vers null3
$ns connect $udp1 $null3
$ns at 0.1 "$cbr0 start"
$ns at 3 "$cbr0 stop"
$ns at 0.1 "$cbr1 start"
$ns at 3 "$cbr1 stop"
```

Figure IV-15 : fichier.tcl

Après l'application de fichier de simulateur nous trouvons les résultats dans le fichier.tr.

```
r 0.050000000 0_ RTR --- 20 cbr 512 [0 0 0 0] ----- [0:0 1:0 32 0] [10] 0 0
s 0.050000000 2_ AGT --- 21 cbr 512 [0 0 0 0] ----- [2:0 3:0 32 0] [10] 0 0
r 0.050000000 2_ RTR --- 21 cbr 512 [0 0 0 0] ----- [2:0 3:0 32 0] [10] 0 0
s 0.050000000 0_ RTR --- 20 cbr 532 [0 0 0 0] ----- [0:0 1:0 30 1] [10] 0 0
s 0.050000000 2_ RTR --- 21 cbr 532 [0 0 0 0] ----- [2:0 3:0 30 3] [10] 0 0
r 0.050464393 3_ MAC --- 7 cbr 532 [13a 3 2 800] ----- [2:0 3:0 30 3] [3] 1 0
s 0.050474393 3_ MAC --- 0 ACK 38 [0 2 0 0]
r 0.050489393 3_ AGT --- 7 cbr 532 [13a 3 2 800] ----- [2:0 3:0 30 3] [3] 1 0
r 0.050778560 2_ MAC --- 0 ACK 38 [0 2 0 0]
s 0.051228560 2_ MAC --- 0 RTS 44 [14ee 3 2 0]
r 0.051580727 3_ MAC --- 0 RTS 44 [14ee 3 2 0]
s 0.051590727 3_ MAC --- 0 CTS 38 [13b4 2 0 0]
r 0.051894893 2_ MAC --- 0 CTS 38 [13b4 2 0 0]
s 0.051904893 2_ MAC --- 3 cbr 590 [13a 3 2 800] ----- [2:0 3:0 30 3] [1] 0 0
r 0.055000000 0_ AGT --- 22 cbr 512 [0 0 0 0] ----- [0:0 1:0 32 0] [11] 0 0
r 0.055000000 0_ RTR --- 22 cbr 512 [0 0 0 0] ----- [0:0 1:0 32 0] [11] 0 0
s 0.055000000 2_ AGT --- 23 cbr 512 [0 0 0 0] ----- [2:0 3:0 32 0] [11] 0 0
r 0.055000000 2_ RTR --- 23 cbr 512 [0 0 0 0] ----- [2:0 3:0 32 0] [11] 0 0
s 0.055000000 0_ RTR --- 22 cbr 532 [0 0 0 0] ----- [0:0 1:0 30 1] [11] 0 0
s 0.055000000 2_ RTR --- 23 cbr 532 [0 0 0 0] ----- [2:0 3:0 30 3] [11] 0 0
r 0.056625060 3_ MAC --- 3 cbr 532 [13a 3 2 800] ----- [2:0 3:0 30 3] [1] 1 0
s 0.056635060 3_ MAC --- 0 ACK 38 [0 2 0 0]
r 0.056650060 3_ AGT --- 3 cbr 532 [13a 3 2 800] ----- [2:0 3:0 30 3] [1] 1 0
r 0.056939227 2_ MAC --- 0 ACK 38 [0 2 0 0]
s 0.057029560 0_ MAC --- 0 RTS 44 [14ee 1 0 0]
```

Figure IV-16 : fichier.tr

2.2 Appliqué le scénario dans le simulateur GloMoSim :

Après le changement de la plupart des paramètres dans le fichier voir (figure 4-17)

- Nœud 0 envoie des paquets vers Nœud 1
- Nœud 2 envoie des paquets vers Nœud 3

- Toutes les **0,005S** secondes envoie un paquet
- Le début de temps d'envoyer des paquets **0.1S**
- Temps arrêter l'envoi des paquets **30S**

```

CBR 0 1 0 512 0.1S 0.1S 30S
CBR 3 4 0 512 0.1S 0.1S 30S
    
```

Figure IV-17 : fichier app.config

```

node: 0, Layer: MacCSMA, Number of UNICAST packets output to the channel: 0
Node: 0, Layer: MacCSMA, Number of BROADCAST packets output to the channel: 14
Node: 0, Layer: MacCSMA, Number of UNICAST packets received clearly: 0
Node: 0, Layer: MacCSMA, Number of BROADCAST packets received clearly: 0
Node: 0, Layer: TransportUdp, Number of pkts from application 290.
Node: 0, Layer: TransportUdp, Number of pkts to application 0.
Node: 1, Layer: MacCSMA, Number of packets from network: 0
Node: 1, Layer: MacCSMA, Number of packets lost due to buffer overflow: 0
Node: 1, Layer: MacCSMA, Number of UNICAST packets output to the channel: 0
Node: 1, Layer: MacCSMA, Number of BROADCAST packets output to the channel: 0
Node: 1, Layer: MacCSMA, Number of UNICAST packets received clearly: 0
Node: 1, Layer: MacCSMA, Number of BROADCAST packets received clearly: 0
Node: 1, Layer: TransportUdp, Number of pkts from application 0.
Node: 1, Layer: TransportUdp, Number of pkts to application 0.
Node: 2, Layer: MacCSMA, Number of packets from network: 0
Node: 2, Layer: MacCSMA, Number of packets lost due to buffer overflow: 0
Node: 2, Layer: MacCSMA, Number of UNICAST packets output to the channel: 0
Node: 2, Layer: MacCSMA, Number of BROADCAST packets output to the channel: 0
Node: 2, Layer: MacCSMA, Number of UNICAST packets received clearly: 0
Node: 2, Layer: MacCSMA, Number of BROADCAST packets received clearly: 0
Node: 2, Layer: TransportUdp, Number of pkts from application 0.
Node: 2, Layer: TransportUdp, Number of pkts to application 0.
Node: 3, Layer: MacCSMA, Number of packets from network: 0
Node: 3, Layer: MacCSMA, Number of packets lost due to buffer overflow: 0
Node: 3, Layer: MacCSMA, Number of UNICAST packets output to the channel: 0
Node: 3, Layer: MacCSMA, Number of BROADCAST packets output to the channel: 14
Node: 3, Layer: MacCSMA, Number of UNICAST packets received clearly: 0
Node: 3, Layer: MacCSMA, Number of BROADCAST packets received clearly: 0
Node: 3, Layer: TransportUdp, Number of pkts from application 290.
Node: 3, Layer: TransportUdp, Number of pkts to application 0.
    
```

Figure IV-18 : résultat de la simulation

Remarque : dans le fichier GloMo.STAT les résultats après l'exécution (voir figure 4-18)

Aucun paquet ni reçu des paquets

4. Etude comparative

Après la simulation du scénario avec les deux simulateurs, nous avons constaté qu'aucun paquet n'est reçu par les nœuds destinataires N1 et N3. Ceci est dû au phénomène suivant. Au début, chacun des nœuds N0 et N2 possède dans la file d'attente de la couche MAC un paquet à émettre. En écoutant le canal, ils ne trouvent aucun signal car le canal est encore vide. Par conséquent, chacun d'eux envoie un message « RTS » pour demander l'accès au canal. Les deux trames « RTS » tombent en collision au niveau de N1. Ainsi, ce dernier ne reçoit pas la trame « RTS » venant de N0. D'autre part, au niveau de N3 les trames « RTS » venant de N0 et N2 tombent en collision pendant que N3 est en train de recevoir la trame « RTS ». Donc, N3 ne réussit pas à recevoir la trame de N2.

V. Scalabilité, simulation et autres facteurs

1. Description

Dans cette section, nous allons considérer un scénario typique [PDA 04]: Une topologie de surface rectangulaire (1000 m, 500 m), contenant 50 nœuds dont 10 sources et 10 puits. Les sources envoient un flux CBR de l’instant t = 0.1s jusqu’à l’instant t = 30s. Le temps de simulation vaut 80s. Tous les nœuds sont mobiles. Le placement des nœuds ainsi que leurs chemins de mobilité sont aléatoires. Pour se faire, nous avons choisir le modèle de placement des nœuds « Random » .

2. Résultats de simulation

2.1 Appliqué le scénario dans le simulateur NS-2 :

```
# Define options
Phy/WirelessPhy set RXThresh_ 0.652e-10 ;# Portée de communication : 385 mètres (-91,1db)
set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# radio-propagation model
set val(netif) Phy/WirelessPhy ;# network interface type
set val(mac) Mac/Tdma ;# MAC type
set val(ifq) Queue/DropTail/PriQueue ;# interface queue type
set val(ll) LL ;# link layer type
set val(ant) Antenna/OmniAntenna ;# antenna model
set val(ifqlen) 1000 ;# max packet in ifq
set val(nn) 50 ;# number of mobilenodes
set val(rp) AODV ;# routing protocol
set val(x) 1000 ;# X dimension of topography
set val(y) 500 ;# Y dimension of topography
set val(stop) 80 ;# time of simulation end
```

Figure IV-19 fichier.tcl définit les variables pour les paramètres

2.2 Appliqué le scénario dans le simulateur Glomosim :

le fichier APP .CONFIG(Figure 4-21) les paramètres des flux CBR transmet .

```
CBR 2 11 0 512 0.005s 0.1s 30s
CBR 3 12 0 512 0.005s 0.1s 30s
CBR 4 13 0 512 0.005s 0.1s 30s
CBR 5 14 0 512 0.005s 0.1s 30s
CBR 6 15 0 512 0.005s 0.1s 30s
```

Figure IV-20 : Fichier APP . CONFIG paramètre de flux CBR

Les simulations montrent le résultat dans un fichier Glomom.STAT dans le dossier BIN

3. Etude comparative

Dans cette comparaison, nous allons calculer les paquets perdus dans plusieurs cas, début 50 paquet/s et 100 paquet/s et 150 paquet/s et cas son mobilité et sans mobilité. entre les deux simulateur « GloMoSim et NS-2 » .

La Formule pour calcule le paquet perdu :

$$\text{taux paquet perdus} = \frac{\text{Nombre des paquets envoyés} - \text{Nombre des paquets reçus}}{\text{Nombre des paquets envoyés}} \times 100$$

Après avoir traité la simulation :

Nous trouvons les résultats sont évidents dans les (figures 4-21) et (figures 4-22) .

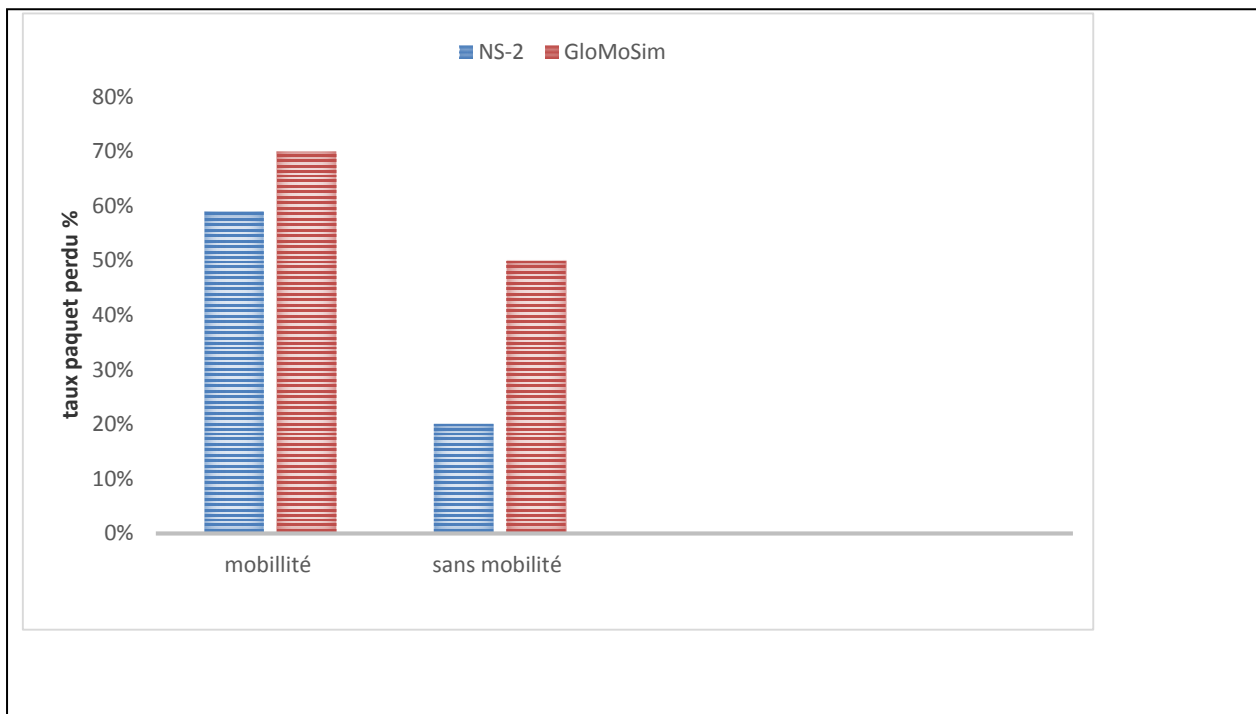


Figure IV-21 : Taux paquet perdu en fonction de la mobilité

Dans la (figure 4-21) le taux de paquets perdus dans le cas de la mobilité de plus que le taux de paquets dans le cas sans mobilité lorsqu'aucun mouvement des nœuds.

La (figure 4-21) montre que la mobilité est un facteur important qui dégrade la performance du réseau en terme de paquets délivrés. Ce ci est dû aux pertes des paquets pendant la phase de reconfiguration du réseau.

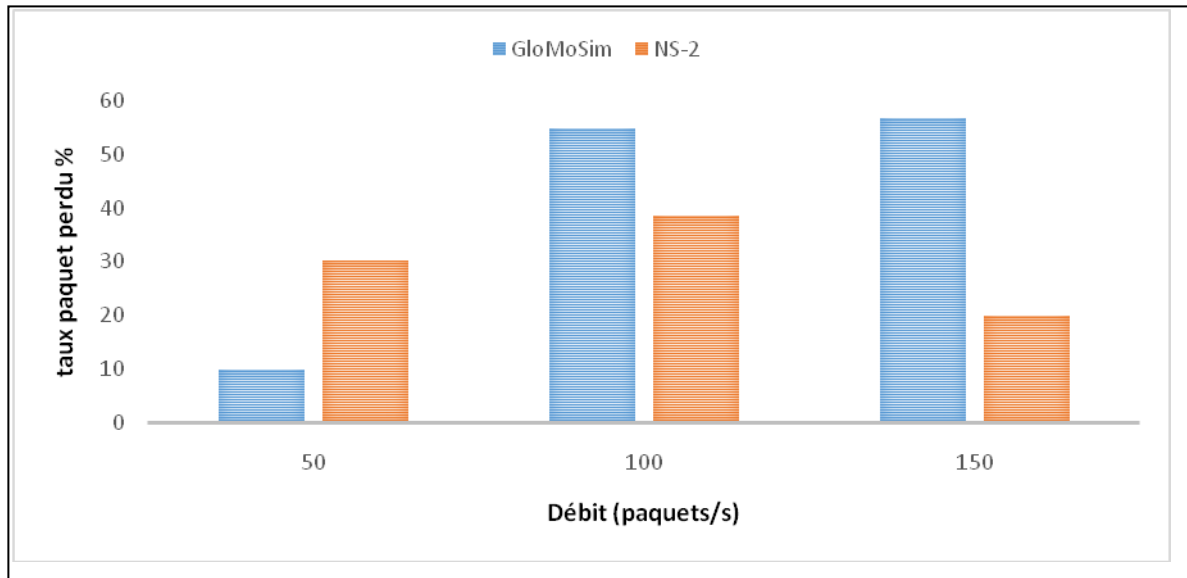


Figure IV-22 : Taux paquets perdus en fonction du débit

Remarque sur la (figure 4-22) résultat le simulateur « GloMoSim » que lorsque vous augmentez début (paquet/s) nous trouvons une augmentation dans le nombre de paquets perdus. Dans la resulta le simulateur « NS-2 » on trouve une augmentation les nombre des paquets perdus dans le case 50paquet/s et le cas 100 paquet/s par conter le case 150 paquet/s

VI. Conclusion

À travers ce dernier chapitre, nous avons vu que NS-2 présente un ensemble d'avantages par rapport à GloMoSim. En revanche, nous avons montré que NS-2 possède les inconvénients suivants :

- ✓ Les statistiques qu'il fournit ne sont pas rigoureuses.
- ✓ Il y a des cas où il ne tient pas des facteurs bruit et d'interférence.

Conclusion générale

L'objectif de ce mémoire de fin d'études était d'évaluer le simulateur Ns-2s en le comparant avec le simulateur GloMoSim et avec les résultats théoriques envisagés. Le choix de l'utilisation de NS-2 est justifié par le fait qu'il est considéré comme étant populaire et a été testé par plusieurs chercheurs et Il est également facile.

A travers notre comparaison, nous avons constaté que malgré les avantages fournis par NS-2 tels que la facilité d'utilisation et la convivialité de ses interfaces, nous avons démontré qu'il présente plusieurs inconvénients liés à l'implémentation de certains de ses protocoles et à la simplification de quelques modèles, ce qui peut engendrer l'inexactitude de certains résultats. Ceci nous mène à mettre des réserves sur les outils de simulation d'une façon générale et par conséquent, sur tous les protocoles qui n'ont été validés que sur la base de simulations, et jamais par une implémentation réelle ou une preuve formelle. Les perspectives de ce travail seraient d'ajuster les incorrections qu'on a pu observer, à fin de faire converger les deux simulateurs vers des résultats proches de la réalité. Dans notre étude comparative nous nous somme basés sur des protocoles de réseaux sans fil Ad hoc, nous pourrons évaluer aussi les protocoles filaires supportés par NS-2 et déduire ainsi les performances de ce simulateur pour les réseaux filaires .

Référence

- [Article 01]** : Acticle " De la Simulation de Réseaux " Enoncés Michel Marto , Vincent Gauthier Télécim SudParis 2001
- [BAK 14]** : BABOURI Karima, thème « Les simulateurs réseaux Technologie réseau » 2013/2014 .
- [BRM 98]**: Bagrodia, R. Meyer et al., "PARSEC: A Parallel Simulation Environment for Complex Systems," IEEEComputer, October 98.
- [BSW 98]**: S. Basagni, I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward. "A distance routing effect algorithm for mobility (DREAM)". In Proceeding of ACM/IEEE MOBICOM'98, Dallas, Texas, pp 76-84, october 1998.
- [BTA 00]** : L. Bajaj, M. Takai, R. Ahuja, K. Tang, R. Bagrodia, M. Gerla, "GloMoSim : A Scalable Network Simulation Environment", computer Science Department, University of California at Los Angles, 2000.
- [CPB 94]**: Charles. E. Perkins and Pravin Bhagwat. "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computer". ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications, pp. 234-244, 1994.
- [CPR 00]**: Charles. E. Perkins, Elizabeth M. Royer and Samir R. Das. "Ad hoc on demand distance Vector (AODV) routing". IETF, Internet Draft, draft-ietf-manet-aodv-05.txt, March 2000.
- [CKT 96]**: Chai-Keong Toh. "A novel distributed routing protocol to support ad hoc mobile computing". Proceeding 1996 IEEE 15th Annual Int'l. Phoenix Conf. Comp. and Commun., pp 480-86, March 1996.
- [CWM 97]** :C. C. Chiang, H-K Wu, Winston Liu, and Mario Gerla. "Routing in clustered multihop, mobile wireless networks". The IEEE Singapore International Conference on Networks, pp 197-211, 1997.
- [DNR 92]**: D. Duchamp and N. F. Reynolds. "Measured performance of wireless LAN". Technical Report, Computer Science Department, Columbia University, NY, United States, September 1992.

- [**ERA 96**] : Pierre-Jean Erard, Pontien Déguénon « Simulation par évènements discrets » presses polytechnique et universitaire Romandes, 1996 1^{ière} édition .
- [**ELB 07**] : Dr. Elbiaze , Rapport de stage Concepteur Développeur en Ingénierie Logicielle eXia 2006-2007 « OBS et NS-2 » , : 2006-2007
- [**GTS 95**]: M. Gerla and J. Tsai. "Multicluster, mobile, multimedia radio network". ACM-Baltaz Journal of Wireless Networks, pp 255-265, 1995.
- [**HYZ 03**]: H. Luo, F. Ye, J. Cheng, S. Lu, and L. Zhang. TTDD : Two-tier data dissemination in largescalewireless sensor networks. ACM Journal of Mobile Networks and Applications (MONET),Special Issue on ACM MOBICOM (2003), 2003.
- [**IBR 94**]: Imienlinski T. and Badrinath B. R. "Mobile wireless computing: solutions and challenges in data management". CACM, 37(10), pp 18-28, October 1994.
- [**ICG 99**]: Iwata, C. C. Chiang, G. Pei, M. Gerla, and T.-W. Chen. "Scalable routing strategies for ad hoc wireless networks". IEEE Journal on Selected Areas in Communications, Special Issue on Ad-Hoc Networks, pp.1369-79, August 1999.
- [**ITB 94**]: Imienlinski T. and Badrinath B. R. "Mobile wireless computing: solutions and challenges in data management". CACM, 37(10), pp 18-28, October 1994.
- [**JNU 04**]: Jorge Nuevo "Comprehensible GloMoSim" Tutorial bcompilation by, INRS - Université du Québec nuevo@inrs-telecom.ubc.ca March 4, 2004
- [**JTL 99**]: M. Joa-Ng and I.-T.Lu "A Peer-to-Peer zone-based two-level link state routing for mobile ad hoc networks". IEEE Journal on Selected Areas in Communications, Special Issue on Ad-Hoc Networks, pp 1415-1425, August 1999.
- [**KAY 12**] : KHADIDJA AYAD Mémoire MAGISTER Thème "Sécurité du routage dans les réseaux ad hoc mobile" Option : Informatique Répartie et Mobile Présenté par : 14 Novembre 2012
- [**KAK 02**]: K. Fall and K. Varadhan, Eds., "The ns Manual, The VINT Project UC Berkeley, LBL, USC/ISI, and Xerox PARC," Apr. 2002, disponible au site : <http://www.isi.edu/nsnam/ns/>.

- [**KBO 12**] : Khaled BOUCHAKOUR, thème « Routage hiérarchique sur les réseaux de capteurs sans fil:Protocole KhLCH (K-hop Layered Clustering Hierarchy) ». Ecole Doctorale Sciences et Technologies de l'Information et de la Communication (STIC) Option : Système d'Information et de Connaissance (SIC). MEMOIRE Présenté pour l'obtention d'un diplôme de MAGISTER EN INFORMATIQUE.2012
- [**KXU 04**] : Addison Lee - Kaixin Xu. "GloMoSim Java Visualization Tool". Documentation version 1.1, SoftwareDistribution.2004
- [**KFV 02**] : K. Fall and K. Varadhan, Eds., "The ns Manual, The VINT Project UC Berkeley, LBL, USC/ISI, and Xerox PARC," Apr. 2002, disponible au site : <http://www.isi.edu/nsnam/ns/>.
- [**LMK 00**] : Law, A. M., et W. D. Kelton. 2000. Simulation Modeling and Analysis. 3ed. McGraw-Hill.
- [**MJT 99**] : Mingliang Jiang, Jinyang Li, Y.C. Tay. "Cluster based routing protocol", IETF Draft, 27 pages. <http://www.ietf.org/internet-drafts/draft-ietf-manet-cbrp-spec-01.txt>. August 1999.
- [**NBA 98**] : NadjibBadache. "La mobilité dans les systèmes répartis". 45 pages, Janvier 1998.
- [**PAD 04**] : Priya Mahadevan, Adolfo Rodriguezn, David Becker, Amin Vahdat, « MobiNet: A Scalable Emulation Infrastructure for Ad Hoc and Wireless Networks », UC, San Diego, 2004.
- [**PEB 93**] : Pitoura E. and Bhargava B. "Dealing with mobility". Issue and Research Challenges. Technical Report CSD-TR-93-070, Department of computer science, Purdue University, November 1993.
- [**PEH 99**] : P. Anelli & E. Horlait : *NS-2: Principes de conception et d'utilisation*, Version 1.3
- [**RBA 96**] : Rajive Bagrodia. - GloMoSim Software. University of California, Los Angeles-De-partment of Computer Science. Box 951596, 3532 Boelter Hall, Los Angeles-CA 90095-1596
- [**RDU 97**] : R. Dube et al. "Signal stability based adaptive routing for ad hoc mobile networks". IEEE Pers. Comm., pp 36-45, February 1997.
- [**SGH 95**] : S. G. Hild. "A brief history of mobile telephony". Technical Report 372, Computer Laboratory, University of Cambridge, England, 1995.

- [SMG 95]: Shree Murthy and J. J. Garcia-Luna-Aceves. "A routing protocol for packet radio networks". Proceeding of the IEEE Mobicom, pp 86-95, November 1995.
- [TAB 12]: TAHAR ABBES Mémoire thème "Proposition d'un protocole à économie d'énergie dans un réseaux hybride GSM et AD HOC Mounir .Universté d'oran , faculté des sciences spécialité informatique 2012
- [TAL 09]:TRAN Alexandre & LAYOUNI Majid « Simulator Network 2 NS-2 » Session 2009
- [TLE 00]: Tayeb LEMLOUMA MINI PROJET Le Routage dans les Réseaux Mobiles Ad Hoc . Présenté par, Septembre 2000 .
- [VDS 99]: V. D. Park and M. S. Corson. "Temporally-ordered routing algorithm (TORA) Version 1, fonctionnal specification". IETF, Internet Draft, draft-ietf-manet-tora-spec-02.txt, October 1999.
- [VPC 99]: V. D. Park and M. S. Corson. "Temporally-ordered routing algorithm (TORA) Version 1, fonctionnal specification". IETF, Internet Draft, draft-ietf-manet-tora-spec-02.txt, October 1999.
- [WYP 93]: Wayner P. "Stretching the Ether". Byte, February 1993 .
- [YMZ 11]: Yassine SADQI , Mohamed ZAOUI Mémoire Mester informati des Systèmes Répartis (MISR) them "Les PROTOCOLES DE ROUTAGE MIS EN PLACE DANS LE CADER DE RESEAUX AD HOC MOBILES (CARACTERISTIQUES , COMPARAISON) Année universitaire : 2010 / 2011 cite: <http://fr.slideshare.net/zaouimed/routage-adhoc>

Les sites web

[Site 1] <http://pcl.cs.ucla.edu/projects/glomosim/>

[Site 2] <http://www-igm.univ-mlv.fr/~badis/IR3/ns2/>

[Site 3] <http://igm.univ-mlv.fr/~dr/XPOSE2002/Sansfils/index.php?rubrique=Principe>

[Site 4] http://nslam.isi.edu/nslam/index.php/User_Information

[Site 5] Tutorial for the Network Simulator NS, <http://www.isi>.

[Site6] http://www.memoireonline.com/12/07/738/m_effets-mobilite-protocoles-routage-reseaux-ad-hoc29.html