

République Algérienne Démocratique et Populaire
Ministère de L'enseignement Supérieur et de la Recherche Scientifique
Université A/Mira de Bejaïa
Faculté des Sciences Exactes
Département Informatique



Mémoire de Fin de Cycle

En vue de l'obtention du diplôme Master en Informatique
Option : Administration et Sécurité des Réseaux

Thème

*Adaptation du simulateur multi agents Netlogo pour
modéliser et simuler un réseau mobile Ad hoc,
notamment les deux couches PHY/MAC d'IEEE 802.11*

Réalisé par :

Mr BOUCHEBBAH Aghilas

Mr HAMOUCHE Islam

Membre du jury :

Présidente	Mme BOUADEM Nassima	M.C.B	U. A/Mira Bejaia
Encadrant	Mme HOUHA Amel	M.A.A	U. A/Mira Bejaia
Examineur	Mr MOKTEFI Mohand	M.A.A	U. A/Mira Bejaia

Promotion 2019-2020

Remerciements

Tout d'abord, nous devons remercier le Dieu le tout puissant de nous avoir donné tout le courage et la santé pour achever cet humble travail.

Nous tenons à remercier notre encadrant Mme HOUHA Amel, de nous avoir proposé cet intéressant sujet et d'avoir accepté de nous encadrer tout au long de ce travail, notamment pour ces conseils précieux, pour sa disponibilité malgré la situation exceptionnelle que nous vivons, qu'elle trouve ici notre reconnaissance et notre respect.

Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont portés à notre recherche en acceptant d'examiner notre travail.

Enfin, un grand merci à nos familles, surtout nos parents pour leur encouragement et leur suivi avec patience tout au long de ce travail.

Tables des matières

Table des matières	i
Liste des figures	iv
Liste des tableaux	vi
Introduction générale	1
1 CHAPITRE 1 SYSTEMES MULTI AGENT POUR LES RESEAUX MANET.....	2
1.1 INTRODUCTION.....	2
1.2 AGENT ET SYSTEMES MULTI-AGENTS	2
1.2.1 Agent.....	3
1.2.1.1 Définition d'un agent.....	3
1.2.1.2 Caractéristiques d'un agent.....	3
1.2.1.3 Types d'agent.....	4
1.2.2 Système multi agents (SMA).....	6
1.2.2.1 Définition d'un SMA.....	6
1.2.2.2 Environnement d'un SMA.....	7
1.2.2.3 Classification du système multi agents	8
1.2.2.4 Caractéristiques d'un SMA	13
1.2.2.5 Notions importantes dans SMA.....	13
1.2.2.6 Avantages d'un système multi agents.....	14
1.2.2.7 Défis des systèmes multi-Agents.....	14
1.2.2.8 Domaines d'application SMA	16
1.3 RESEAUX INFORMATIQUE A BASE DES SYSTEMES MULTI AGENTS.....	16
1.3.1 Réseaux mobile ad-hoc (MANET).....	17
1.3.2 SMA pour les réseaux MANET.....	17
1.3.3 SMA et le routage dans les réseaux MANET.....	18
1.4 CONCLUSION.....	18
2 CHAPITRE 2 GENERALITES SUR LES RESEAUX AD HOC.....	19
2.1 INTRODUCTION.....	19
2.2 RESEAUX SANS FILS.....	19
2.2.1 Classification des réseaux sans fils.....	20
2.3 RESEAUX AD HOC.....	22
2.3.1 Définition d'un réseau Ad hoc.....	22
2.3.2 Rappels sur la topologie d'un réseau	23

2.3.3	<i>Pile protocolaire dans les réseaux Ad hoc</i>	24
2.3.4	<i>Caractéristiques et contraintes des réseaux ad hoc</i>	25
2.3.5	<i>Avantages des réseaux Ad hoc</i>	26
2.3.6	<i>Inconvénients des réseaux Ad hoc</i>	27
2.3.7	<i>Domaines d'utilisation des réseaux Ad hoc</i>	27
2.3.8	<i>Technologies de mise en réseau ad hoc</i>	29
2.3.9	<i>Protocoles de contrôle d'accès (MAC) pour les réseaux Ad hoc</i>	32
2.3.9.1	Objectifs du protocole MAC dans les réseaux Ad hoc.....	32
2.3.9.2	Classification des protocoles MAC	32
2.4	CONCLUSION.....	34
3	CHAPITRE 3 PRESENTATION DU STANDARD IEEE 802.11	35
3.1	INTRODUCTION.....	35
3.2	NORME IEEE 802.11	35
3.2.1	<i>Evolutions du 802.11</i>	36
3.2.2	<i>Couche physique PHY d'IEEE 802.11</i>	37
3.2.2.1	Les grandeurs physiques des ondes	37
3.2.2.2	Les règles de la transmission radio.....	39
3.2.2.3	Les différentes techniques de modulation.....	42
3.2.2.4	Les trames 802.11.....	46
3.2.3	<i>Couche de contrôle d'accès au medium (MAC) 802.11</i>	47
3.2.3.1	Méthodes d'accès au médium sans fil IEEE 802.11.....	47
3.3	CONCLUSION.....	54
4	CHAPITRE 4 SIMULATION ET SIMULATEURS RESEAUX	55
4.1	INTRODUCTION.....	55
4.2	LA SIMULATION INFORMATIQUE	55
4.2.1	<i>Définition de la simulation</i>	55
4.2.2	<i>Types de la simulation</i>	56
4.2.3	<i>Intérêts de la simulation</i>	57
4.2.4	<i>Définition de la simulation d'un réseau</i>	57
4.2.5	<i>Contraintes et défis de la simulation dans les réseaux sans fil</i>	58
4.2.6	<i>Comparaison entre les simulateurs traditionnels des réseaux sans fils</i>	58
4.2.6.1	NS2.....	58
4.2.6.2	NS3.....	59
4.2.6.3	OMNET++.....	60
4.3	SIMULATION A BASE D'AGENTS	62
4.3.1	<i>Modélisation et la simulation (M&S) basées sur des agents</i>	62
4.3.2	<i>Structure d'un modèle basé sur un agent</i>	62
4.3.3	<i>Conception du modèle d'agent</i>	63
4.3.4	<i>Outils basés sur l'agent</i>	64
4.3.4.1	NETLOGO	64
4.3.4.2	Pourquoi le SMA NETLOGO ?	65
4.4	CONCLUSION.....	66

5	CHAPITRE 5 SIMULATION D'UN RESEAU MANET SUR NETLOGO	67
5.1	INTRODUCTION.....	67
5.2	PRESENTATION DU MODEL SIMULE	67
5.2.1	<i>Objectifs du modèle</i>	<i>67</i>
5.2.1.1	Conception du modèle simulé.....	68
5.2.1.2	Paramètres de la transmission radio simulés.....	68
5.2.1.3	Protocoles simulé dans le model.....	69
5.3	NETLOGO	72
5.3.1	<i>Concepts de programmation en NetLogo.....</i>	<i>72</i>
5.3.2	<i>L'interface graphique de Netlogo</i>	<i>76</i>
5.4	LA MISE EN ŒUVRE DU MODELE	77
5.4.1	<i>Création et initialisation de l'environnement</i>	<i>77</i>
5.4.1.1	Code.....	77
5.4.1.2	Interface.....	87
5.4.2	<i>Envoie de message simple.....</i>	<i>89</i>
5.4.3	<i>Envoie de messages multiples avec CSMA/CA</i>	<i>91</i>
5.5	CONCLUSION.....	93
	Conlusion et perspectives	94
	Bibliographie	95
	References webographique	99
	Resumé	100

Liste des figures

- Figure 1.1** Agent et système multi-agents [1]
Figure 1.2 Les propriétés des agents selon leurs types [2]
Figure 1.3 Agent réactif [1]
Figure 1.3 Agent cognitif [1]
Figure 1.4 Une architecture d'agent hiérarchique [6]
Figure 1.5 Holons imbriqués ressemblant au SMA hiérarchique [6]
Figure 1.6 Architecture multi-agents de coalition [6]
Figure 1.7 Architecture multi-agents basée sur les équipes [6]
Figure 1.8 Communication à base de tableau noir [2]
Figure 1.9 Communication entre agents [2]
Figure 1.10 Réseau Ad hoc et SMA [10]
- Figure 2.1** Réseau sans fils avec infrastructure [16]
Figure 2.2 Réseau sans fils sans infrastructure Ad hoc [17]
Figure 2.3 Topologies de réseau
Figure 2.4 Différentes utilisations des réseaux Ad hoc [21]
Figure 2.5 Classification des protocoles MAC [18]
- Figure 3.1** Architecture en couche de la norme IEEE 802.11 [26].
Figure 3.2 Spectre électromagnétique : le WiFi repose sur les micro-ondes.
Figure 3.3 Une onde et ses grandeurs physiques [30]
Figure 3.4 Chemins multiples (multipath) [27].
Figure 3.5 Débit théorique maximal du signal en fonction de la distance [27]
Figure 3.6 Zone de portée [31].
Figure 3.7 Modulateur I/Q : utilisé pour les modulations dans 802.11 [28]
Figure 3.8 Couches physiques du WiFi : débit, fréquence et compatibilité [27].
Figure 3.9 Exemple de partage des ondes avec le FHSS [27].
Figure 3.10 Etalement de spectre sur une porteuse [29]
Figure 3.11 Canaux définis par OFDM [29]
Figure 3.12 Format général d'une trame 802.11
Figure 3.13 Format d'une trame 802.11 [29].
Figure 3.14 Organigramme du protocole CSMA/CA [29]
Figure 3.15 Déroulement du protocole CSMA/CA [29]

- Figure 3.16** Relation entre les différentes unités temporelles IFS [28]
- Figure 3.17** Evolution de la Contention Window CW selon le BEB [29].
- Figure 3.18** Mécanisme d'accès RTS/CTS [28].
- Figure 3.19** Problème de la station cachée [22].
- Figure 3.20** Problème de la station exposée [18]
-
- Figure 4.1** Etapes de conception d'une simulation informatique [32]
- Figure 4.2** Structure d'un modèle basé sur un agent [40]
-
- Figure 5.1** Changement de topologie dans réseau MANET
- Figure 5.2** Djikstra : le plus court chemin
- Figure 5.3** Composants de la fenetre Netlogo [44]
- Figure 5.4** Code d'initialisation de la simulation
- Figure 5.5** Code de mobilité
- Figure 5.6** Code d'ondes radio
- Figure 5.7** Code génération messages multiples pour CSMA/CA
- Figure 5.8** Code CSMA/CA et Routage
- Figure 5.9** Interface de la simulation
- Figure 5.10** Visualisation de la portée radio
- Figure 5.11** Envoi simple à un saut
- Figure 5.12** Envoi simple à sauts multiple (routage)
- Figure 5.13** Envoi de messages avec CSMA-CA
- Figure 5.14** Commentaires de sortie détaillant la simulation

Liste des tableaux

Tableau 1.1 Comparaison entre agent cognitif et agent réactif

Tableau 2.1 Tableau comparatif entre réseau cellulaire et Ad hoc [18]

Tableau 2.2 La structure en couches des réseaux Ad hoc par rapport à OSI et TCP/IP

Tableau 4.1 Tableau comparatif entre les simulateurs réseau traditionnels.

Introduction générale

La modélisation et la simulation (M&S) jouent un rôle essentiel dans la conception et le développement de systèmes interactifs distribués en raison de leur nature stochastique particulière, en particulier s'ils impliquent des systèmes dotés de capacités d'auto-organisation décentralisées. Bien que les réseaux Ad hoc aient augmenté à la fois en taille et en complexité, les outils traditionnels pour les modéliser et les simuler n'ont pas évolué au même rythme.

De ce fait, il existe un besoin d'outils alternatifs flexibles équipés de techniques efficaces pour modéliser les réseaux modernes à mesure qu'ils évoluent vers des systèmes complexes. L'évolution d'un certain nombre de domaines technologiques à l'image des systèmes multi-agents (SMA), peut permettre une résolution de problèmes autonome et basée sur l'équipe dans des conditions environnementales dynamiques et mobiles tel dans les réseaux MANET. Le fonctionnement combiné de ces deux domaines technologiques, a été largement inexploré à ce jour.

Ce travail cherche à aborder cet important domaine de recherche en démontrant l'utilisation d'outils de modélisation et de simulation basés sur des agents pour modéliser et simuler les aspects liés aux réseaux mobiles Ad hoc (architectures décentralisées, auto-organisation et mobilité des nœuds, transmission radio, l'accès au canal, routage...).

Pour faire, nous nous concentrons sur l'outil NetLogo, un simulateur qui a fait ses preuves dans le domaine de la modélisation et de la simulation de systèmes complexes et multi-agents, dans le but est de modéliser et simuler les réseaux ad hoc. Nous utilisons NetLogo pour simuler et évaluer en particulier les protocoles des deux couches Physique et contrôle d'Accès au médium MAC utilisées pour les réseaux MANET, et le routage qui assure l'acheminement des paquets dans le réseau d'une manière générale.

Mot-clés : Système multi-agents, MANET, Auto-organisation, simulation et modélisation, Netlogo.

Chapitre 1 *Systèmes multi agent pour les réseaux MANET*

1.1 Introduction

Les systèmes multi-agents (SMA) sont actuellement l'un des domaines les plus actifs de la recherche avec une assise applicative de plus en plus importante. Cette discipline, joignant plusieurs domaines informatique et autres, s'intéresse aux comportements collectifs produits par les interactions de plusieurs entités autonomes appelées agents. Elle permet d'offrir une alternative intéressante pour la conception, la mise en œuvre et la simulation de systèmes complexes distribués [1]. Dans ce chapitre, nous serons menées à étudier les systèmes multi-agents et comprendre le comportement inter-agent soit en coopération ou en compétition, pour ensuite expliquer leur utilisation dans le domaine des réseaux ad hoc.

1.2 Agent et Systèmes multi-Agents

Un système multi-agents (SMA) est une extension de la technologie des agents dans laquelle un groupe d'agents autonomes faiblement connectés qui communiquent entre eux et agissent dans un environnement pour atteindre un objectif commun et résoudre un problème qui dépasse le cadre d'un seul agent. Cela se fait soit en coopérant ou en compétition, en partageant ou en ne partageant pas les connaissances les uns avec les autres.

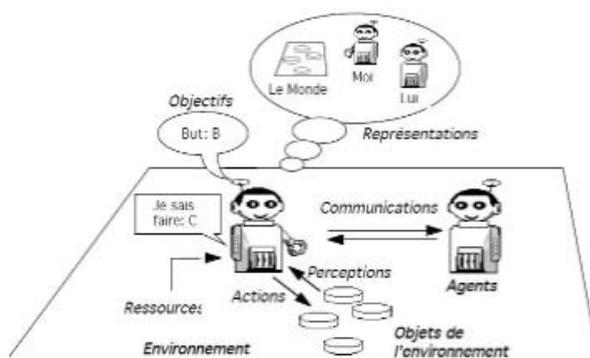


Figure 1.1 Agent et système multi-agents [1]

1.2.1 Agent

L'agent représente sans doute la brique de base utilisée pour la réalisation des SMA. En plus, nous pouvons considérer l'agent comme la notion la plus polémique dans ce domaine.

1.2.1.1 Définition d'un agent

On appelle agent une entité réelle ou virtuelle plongée dans un environnement sur lequel elle est :

- Capable d'agir
- Qui peut communiquer directement avec d'autres agents
- Qui est mue par un ensemble de tendances (sous la forme d'objectifs individuels ou d'une fonction de satisfaction, voire de survie, qu'elle cherche à optimiser)
- Qui possède des ressources propres.
- Qui est capable de percevoir (mais de manière limitée) son environnement.
- Qui ne dispose que d'une représentation partielle de cet environnement (éventuellement aucune).
- Qui possède des compétences et offre des services.
- Qui peut éventuellement se reproduire.
- Dont le comportement tend à satisfaire ses objectifs, en tenant compte des ressources et des compétences dont elle dispose, et en fonction de sa perception, de ses représentations et des communications qu'elle reçoit [3].

1.2.1.2 Caractéristiques d'un agent

- ✓ **Autonomie** : les agents peuvent fonctionner sans aucun type d'incorporation ou d'intervention par d'autres programmes ou utilisateurs.
- ✓ **Réactivité** : les agents sont capables de réagir aux changements de l'environnement en fonction de leur objectif.
- ✓ **Capacité sociale** : les agents du système interagissent entre eux et éventuellement avec les utilisateurs. Ces interactions sont réalisées dans le but de renforcer la coopération et la concurrence, et se font par le partage d'information ou par l'envoi de messages.

Selon le domaine d'étude, certaines caractéristiques s'ajoutent à celles qui ont été présentées ci-dessus :

- ✓ **Proactivité** : les agents sont capables d'initier des actions ainsi que de réagir aux changements de l'environnement. Ces actions sont effectuées afin d'atteindre un objectif particulier.
- ✓ **Adaptation et apprentissage** : les agents du système peuvent s'améliorer en apprenant et en s'adaptant à leur environnement.
- ✓ **Mobilité** : les agents peuvent se déplacer entre les nœuds d'un réseau.

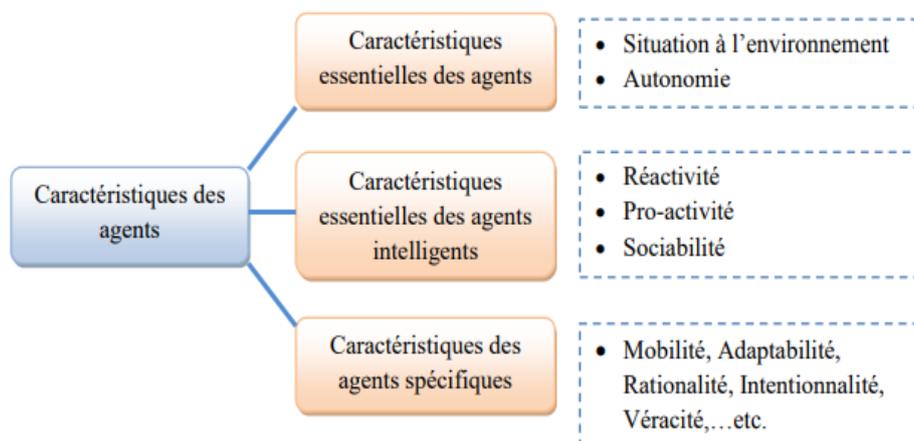


Figure 1.2 Les propriétés des agents selon leurs types [2]

1.2.1.3 Types d'agent

Selon le degré de couplage de l'agent à l'environnement on peut considérer trois catégories d'agent : réactif, cognitif et hybride :

- **Agent réactif**

Un agent réactif ne fait que réagir aux changements qui surviennent dans l'environnement. Il se contente simplement d'acquiescer des perceptions et de réagir à celles-ci en appliquant certaines règles prédéfinies. Etant donné qu'il n'y a pratiquement pas de raisonnement, ces agents peuvent agir et réagir très rapidement [4].

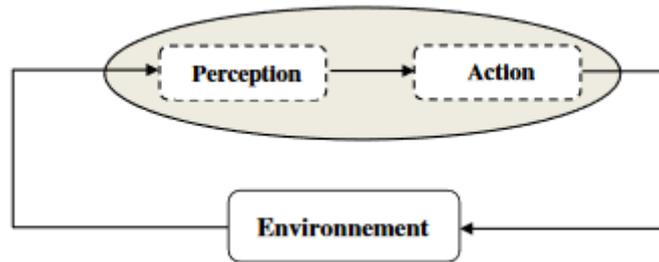


Figure 1.3 Agent réactif [1]

- **Agent cognitif**

Ces agents disposent des capacités fondamentales de perception, délibération et action, couplés à des concepts nouveaux de représentation et de régulation des processus comportementaux [4].

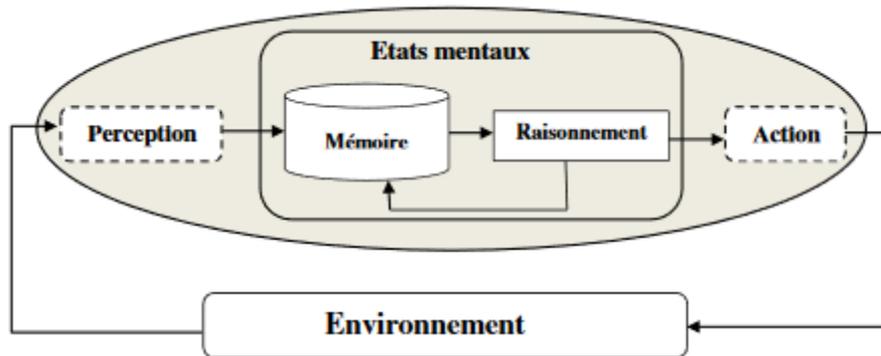


Figure 1.3 agent cognitif [1]

Agent cognitif	Agent réactif
<ul style="list-style-type: none"> • Représentation explicite de l'environnement • Peut tenir compte de son passé • Comportement avec décision délibérative • Petit nombre d'agent • Modèle de communication directe par message. 	<ul style="list-style-type: none"> • Pas de représentation explicite • Pas de mémoire de son histoire • Comportement réflexe (Stimulus/Action) • Grand nombre d'agents • Communication principalement via l'environnement, et/ou au travers de signaux.

Tableau 1.1 Comparaison entre agent cognitif et agent réactif

- **Agent hybrid**

La division des agents en deux classes ne représente qu'une vue idéaliste. En fait, ces deux classes ne présentent que les cas extrêmes. En réalité, un agent nécessite des capacités réactives et des capacités cognitives. Ainsi, il est important de combiner les deux approches citées ci-dessus.

Un agent hybride est un agent conçu en couches. Les couches inférieures assurent des comportements réactifs. Par contre, les couches supérieures sont responsables de capacités cognitives complexes comme les aspects sociaux. Bien entendu, les couches supérieures manipulent des connaissances alors que les couches inférieures manipulent directement des données. En conséquence, des couches intermédiaires sont responsables de transformer des données en connaissances [2].

1.2.2 Système multi agents (SMA)

La section précédente a présenté des systèmes où il n'y avait qu'un seul agent mais, dans la plupart des situations réelles, l'agent n'est pas seul dans son environnement, il y a d'autres agents présents autour de lui. Il nous faut donc aborder des systèmes où plusieurs agents doivent interagir entre eux pour effectuer leurs tâches. De tels systèmes sont appelés « systèmes multi-agents ».

1.2.2.1 Définition d'un SMA

Un Système multi-agent (SMA) est un ensemble des agents travaillent pour résoudre un problème commun, les possibilités de chaque agent prises séparément ne permettent pas de le résoudre. Chaque agent possède donc des connaissances et un savoir-faire limités, ce qui l'oblige à interagir avec d'autres pour mener à bien le projet commun. Les SMA mettent en œuvre des interactions complexes, comme la coopération, la coordination, la collaboration, la compétition, et la négociation [5], [4] :

- Il y a **coopération** lorsque les agents travaillent ensemble pour atteindre un but commun.
- **La coordination** consiste à organiser les activités liées à la résolution d'un problème de manière à éviter les interactions néfastes et à exploiter les interactions bénéfiques.

- La **collaboration** : Les agents du système partagent un même but de façon intermittente.
- La **compétition** : Les agents du système ont des buts incompatibles
- La **négociation** a pour objectif la gestion des conflits entre agents, c'est à dire qu'elle consiste à aboutir à un accord acceptable par l'ensemble des agents impliqués.

1.2.2.2 Environnement d'un SMA

L'environnement est un élément important dans le système multi-agents. C'est grâce à lui que les agents peuvent coexister et interagir. L'environnement doit pouvoir être perçu par les agents et ces derniers doivent pouvoir agir dessus et interagir au travers. Avec celle des comportements individuels, la spécification de l'environnement permet de définir la dynamique d'un SMA [1].

1) Types d'environnement

Selon le point de vue que l'on adopte, on peut identifier différents types d'environnements :

- **Point de vue du système multi-agents** : l'environnement correspond à l'ensemble des entités extérieures au système.
- **Point de vue de l'agent** : l'environnement est tout ce qui est extérieur à lui-même.
- **Point de vue du concepteur** : il peut correspondre à l'état du système, ou représenter l'ensemble des outils permettant de simuler, de visualiser et d'évaluer le SMA.

2) Propriétés d'environnement

L'environnement possède certaines propriétés

- **Accessible ou inaccessible** : un agent a accès à l'état complet de l'environnement ou non.
- **Déterministe ou indéterministe** : le changement de l'état de l'environnement est uniquement déterminé par l'état courant et les actions des agents ou non.
- **Statique ou dynamique** : l'environnement peut changer quand l'agent est en action (réflexion) ou non.
- **Discret ou continu** : le nombre de perceptions et d'actions est limité ou pas.

1.2.2.3 Classification du système multi agents

La classification du SMA est une tâche difficile car elle peut être effectuée sur plusieurs différents attributs tels que : l'architecture, l'organisation, Coordination, communication ... [6].

A) Architecture

Sur la base de l'architecture interne des agents individuels particuliers formant le système multi-agents, il peut être classé en deux types :

- **Structure homogène**

Dans une architecture homogène, tous les agents formant le système multi-agents ont la même architecture interne. La différence entre les agents est leur emplacement physique et la partie de l'environnement où l'action est faite. Chaque agent reçoit une entrée de différentes parties de l'environnement. Il peut y avoir un chevauchement dans les entrées de capteur reçues.

- **Structure hétérogène**

Dans une architecture hétérogène, les agents peuvent différer en termes de capacité, de structure et fonctionnalité. Sur la base de la dynamique de l'environnement et de la localisation de l'agent particulier, les actions choisies par un agent peuvent différer de l'agent situé dans une partie différente où Chaque agent peut avoir différents objectifs locaux pouvant aller à l'encontre de l'objectif d'autres agents [6].

B) Organisation des agents

On peut distinguer plusieurs types d'organisation des agents dans un système multi agents [6]

- **Organisation hiérarchique**

Organisation hiérarchique est l'une des premières conceptions organisationnelles de systèmes multi-agents. L'architecture hiérarchique a été appliquée à un grand nombre de problèmes distribués. Dans l'architecture hiérarchique des agents, les données des niveaux inférieurs de la hiérarchie sont généralement flux vers le haut pour les agents avec une hiérarchie supérieure.

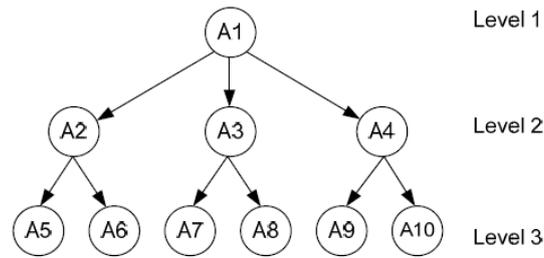


Figure 1.4 Une architecture d'agent hiérarchique [6]

- **Organisation des agents holoniques**

Un 'Holon' est une structure similaire, stable et cohérente composée de plusieurs "holons" en tant que sous-structure et fait elle-même partie d'un cadre plus vaste.

Dans un système multi-agents holonique, un agent qui apparaît comme une entité unique peut être composé de nombreux sous-agents liés ensemble. Chaque holon désigne ou sélectionne un agent principal capable de communiquer avec l'environnement ou avec d'autres agents situés dans l'environnement. La sélection du L'agent principal est généralement basé sur la disponibilité des ressources, la capacité de communication et l'architecture interne de chaque agent. Dans un système multi-agents homogène, la sélection peut être aléatoire et une politique de rotation pourrait être utilisée similaire à celle utilisée avec des réseaux de capteurs sans fil distribués.

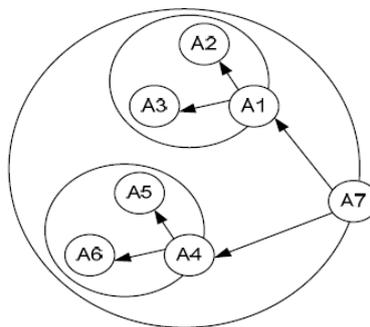


Figure 1.5 Holons imbriqués ressemblant au SMA hiérarchique [6]

- **Organisation de Coalitions**

Dans l'architecture de coalition, un groupe d'agents se réunit pendant une courte période pour augmenter l'utilité ou performance des agents individuels d'un groupe. La coalition cesse d'exister lorsque l'objectif de performance est atteint. La figure 3 montre un multi agent de coalition typique.

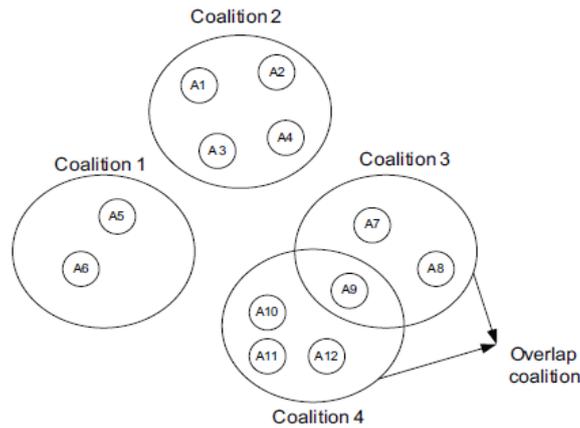


Figure 1.6 Architecture multi-agents de coalition [6]

Théoriquement, former un seul groupe constitué de tous les agents de l'environnement optimisera les performances du système. C'est parce que chaque agent a accès à toutes les informations et ressources nécessaires pour calculer la condition pour une action optimale. Il est impossible de former une telle coalition en raison des contraintes imposées à la communication et ressources.

- **Les équipes**

L'architecture équipes SMA est similaire à l'architecture de coalition dans la conception, sauf que les agents d'une équipe travaillent ensemble pour augmenter la performance globale du groupe plutôt que de travailler chacun en tant qu'agents individuels. Les interactions des agents au sein d'une équipe peuvent être assez arbitraires et les objectifs ou les rôles assignés à chacun des agents peut varier avec le temps en fonction des améliorations résultant des performances de l'équipe.

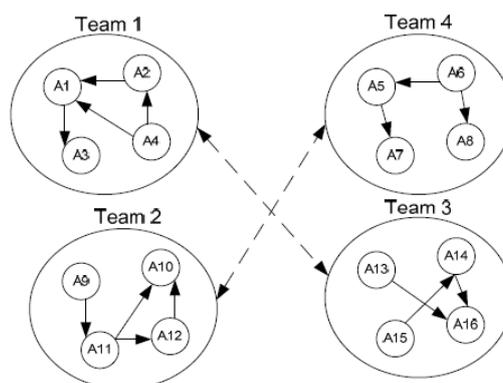


Figure 1.7 Architecture multi-agents basée sur les équipes [6]

C) Coordination dans un système multi-agents

La coordination est la question centrale dans la conception des systèmes multi-agents. Les agents sont des systèmes rarement autonomes et impliquent généralement plus d'un agent travaillant parallèlement pour atteindre un objectif commun. Lorsque plusieurs agents sont employés pour obtenir un objectif, il est nécessaire de coordonner ou de synchroniser les actions pour assurer la stabilité du système. La coordination entre les agents augmente les chances d'atteindre une solution globale optimale [6].

- **Coordination par protocole**

Une technique de coordination classique entre les agents d'une architecture distribuée consiste au protocole de communication. Le protocole est généralement en langage de haut niveau qui spécifie la méthode de coordination entre les agents et consiste en une série de tâches et méthodes d'allocation de ressources.

Chaque agent assume le rôle de gestionnaire et d'entrepreneur selon les besoins. Le gestionnaire sert essentiellement à casser un problème plus important en sous-problèmes plus petits et trouve des sous-traitants capables d'exécuter efficacement ces fonctions. Un entrepreneur peut devenir un gestionnaire et décomposer le sous-problème afin de réduire le coût de calcul et accroître l'efficacité.

- **Coordination à travers des modèles de croyance**

Dans les scénarios où le temps est d'une importance cruciale, la coordination via les protocoles échoue lorsqu'un agent disposant d'une ressource spécifique pour résoudre le sous-problème rejette le problème offert. Dans de tels scénarios, les agents ayant un modèle de croyance interne des agents voisins pourraient résoudre le problème. Le modèle de croyance interne pourrait être soit développé par observer la variation de la dynamique de l'environnement ou développé sur la base de connaissance heuristique et expertise du domaine.

D) La communication

La communication en effet agrandit les capacités perceptives des agents en leur permettant de bénéficier des informations et du savoir-faire des autres agents. Les communications sont indispensables à la coopération et il est difficile de concevoir un système d'agents coopérants

s'il n'existe pas un système permettant aux agents d'échanger des informations ou de véhiculer des requêtes. Dans les systèmes cognitifs, les communications s'effectuent par envoi de messages, alors que dans les systèmes réactifs, elles résultent de la diffusion d'un signal dans l'environnement. La communication constitue l'un des moyens fondamentaux pour assurer la répartition des tâches et la coordination des actions [7]. On peut distinguer deux types de communication dans un SMA :

- **Communication indirect** (système de contrôle centralisé) :

Ce type d'interaction est caractérisé par l'absence de communication directe entre les agents, mais par l'utilisation d'une zone commune qui est le tableau noir: une zone de travail commune, dévolue à la transition d'informations entre les différents agents.

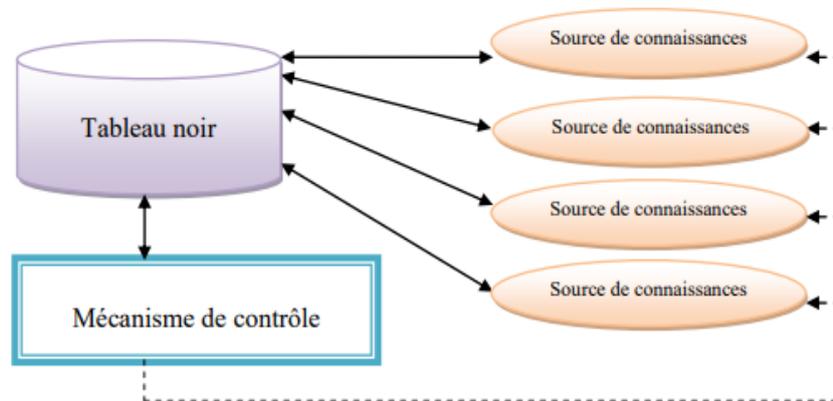


Figure 1.8 Communication à base de tableau noir [2]

- **Communication direct** (système de contrôle distribué) :

La communication se fait directement entre les agents par envoi de messages sans aucun intermédiaire, soit en mode point à point ou en mode diffusion. Elle crée une architecture distribuée et permet d'éviter le dysfonctionnement causé par la défaillance des agents centraux.

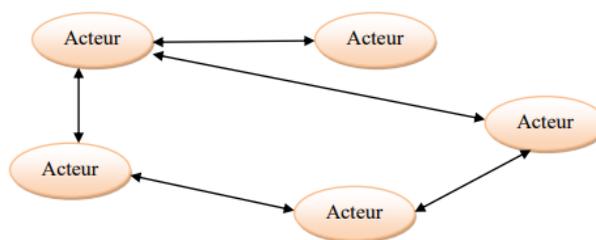


Figure 1.9 Communication entre agents [2]

La communication entre agents présente les avantages suivants par rapport à la communication traditionnelle basée sur le serveur client (RPC) :

- Communication Peer-Peer décentralisée, contrairement aux rôles traditionnels client-serveur.
- Échange asynchrone de messages.

1.2.2.4 Caractéristiques d'un SMA

Un SMA possède la plupart des caractéristiques suivantes :

- **Distribution:** le système est décomposable, l'élément de base étant l'agent.
- **Décentralisation :** les agents sont indépendants, il n'y a pas de décisions centrales valables pour tout le système.
- **Autonomie:** un agent est en activité permanente et prend ses propres décisions en fonction de ses objectifs et de ses connaissances.
- **Echange de connaissances :** les agents sont capables de communiquer entre eux, selon des langages plus ou moins élaborés.
- **Interaction :** les agents ont une influence localement sur le comportement des autres agents.
- **Organisation :** les interactions créent des relations entre les agents, et le réseau de ces relations forme une organisation qui peut évoluer au cours du temps [1].

1.2.2.5 Notions importantes dans SMA

1. Auto organisation

Il existe plusieurs définitions d'auto-organisation, citons :

- Motif ou fonction du niveau global d'un système (SMA) qui émerge suite aux interactions entre composants de plus bas niveau (agents).
- Transformation autonome de la topologie d'un système (exp : connexions réseau) par ses composants résultant du fonctionnement de ce réseau.
- Processus par lequel un système change son organisation interne pour s'adapter aux changements de ses buts et de l'environnement sans contrôle externe explicite. L'auto-organisation résulte souvent en un comportement émergent désirable ou non.

- Le système doit pouvoir s'adapter à son environnement
- Auto-organisation des composants ➡ adaptation [8].

2. Adaptation

Un système multi-agent adaptatif est donc défini comme un système multi-agent qui est capable de changer son comportement en cours de fonctionnement pour l'ajuster dans un environnement dynamique, soit pour réaliser la tâche pour laquelle il a été conçu, soit pour améliorer sa fonction ou ses performances. Il est caractérisé par le fait d'être plongé dans un environnement dynamique, de réaliser une tâche (fonction) et d'être composé d'agents en interaction [8].

1.2.2.6 Avantages d'un système multi agents

Les systèmes multi-agents ont été largement adoptés dans de nombreux domaines d'application en raison des avantages qu'ils offrent. Parmi les avantages offerts par l'utilisation de la technologie SMA dans les grands systèmes, citons :

- Une augmentation de la vitesse et de l'efficacité de l'opération due au calcul parallèle et au fonctionnement asynchrone.
- Une dégradation progressive du système lorsqu'un ou plusieurs agents échouent. Cela augmente ainsi la fiabilité et la robustesse du système.
- Évolutivité et flexibilité : des agents peuvent être ajoutés en fonction des besoins.
- Coût réduit : les agents individuels coûtent beaucoup moins cher qu'une architecture centralisée.
- Les agents de réutilisation utilisables à structure modulaire et ils peuvent être facilement remplacés dans d'autres systèmes ou être mis à niveau plus facilement qu'un système monolithique [6].

1.2.2.7 Défis des systèmes multi-Agents

Bien que les systèmes multi-agents possèdent des fonctionnalités plus utiles que les systèmes à agent unique, ils présentent également des défis critiques. Certains des défis sont mis en évidence dans ce qui suit :

1. Environnement

Dans un système multi-agents, l'action d'un agent modifie non seulement son propre environnement, mais également celui de ses voisins. Cela nécessite que chaque agent prédise l'action des autres agents afin de décider de l'action optimale à cibler. Ce type d'apprentissage simultané peut entraîner un comportement instable et éventuellement causer le chaos. Le problème se complique davantage si l'environnement est dynamique. Ensuite, chaque agent doit différencier les effets dus aux autres actions d'autres agents et les variations de l'environnement lui-même.

2. Perception

Dans un système multi-agents distribué, les agents sont dispersés dans l'environnement. Chaque agent a une capacité de détection limitée en raison de la portée et de la couverture des capteurs qui lui sont connectés. Cela limite la vue disponible pour chacun des agents de l'environnement. Par conséquent, les décisions basées sur les observations partielles effectuées par chacun des agents pourraient être sous-optimales et la recherche d'une solution globale par ce moyen devient alors impossible à résoudre.

3. Abstraction

Dans un système d'agent, il est supposé qu'un agent connaît l'intégralité de son espace d'actions et que le mappage de l'espace d'états à l'espace d'actions peut être effectué par expérience. Dans SMA, chaque agent ne rencontre pas tous les états. Pour créer une carte, il doit être en mesure de tirer parti de l'expérience d'autres agents dotés de capacités ou de pouvoirs décisionnels similaires. Dans le cas d'agents coopérants ayant des objectifs similaires, cela peut être facilement réalisé en créant une communication entre les agents.

4. Résolution de conflit

Les conflits résultent du manque de vue globale disponible pour chacun des agents. Une action sélectionnée par un agent pour modifier un état interne spécifique peut être mauvaise pour un autre agent. Dans ces circonstances, les informations sur les contraintes, les préférences d'action et les objectifs prioritaires des agents doivent être partagées pour améliorer la coopération. Un problème majeur est de savoir quand communiquer cette information et à quel agent [6] .

1.2.2.8 Domaines d'application SMA

Parmi les domaines d'application des SMA :

- **Télécommunication, système de transport, réseaux** : Systèmes hétérogènes et ouverts
- **Travail collaboratif assisté par ordinateur** : Agents assistants, agents médiateurs
- **Robotique** : Robots automates mobiles
- **Télématique (internet)** : Agent “ intelligents“, agents d'interface.
- **Simulation de systèmes complexes** : Simulation individu-centrée
- **Systèmes multi-capteurs**
- **Les jeux vidéo** (intelligence des caractères) [9].

1.3 Réseaux informatique à base des systèmes multi agents

La collaboration d'un réseau connecté avec un système multi agent s'harmonise bien et nous pensons qu'il s'agit d'un progrès important dans la prise en charge de la conception du système SMA pour les réseaux informatique.

Les composants communiquant dans un réseau sont représentés par plusieurs nœuds d'agents qui s'occupent de la tâche d'acheminement des données entre les nœuds. Ce modèle a été choisi en partie parce que la réussite de la tâche nécessite une mobilité des agents qui influence de manière dynamique à la communication et la fiabilité du réseau. Cela en introduisant le concept de travail d'équipe (coopération entre agents pour l'acheminement des paquets de données entre les nœuds) et le concept de compétition entre agents pour résoudre la contrainte d'accès au medium de communication.

La communication entre les nœuds d'agents consiste en une signalisation pour l'affectation et la coordination d'équipe, le partage de l'intention d'un agent individuel et la diffusion d'autres connaissances relatives à l'environnement partagé. La diffusion de l'intention d'un agent permettait aux agents de résoudre les conflits d'affectations d'équipe inefficaces ou contradictoires. Prenons l'exemple des réseaux mobile ad hoc MANET.

1.3.1 Réseaux mobile ad-hoc (MANET)

Le réseau MANET est un ensemble de nœuds mobiles sans fil formant dynamiquement un réseau temporaire sans l'utilisation d'une administration centralisée. Le défi dans les MANET est de trouver un chemin entre les nœuds communicants. Un tel type de réseau se caractérise par l'absence d'infrastructure centralisée, la topologie dynamique, la contrainte d'énergie, l'hétérogénéité des nœuds, la communication multi-sauts et la bande passante limitée [11]. Les nœuds sont libres de se déplacer, ainsi la topologie du réseau peut changer dynamiquement. Par conséquent, les protocoles de routage doivent pouvoir trouver des chemins (séquences de nœuds intermédiaires à une destination), et les protocoles d'accès au médium doivent pouvoir assurer une bonne gestion d'occupation de canal radio partagé par les nœuds voisins.

1.3.2 SMA pour les réseaux MANET

Il y a eu des progrès significatifs dans le développement des solutions MANET ces dernières années, mais il reste encore beaucoup à faire. Dans l'ensemble, il reste une expérience limitée dans l'utilisation / l'adaptation de protocoles et d'applications de couche bases dans ces environnements. Des travaux supplémentaires sont également nécessaires pour développer et adapter le routage dans ces environnements afin de mieux prendre en charge les communications orientées groupe. Les systèmes multi-agents intelligents sont intéressants dans ce type d'environnement. Ces systèmes sont composés d'entités logicielles autonomes en en compétition ou en coopération. Dans ce dernier cas, ces entités peuvent être axées sur les objectifs et capables de se coordonner (communiquer) au sein d'une équipe pour atteindre les objectifs [10].

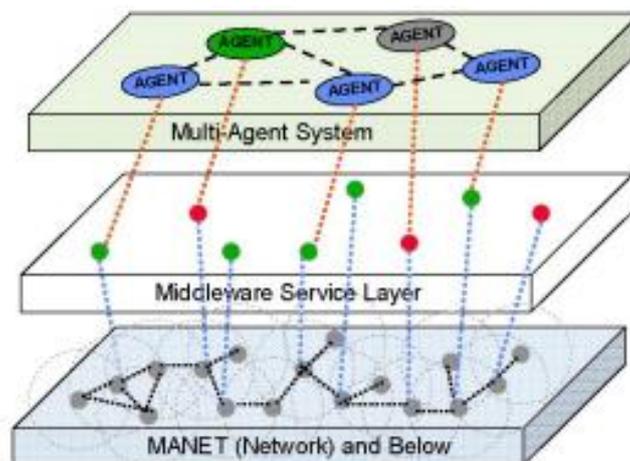


Figure 1.10 réseau Ad hoc et SMA [10]

1.3.3 SMA et le routage dans les réseaux MANET

Le système de gestion de routage dans les réseaux Ad hoc nécessite une distribution des opérations, car il n'existe aucun élément central fixe qui peut organiser le réseau, il nécessite la coopération entre les différents nœuds mobiles car un nœud mobile ne peut communiquer qu'avec les nœuds proches de lui, il a donc besoin de coopérer avec les autres nœuds mobiles qui vont l'aider à transmettre les informations.

Le système de gestion de routage dans les réseaux Ad hoc doit aussi être robuste, il doit pouvoir continuer à fonctionner normalement en cas de défaillance d'un ou de plusieurs nœuds, il doit être intelligent, il doit choisir les meilleures routes pour transmettre les paquets et réagir rapidement aux événements qui peuvent intervenir. Les systèmes multi agent prennent en compte les aspects de coopération, d'autonomie, de distribution et l'intelligence. Ils semblent donc appropriés pour modéliser le comportement des réseaux Ad hoc [5].

1.4 Conclusion

Dans ce chapitre nous avons présenté la technologie des systèmes multi-agents (SMA) et sa capacité à résoudre des problèmes dynamique et distribuée dans laquelle les agents logiciels distribués sont capables à la fois de détecter et d'agir dans des environnements complexes. Les architectures basées sur MAS peuvent être très utiles et fournir des solutions réparties basées sur le travail d'équipe à des problèmes complexes.

Nos premières analyses ont montré que les communications de travail d'équipe SMA peuvent être utilisées dans le contexte MANET grâce à une meilleure conception entre les couches. De ce fait, le principal objectif de notre travail est d'étudier la résolution de problèmes SMA en équipe en utilisant l'attribution dynamique des rôles dans le contexte des environnements Réseau mobile Ad hoc (MANET). Pour faire nous utilisons l'un des simulateurs multi Agents les plus utilisé dans ce domaine à savoir NETLOGO pour modéliser et simuler un modèle MANET. Une présentation de l'outil est faite dans le chapitre 3.

Chapitre 2 *Généralités sur les réseaux ad hoc*

2.1 Introduction

Dans ce chapitre, nous décrirons les réseaux ad hoc, Certains avantages et possibilités d'application des réseaux ad hoc sont également mentionnés. Comme tout autre système de communication sans fil, les réseaux ad hoc sont limités dans leurs capacités par les limitations de la technologie radio sur les vitesses et la portée de transmission des données. Afin d'avoir une idée juste de ces restrictions, nous résumerons dans ce chapitre les caractéristiques de base de certaines technologies radio couramment utilisées au niveau physique dans les réseaux ad hoc.

2.2 Réseaux sans fils

Un réseau sans fils est système de communication qui repose sur la transmission du signal sur un support sans la présence de fils ou de câbles entre l'expéditeur et le récepteur. Les supports de communication possibles pour la communication sans fil comprennent l'air, l'eau ou le vide. Les communications sans fil peuvent prendre en charge un degré élevé de mobilité et de flexibilité de déploiement, ils sont donc le principal moyen de communication de choix pour les réseaux ad hoc et de capteurs.

L'attractivité des communications sans fil, l'absence de fils, présente également des inconvénients, à savoir la vulnérabilité aux interférences et au bruit de fond lors de la traversée du support sans fil. En conséquence, la qualité de signal attendue d'une liaison de communication sans fil est relativement inférieure, moins stable et moins prévisible qu'une liaison filaire. Les communications sans fil sont également intrinsèquement moins sécurisées que les communications filaires [12].

2.2.1 Classification des réseaux sans fils

Les réseaux sans fil peuvent être classés selon la zone de couverture (le périmètre géographique offrant une connectivité) ou selon l'existence d'une infrastructure de communication [13].

a) Suivant la portée (zone de couverture)

En fonction de cette caractéristique les réseaux sans fils peuvent être classés en 4 catégories :

- **Les réseaux personnels sans fil** (WPAN pour Wireless Personal Area Network) : technologie Bluetooth, infrarouge.
- **Les réseaux locaux sans fil** (WLAN pour Wireless Local Area Network) : technologie Wifi.
- **Les réseaux métropolitains sans fil** (WMAN pour Wireless Metropolitan Area Networks) : technologie BLR (Boucle Local Radio).
- **Les réseaux étendus sans fil** (WMAN pour Wireless Wide Area Networks) technologie GPRS [14].

b) Suivant l'infrastructure

Selon l'infrastructure d'un réseau, on peut distinguer deux types de réseaux sans fils :

- **Réseau mobile avec infrastructure** : intègre deux ensembles d'entités distinctes :
 - Les sites fixes d'un réseau de communication filaire classique
 - Les sites mobiles.

Cette infrastructure contient une station de base (SB) et des unités mobiles qui se trouve au près de cette SB formant des cellules (voir **Figure 2.2**). A chaque station de base correspond une cellule à partir de laquelle des unités mobiles peuvent émettre et recevoir des messages. 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.

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 [15].

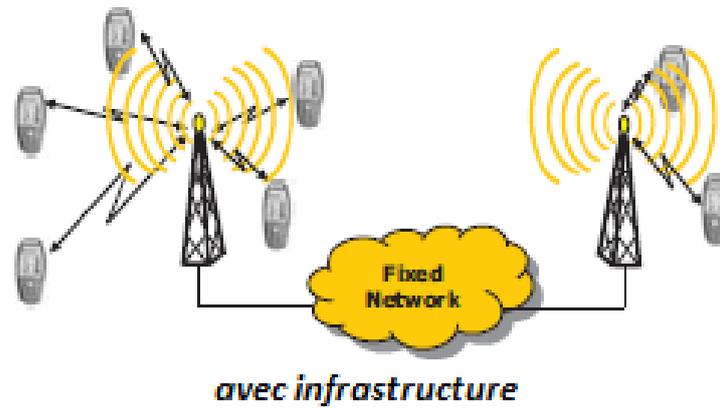


Figure 2.1 Réseau sans fils avec infrastructure [16]

➤ Réseau mobile sans infrastructure (Ad hoc)

Ce type de réseaux ne comporte pas de site fixe. Tous les sites du réseau sont mobiles et communiquent d'une manière directe en utilisant leurs interfaces de communication sans fil (voir Figure 2.3). L'absence d'infrastructure ou de réseau filaire composé de 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 [15].

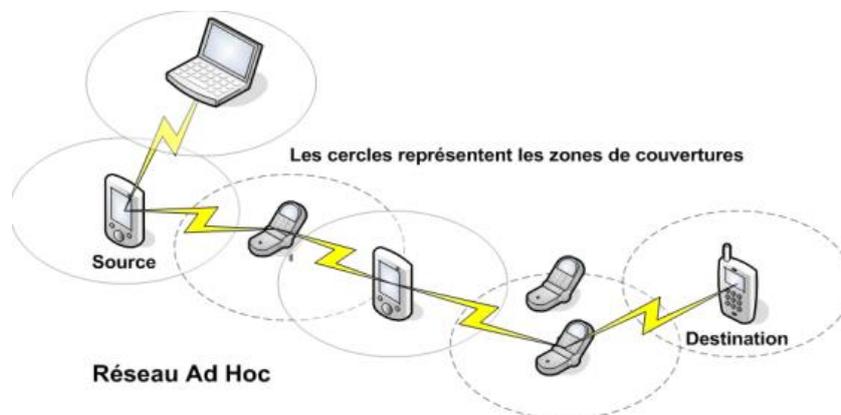


Figure 2.2 Réseau sans fils sans infrastructure Ad hoc [17]

Réseau cellulaire	Réseau sans fils Ad hoc
Réseau d'infrastructure	Réseau sans infrastructure
Sites cellulaires et station de base fixes et pré-localisés	Aucune station de base et déploiement rapide
Topologie de réseau dorsale statique	Topologies de réseau hautement dynamiques avec multi-sauts
Environnement relativement attentionné et connectivité stable	Environnement hostile (bruit, pertes) et connectivité irrégulière
Planification détaillée avant l'installation de la station de base	Le réseau ad hoc forme et s'adapte automatiquement aux changements
Coûts d'installation élevés	Coûts rentable
Plus de temps de configuration	moins de temps d'installation

Tableau 2.1. Tableau comparatif entre réseau cellulaire et Ad hoc [18]

2.3 Réseaux ad hoc

2.3.1 Définition d'un réseau Ad hoc

Un réseau ad-hoc est un type de réseau sans fil poste-à-poste (peer-to-peer ou P2P) qui permet à un ensemble de nœuds (ou routeurs) mobiles sans fil de former dynamiquement un réseau temporaire sans recourir à une infrastructure réseau existante ou à une administration centralisée. Les routeurs sont libres de se déplacer au hasard et de s'organiser arbitrairement, ainsi, la topologie sans fil du réseau peut changer rapidement et de façon imprévisible [18].

Chaque nœud est équipé d'un émetteur et récepteur radio qui lui permet de communiquer avec les autres nœuds. Les réseaux ad hoc étant des réseaux auto-organisés, la communication dans les réseaux ad hoc ne nécessite pas de station de base centrale. Chaque nœud d'un réseau ad hoc peut générer des données pour tout autre nœud du réseau. Tous les nœuds peuvent fonctionner, si nécessaire, en tant que stations relais pour acheminer les paquets de données vers leur destination finale. Un réseau ad hoc peut être connecté via des passerelles dédiées, ou des nœuds

fonctionnant comme des passerelles, à d'autres réseaux fixes ou à Internet. Dans ce cas, le réseau mobile ad-hoc étend l'accès aux services de réseau fixe [16].

2.3.2 Rappels sur la topologie d'un réseau

La topologie de réseau est déterminée par la configuration des raccordements entre les nœuds, c'est donc une partie de la théorie des graphes. Il existe plusieurs topologies :

- **Une topologie complète** : C'est une topologie de réseau dans laquelle il y a un lien direct entre toutes les paires de nœuds. Dans un réseau entièrement relié avec n nœuds, il y'a $n(n-1)/2$ liens directs, voir Figure.2.3. (a).
- **Topologie linéaire** : Un réseau avec une topologie linéaire est tel qu'il y a une ligne simple à laquelle tous les nœuds sont reliés, et les nœuds se relient seulement à ce lien, voir Figure.2.3. (b).
- **Topologie de maille** : C'est une topologie de réseau dans laquelle il y a au moins deux nœuds avec deux chemins ou plus entre eux, voir Figure.2.3. (c).
- **Topologie d'anneau** : C'est une topologie de réseau dans laquelle chaque nœud a exactement deux branches reliées. Ces nœuds et branches forment un anneau. Si un des nœuds sur l'anneau échoue, l'anneau est cassé et il ne devient plus fonctionnel, voir Figure.2.3. (d).
- **Topologie d'étoile** : C'est une topologie de réseau dans laquelle les nœuds périphériques sont reliés à un nœud central. Ces nœuds peuvent communiquer avec tous les autres qu'à travers le nœud central. L'échec d'une ligne de transmission liant n'importe quel nœud périphérique au nœud central aura comme conséquence l'isolement de ce nœud périphérique de tous les autres, voir Figure.2.3. (e).
- **Topologie d'arbre** : C'est une topologie de réseau dans laquelle les nœuds sont arrangés dans une structure d'arbre. D'un point de vue purement topologique, ceci ressemble à une intercommunication des réseaux d'étoile. De différents nœuds périphériques (feuilles) sont uniquement exigés pour la transmission ou la réception relative à un autre nœud, voir Figure.2.3 (f) [19].

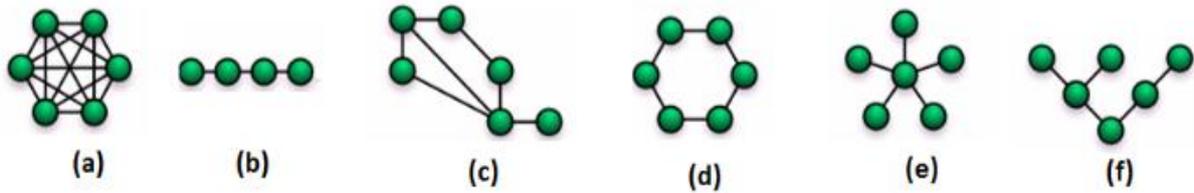


Figure.2.3 Topologies de réseau

2.3.3 Pile protocolaire dans les réseaux Ad hoc

Comme la plupart des réseaux filaires ou mobiles, les réseaux mobiles Ad hoc sont organisés selon une structure en couches. Cette dernière se décompose pour les réseaux mobiles Ad hoc en cinq couches (couche physique, couche liaison de données, couche réseau, couche transport, couche application).

Le modèle de structure adopté par les réseaux mobiles Ad hoc est en fait une adaptation du modèle TCP/IP existant pour les réseaux filaires, ce dernier étant lui-même inspiré du modèle OSI (Open Systems Interconnexion) et comme Le modèle OSI qui a réparti les protocoles utilisés selon sept couches, définissant ainsi un langage commun pour le monde des télécommunications et de l'informatique [19].

OSI	TCP/IP	Ad hoc
Couche application	Couche application (HTTP, FTP)	Couche application
Couche présentation		
Couche session	Couche transport	Couche transport
Couche transport		
Couche réseau	Couche Internet (IP)	Couche réseau

Couche liaison	Couche interface réseau	Couche de liaison de données (MAC et LLC)
Couche physique		Couche physique

Tableau 2.2 La structure en couches des réseaux Ad hoc par rapport à OSI et TCP/IP

2.3.4 Caractéristiques et contraintes des réseaux ad hoc

En général, un réseau Ad Hoc mobile est considéré comme un système autonome dynamique composé de nœuds mobiles interconnectés par des liens sans fil, sans l'utilisation d'une infrastructure fixe et sans administration centralisée. Les nœuds sont libres de se déplacer aléatoirement et s'organisent arbitrairement. Par conséquent, la topologie du réseau peut varier de façon rapide et surtout imprévisible. Plusieurs défis doivent être relevés ; les principaux sont cités dans [20].

- **Mobilité** : le fait que les nœuds puissent être rapidement repositionnés et / ou déplacés est la raison d'être des réseaux ad hoc. Un déploiement rapide dans des zones sans infrastructure implique souvent que les utilisateurs doivent explorer une zone et peut-être former des équipes. Le modèle de mobilité peut avoir un impact majeur sur la sélection d'un schéma de routage et peut ainsi influencer les performances.
- **Multihopping** : un réseau multihop est un réseau où le chemin de la source à la destination traverse plusieurs autres nœuds. Les réseaux ad hoc présentent souvent plusieurs sauts pour la négociation d'obstacles, la réutilisation du spectre et la conservation de l'énergie.
- **Auto-organisation** : le réseau ad hoc doit déterminer de manière autonome ses propres paramètres de configuration, notamment : adressage, routage, mise en cluster, identification de position, contrôle de l'alimentation, etc.
- **Évolutivité** : dans certaines applications (par exemple, les grandes structures de capteurs environnementaux, les déploiements sur le champ de bataille, les réseaux de véhicules urbains, etc.), le réseau ad hoc peut atteindre plusieurs milliers de nœuds.

- **Économie d'énergie** : la plupart des nœuds ad hoc (par exemple, ordinateurs portables, PDA, capteurs, etc.) ont une alimentation électrique limitée et aucune capacité à générer leur propre énergie (par exemple, des panneaux solaires). La conception de protocoles éconergétiques (par exemple, MAC, routage, découverte de ressources, etc.) est essentielle pour la longévité de la mission.
- **Sécurité** : les défis de la sécurité sans fil sont bien connus - capacité des intrus à écouter et à brouiller le canal. Une grande partie du travail effectué dans les réseaux d'infrastructure sans fil généraux s'étend au domaine ad hoc. Les réseaux ad hoc sont cependant encore plus vulnérables aux attaques que leurs homologues d'infrastructure.

2.3.5 Avantages des réseaux Ad hoc

Les avantages de cette technologie sont nombreux du fait qu'il n'y a pas besoin d'infrastructure préexistante :

- Les réseaux ad hoc peuvent être déployés dans un environnement quelconque.
- Le coût d'exploitation du réseau est faible : aucune infrastructure n'est à mettre en place initialement et surtout aucun entretien n'est à prévoir.
- Le déploiement d'un réseau ad hoc est simple : ne nécessite aucun pré requis puisqu' il suffit de disposer d'un certain nombre de terminaux dans un espace pour créer un réseau ad hoc, et rapide puisqu' il est immédiatement fonctionnel dès lors que les terminaux sont présents.
- La souplesse d'utilisation : est un paramètre très important puisque les seuls éléments pouvant tombés 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) [17].

2.3.6 Inconvénients des réseaux Ad hoc

Même si les perspectives pour les réseaux ad hoc sont prometteuses, plusieurs contraintes restent encore à traiter :

- La connectivité limite les possibilités de communication. Ainsi, deux stations ne sont joignables que s'il existe un ensemble de stations pouvant assumer la fonction de routeur afin de faire suivre les paquets de données échangées entre les deux stations.
- Les liens entre les stations ne sont pas isolés les uns des autres et polluent le voisinage, par diffusion, lors de chaque émission/réception de données. Par conséquent, tout paquet de diffusion émis vers une station en cours de communication (que le paquet lui soit destiné ou pas) va altérer la communication de cette station. La diffusion est un facteur qui alourdit aussi d'autres paramètres tels que la bande passante et la consommation de batterie.
- La sécurité dans les réseaux ad hoc est difficile à contrôler, notamment parce que dans l'interface air l'écoute clandestine est très simple à réaliser.
- Enfin, la faible autonomie des batteries constitue un frein à une utilisation longue du terminal et à la mise en place de nouveaux services. C'est une contrainte qui existe certes dans les réseaux de type gsm ou umts, mais qui est plus forte dans les réseaux ad hoc, puisque les ressources énergétiques sont mises en commun même pour les besoins du routage. Nous nous intéressons dans cette thèse, plus spécialement, à ce dernier point. Nous proposons, dans le dernier chapitre, des solutions permettant de mieux gérer cette consommation des batteries [17].

2.3.7 Domaines d'utilisation des réseaux Ad hoc

Les réseaux sans fil ad hoc sont utiles dans les situations qui nécessitent un déploiement de réseau local rapide ou sans infrastructure, comme la réponse aux crises, les réunions de conférence, les applications militaires et par le personnel de secours en cas de catastrophe. Ci-dessous, nous présentons Certains scénarios dans lesquels un réseau ad hoc peut être utilisé :

- **Applications de collaborations** : Les utilisateurs professionnels ont besoin d'applications particulières lors d'échanges entre collaborateurs. Ainsi, au cours de réunions ou de conférences, ces utilisateurs peuvent ressentir le besoin de former dans n'importe quel lieu un réseau pour s'échanger des informations, ou faire une vidéo conférence entre bureaux voisins. Les réseaux ad hoc sont bien appropriés à ces besoins.
- **Jeux Vidéo** : Les réseaux sans fil sont bien adaptés pour permettre l'échange d'informations entre applications personnelles. Ainsi, pour les utilisateurs voulant jouer en réseau, il est facile et à faible coût de déployer un réseau ad hoc, le meilleur exemple est la PSP de Sony.
- **Urgences** : Lors de catastrophes d'origine naturelles (comme les tremblements de terre, les tsunamis, les feux de forêt ou d'habitations...) ou non, les infrastructures préexistantes peuvent ne pas être opérationnelles compliquant d'autant plus les besoins de communications des moyens de secours. Les réseaux sans fil, par leur compacité et leur rapidité de déploiement, permettent aux différentes équipes de secours d'établir rapidement des liaisons et d'échanger des informations.
- **Militaires** : Lors d'interventions en milieu hostile, il peut être difficile ou trop encombrant d'utiliser un réseau à infrastructure. Les réseaux sans fil sont parfaitement bien adaptés à ce type d'environnement où les déplacements restent peu rapides et peu soutenus.
- **Étendre les réseaux** : Un des major problème des réseaux avec infrastructure est la couverture limitée, pour cela les réseaux ad hoc sont sollicités afin d'étendre la couverture des réseaux cellulaire par exemple [17].

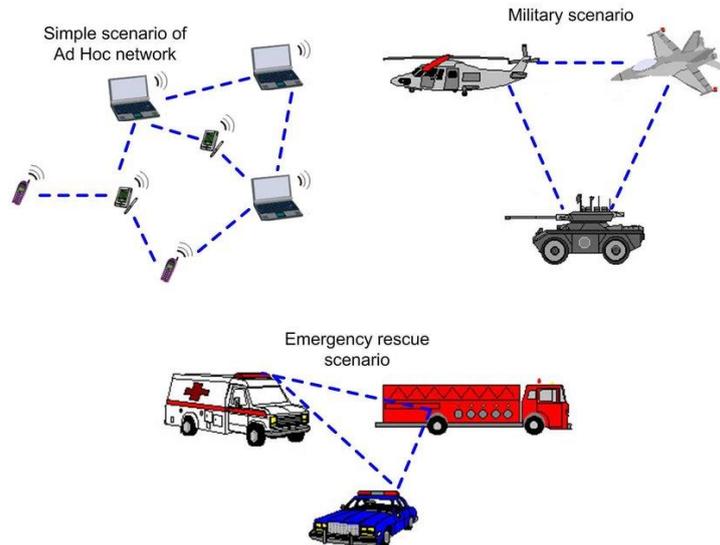


Figure 2.4 Différentes utilisations des réseaux Ad hoc [21]

2.3.8 Technologies de mise en réseau ad hoc

En mode ad hoc, Les stations qui se trouvent dans la même zone de couverture, communiquent entre elles d'une façon directe. Cette communication est de type pair-à-pair sans aucun protocole de routage. Ce mode de fonctionnement représente le type le plus basique des réseaux Ad Hoc. La plupart des chercheurs considéreront les réseaux sans fil ad hoc comme un sous-ensemble spécial de réseaux sans fil. En fait, la technologie radio ad hoc et la plupart des technologies MAC seront tirées par les progrès des réseaux d'infrastructure sans fil.

Selon la taille de la zone de service, une technologie radio développée pour les réseaux personnels sans fil (WPAN), les réseaux locaux sans fil (WLAN) ou les réseaux sans fil métropolitains (WMAN) peut être adoptée pour les réseaux ad hoc [16]. Pour chaque type de réseau, différentes technologies sans fil ont été proposées. Quelques exemples sont :

- WPAN (couverture jusqu'à 20m) : Bluetooth (802.15), UWB
- WLAN (couverture jusqu'à 100 m) : IEEE 802.11 (IEEE 802.11 a/b/g/n/i/h).
- WMAN (couverture de l'ordre de quelques kilomètres) : IEEE 802.16 (WiMAX).

Le débit binaire maximal pris en charge, l'allocation de fréquences et les plages typiques sont des caractéristiques importantes qui déterminent la pertinence de chaque technologie pour les applications à fournir par le réseau ad hoc. Par exemple, des réseaux de capteurs denses à bas débit peuvent être construits sur la base d'une technologie WPAN, tandis que pour la communication entre des voitures en mouvement à des distances de l'ordre de dizaines de mètres, une technologie WLAN comme IEEE 802.11b peut être plus appropriée. Dans ce qui

suit, nous présentons les principales technologies utilisées pour satisfaire le besoin des réseaux ad hoc.

a) IEEE802.11

Le premier standard de WLAN est l'IEEE 802.11, proposé par l'IEEE en 1997, connue aussi sous la dénomination Wi-Fi (Wireless Fidelity), qui est une technologie de réseau sans fil [22].

La première norme Wi-Fi est appelée IEEE802.11b. Il a ensuite été décliné en IEEE802.11 a/g/n/i/e/h, qui se concentre respectivement sur la qualité de service (QoS) et la sécurité. En règle générale, les appareils compatibles Wi-Fi ont des distances de couverture allant de 50 à plus de 100 mètres. En pratique, cette distance de couverture dépend fortement de la nature de l'antenne et sur l'environnement dans lequel évoluent les appareils [23].

b) Bluetooth

Le Bluetooth est l'une des principales technologies sans fil développées pour réaliser des WPAN (Wireless PAN), et qui utilise les ondes radios dans la bande de fréquence de 2,4 GHz [27]. Il était initialement limité à une distance de fonctionnement de seulement 10 mètres et à une vitesse d'environ 1 Mbit / s, et utilise la technique du spectre étalé à sauts de fréquence (FHSS). Lorsque les appareils Bluetooth se trouvent à portée l'un de l'autre, ils établissent le contact et forment un réseau temporaire appelé PAN (Personal Area Network). Comme le WiFi, le Bluetooth connaît un succès considérable : il existe des souris Bluetooth, des écrans Bluetooth, des PDA Bluetooth, etc.

c) Hiperlan

La norme HiperLAN2 est très proche de 802.11a / g en termes de couches physiques qu'elle utilise, toutes les deux utilisent la technologie OFDM, mais sont très différentent au niveau MAC et dans la façon dont les paquets de données sont formés et les périphériques sont adressés. [23].

Hiperlan fonctionne en partageant les canaux de 20 MHz dans le spectre de 5 GHz dans le temps, en utilisant l'accès multiple par répartition dans le temps (TDMA) pour fournir une QoS via des mécanismes de type ATM. Il prend en charge deux modes de fonctionnement de base : le mode centralisé et le mode direct :

- Le mode centralisé est utilisé dans la topologie de réseau cellulaire, où chaque cellule radio est contrôlée par un point d'accès couvrant une certaine zone géographique.
- Le mode direct est utilisé dans la topologie de mise en réseau ad hoc, principalement dans le contexte du réseau personnel où une cellule radio couvre toute la zone de desserte.

d) ZigBee

Les appareils compatibles ZigBee sont conformes à la norme IEEE 802.15.4-2003. Cette norme spécifie ses couches de protocole inférieures, la couche physique (PHY) et le contrôle d'accès moyen (MAC). Il cible le réseau personnel sans fil à faible débit (WPAN). La recherche sur les réseaux de type ZigBee a commencé en 1998.

Zigbee était censé fonctionner dans des contextes dans lesquels le Wi-Fi et le Bluetooth ne conviennent pas. Zigbee fonctionne dans les bandes ISM 2,4 GHz, 915 MHz et 868 MHz non autorisées. Il utilise un codage à spectre étalé à séquence directe (DSSS). La puissance de sortie maximale des antennes ZigBee étant généralement de 1 mW, la portée de transmission des nœuds ZigBee est comprise entre 10 et 75 mètres. La couche MAC spécifiée par IEEE 802.15.4-2003 est CSMA / CA [23].

e) Réseau sans fil à large bande

Parmi les technologies de communication sans fil utilisées pour la mise en place des réseaux ad hoc portant sur une zone géographique étendue :

- **WiMAX** (IEEE 802.16) signifie Interopérabilité mondiale pour l'accès aux micro-ondes. IEEE 802.16 offre des débits de données allant jusqu'à 70 Mbit / s sur une distance de 50 km. Cependant, les limites pratiques des tests du monde réel semblent se situer entre 500 kbit / s et 2 Mbit / s à une distance d'environ 5-8kms.
- **HIPERMAN**, qui signifie High Performance Radio Metropolitan Area Network, est une alternative européenne au WiMAX. La norme a été créée par l'Institut européen des normes de télécommunications (ETSI). Il fournit une communication réseau sans fil dans les bandes 2-11 GHz [23].

2.3.9 Protocoles de contrôle d'accès (MAC) pour les réseaux Ad hoc

Il existe plusieurs problèmes importants dans les réseaux sans fil ad hoc. La plupart des applications de réseau sans fil ad hoc utilisent la bande industrielle, scientifique et médicale (ISM), qui est libre de formalités de licence. Le sans-fil étant un support étroitement contrôlé, sa bande passante de canal est limitée, généralement bien inférieure à celle des réseaux filaires. En outre, le support sans fil est intrinsèquement sujet aux erreurs. Même si une radio peut avoir une bande passante de canal suffisante, des facteurs tels que l'accès multiple, l'évanouissement du signal, le bruit et les interférences peuvent entraîner une baisse significative du débit effectif dans les réseaux sans fil, ce qui nécessite un traitement spécial par les protocoles MAC [18].

2.3.9.1 Objectifs du protocole MAC dans les réseaux Ad hoc

Les principaux objectifs d'utilisation du protocole MAC dans les Ah doc sont :

- Permettre *un accès équitable* au média radio partagé. Le fonctionnement du protocole devrait être distribué.
- Devrait prendre en charge le trafic en temps réel.
- Le délai d'accès doit être minimisé.
- La bande passante disponible doit être utilisée efficacement.
- Allocation équitable de bande passante aux nœuds concurrents.
- Les frais généraux de contrôle doivent être réduits au minimum.
- Les effets des terminaux cachés / exposés doivent être minimisés.
- Doit être évolutif.
- Devrait minimiser la consommation d'énergie.
- Devrait assurer la synchronisation entre les nœuds. [24].

2.3.9.2 Classification des protocoles MAC

Cette partie décrit la classification des protocoles MAC et les divers facteurs considérés pour la classification. Divers schémas MAC développés pour les réseaux ad hoc sans fil peuvent être classés comme le montre la figure ci-dessous :

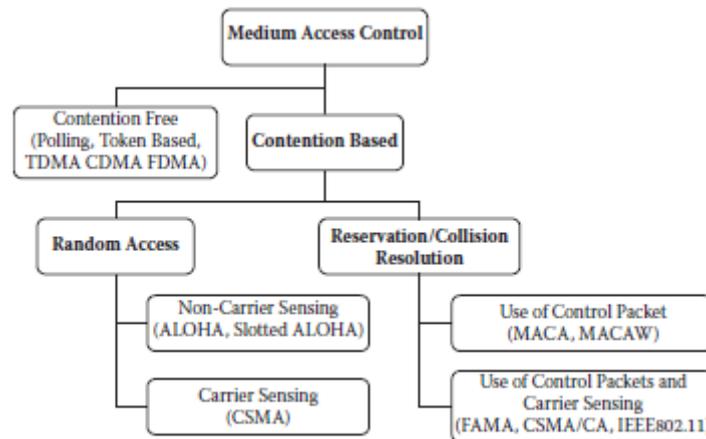


Figure 2.5 Classification des protocoles MAC [18]

Parmi les facteurs de classification :

- **Contention-based protocols without reservation/scheduling:**

- Aucune réservation de bande passante n'est effectuée.
- Les garanties ne sont pas possibles.

- **Contention-based protocols with reservation mechanisms**

- La bande passante pour la transmission est réservée à l'avance.
- Des garanties sont possibles.

- **Contention-based protocols with scheduling mechanisms :**

- La planification distribuée entre les nœuds est utilisée.
- Des garanties sont possibles [24].

La norme IEEE 802.11 spécifie deux modes de protocole MAC : le mode de fonction de coordination distribuée (DCF) (pour les réseaux ad hoc) et le mode de fonction de coordination de point (PCF) (pour les réseaux basés sur une infrastructure à coordination centralisée) [22].

DCF dans IEEE 802.11 est basé sur CSMA avec évitement de collision (CSMA / CA), qui peut être considéré comme une combinaison des schémas CSMA et MACA. Le protocole utilise la séquence RTS-CTS-DATA-ACK pour la transmission de données.

Le protocole utilise non seulement la détection de porteuse physique, mais introduit également le nouveau concept de détection de porteuse virtuelle. Ceci est mis en œuvre sous la forme d'un vecteur d'allocation de réseau (NAV), qui est maintenu par chaque nœud [18]. La norme IEEE 802.11 sera présentée de façon détaillée dans le chapitre suivant.

2.4 Conclusion

Ce chapitre a été consacré à la présentation des réseaux sans fils en particulier les réseaux Ad hoc sont utilisés dans toute application où le déploiement d'une infrastructure réseau filaire est trop contraignant, soit parce qu'il est difficile à mettre en place, soit parce que la durée d'installation du réseau ne justifie pas de câblage à demeure.

Ces réseaux qui consistent en une collection de terminaux (ou nœuds) capable de s'auto organiser, et qui communiquent les uns avec les autres sans le support d'une quelconque infrastructure de gestion prédéfinie. Ceci signifie que les nœuds jouent à la fois le rôle d'hôte et de routeur : ils sont d'une part responsables de l'émission et de la réception de leurs propres données, et ils assurent d'autre part la retransmission du trafic des autres nœuds.

L'un des défis fondamentaux des réseaux Ad hoc est de savoir comment augmenter le débit global du réseau et de fournir un accès équitable au canal de diffusion partagé, tout en maintenant une faible consommation d'énergie pour le traitement des paquets et les communications [18]. Cet objectif est basé sur la contention des protocoles MAC couramment utilisés dans les réseaux MANET, à savoir les protocoles issus de la norme 802.11. Cette dernière sera présentée dans le chapitre suivant.

Chapitre 3 *Présentation du standard IEEE 802.11*

3.1 Introduction

Comme nous l'avons vu dans le chapitre précédent, la technologie IEEE 802.11 est une bonne plate-forme pour implémenter des réseaux ad hoc. Le standard IEEE 802.11 ne concerne qu'une partie de la couche liaison de données (la sous couche MAC) et la couche physique (PHY) du modèle TCP/IP. Donc il est totalement transparent par rapport aux applications et aux protocoles des couches supérieures (TCP/IP, UDP/IP, http, FTP,...). Dans ce chapitre nous présentons le standard IEEE 802.11, notamment les deux couches PHY / MAC de la norme, en s'intéressant en particulier sur le mode DCF utilisé dans les réseaux Ad hoc.

3.2 Norme IEEE 802.11

La norme IEEE 802.11 connue aussi sous la dénomination Wi-Fi (Wireless Fidelity), a été publiée pour la première fois en 1999 pour définir les couches MAC et PHY pour les réseaux sans fils WLAN. Par la suite, en 2007, la norme a été rééditée avec quelques modifications. Les couches MAC et PHY permettent d'appliquer un protocole de données unique pour plusieurs techniques de transmission RF. La norme IEEE 802.11 définit deux types de réseaux : Ad hoc et Infrastructure [25].

La couche MAC (Media Access Control) IEEE 802.11 est divisée en deux modes : la fonction de coordination distribuée (DCF) et la fonction de coordination de point (PCF), et s'occupe de coordonner l'accès à la couche physique.

La couche physique est chargée de véhiculer le flux binaire depuis l'émetteur jusqu'au récepteur.

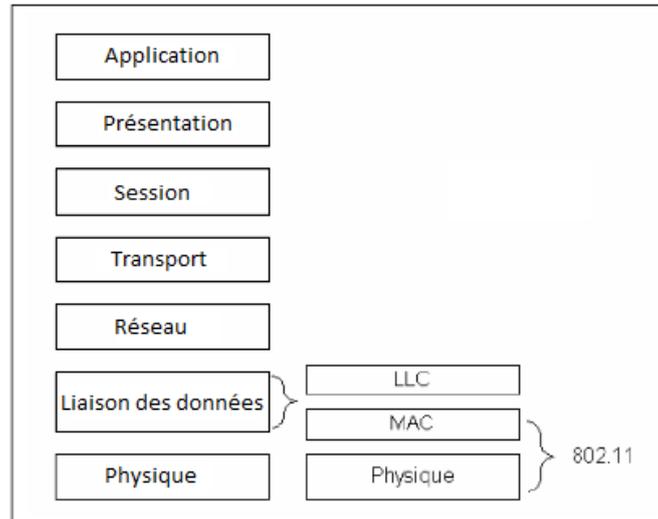


Figure 3.1. Architecture en couche de la norme IEEE 802.11 [26].

3.2.1 Evolutions du 802.11

Au fil des années, des améliorations importantes ont été apportées au standard 802.11. Certaines concernent la couche physique, d'autres concernent la couche MAC. Ces améliorations sont simplement désignées par une lettre rajoutée au nom du standard, de façon simplement séquentielle (802.11a, 802.11b...). Les principales améliorations concernant les couches physiques sont :

- 802.11a : fréquence radio à 5 GHz au lieu de 2,4 GHz, modulation radio de type OFDM (voir paragraphes suivants), débit maximal théorique de 54 Mb/s.
- 802.11b : fréquence radio à 2,4 GHz, modulation DSSS ou HR-DSSS, débit maximal théorique de 11 Mb/s.
- 802.11g : fréquence radio à 2,4 GHz, modulation DSSS, HR-DSSS ou OFDM, débit maximum théorique 54 Mb/s.
- 802.11n : devrait être ratifié en janvier 2010, mais des études (draft) du standard ont été publiées depuis 2006 et sont déjà utilisées aujourd'hui. Compatible avec le 802.11a et le 802.11b/g, il permet, grâce à de nombreuses améliorations techniques telles que le MIMO, d'atteindre des débits très élevés (> 100 Mb/s réels) [27].

Nous allons maintenant aborder plus en détail la couche physique de l'IEEE 802.11, en commençant par quelques concepts de base sur la radio, puis en détaillant les modulations radio les plus importantes.

3.2.2 Couche physique PHY d'IEEE 802.11

IEEE 802.11 repose principalement sur les ondes radio pour la transmission d'information. Les suites de bits à transmettre (par exemple, des paquets suivant le protocole TCP/IP) ne sont pas transmis tels quel dans les airs. En effet, le signal binaire transmis directement (transmission dite en bande de base) ne serait pas adapté au canal de communication hertzien. Afin de faciliter la propagation, on module le signal autour d'une fréquence porteuse.

Le réseau IEEE 802.11 étant conçu pour une utilisation sans licence, les concepteurs s'orientent rapidement vers des bandes de fréquences libres, dont les bandes ISM (Industrial, Scientific and Medical) à 2.4GHz puis les bandes N-II à 5GHz. Ces fréquences appartiennent au domaine fréquentiel des micro-ondes dont la longueur d'onde est réduite, ce qui facilite l'utilisation d'antennes de tailles réduites et adaptées à l'intégration dans des équipements mobiles. Enfin, à cette fréquence, et pour des puissances faibles (entre 10 et 100mW), la portée du signal radio atteint les 500 mètres, ce qui couvre le spectre des usages prévus pour IEEE 802.11 [28].

Deux types de sous couches ont été définies dans [29] :

- **PLCP** (Physical Layer Convergence Protocol): Qui écoute le support et indique ainsi à la couche MAC si le canal de transmission est libre ou non.
- **PMD** (Physical Medium Depender) : Qui garantit l'encodage des données. La norme de base spécifie quatre types de modulation présentés dans ce qui suit.

3.2.2.1 Les grandeurs physiques des ondes

Les ondes radio, également appelées ondes hertziennes car elles furent découvertes par le physicien allemand Heinrich Hertz en 1888, sont des ondes électromagnétiques, c'est-à-dire des oscillations combinées d'un champ électrique et d'un champ magnétique. Les ondes radio, les infrarouges, la lumière visible, les ultraviolets, les rayons X ou encore les rayons gammas sont tous des exemples d'ondes électromagnétiques [27]. Voir la figure ci-dessous :

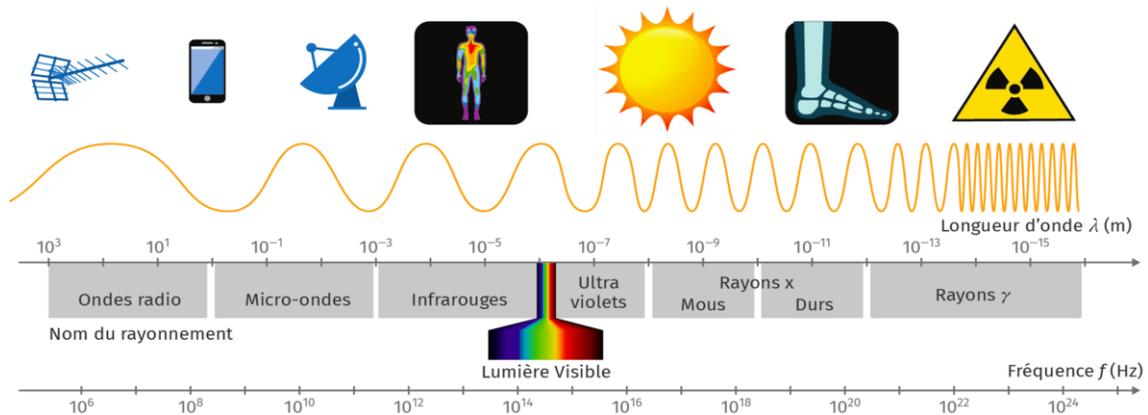


Figure 3.2. Spectre électromagnétique : le WiFi repose sur les micro-ondes.

Ces ondes transportent de l'énergie et qui peuvent se propager dans le vide .Comme toutes les oscillations (un mouvement périodique), Une onde électromagnétique a une certaine *vitesse*, *fréquence* et *longueur*. Celles-ci sont unies par une simple relation :

$$\text{Vitesse} = \text{Fréquence} * \text{Longueur d'onde}$$

La longueur d'onde (parfois nommé *lambda*, λ) est la distance séparant deux crêtes successives d'une onde périodique. La fréquence est le nombre d'ondes entières qui passent par un point fixe en une seconde. La vitesse est mesurée en mètres/secondes, la fréquence est mesurée en cycles par seconde (ou Hertz, abrégé *Hz*), et la longueur d'onde est mesurée en mètres.

Par exemple, si une onde voyage sur l'eau à un mètre par seconde, et oscille cinq fois par seconde, alors chaque onde aura une longueur de vingt centimètres :

$$1 \text{ mètre/seconde} = 5 \text{ cycles/seconde} * \lambda$$

$$0 = 1/5 \text{ mètres}$$

$$0 = 0,2 \text{ mètres} = 20 \text{ cm}$$

Les ondes ont également une caractéristique nommée *amplitude*. Celle-ci est la distance entre le centre d'une onde et l'extrémité d'une de ses crêtes, pouvant être illustrée comme étant la « hauteur » d'une vague d'eau [30].

La relation entre fréquence, longueur d'onde et amplitude est illustrée par la **figure 3.4**

Dans la pratique, avec le WiFi, on préfère utiliser directement la grandeur de puissance plutôt que des encombrer avec l'amplitude. La puissance de l'onde dépend de l'amplitude et de la

fréquence. Elle se mesure en Watt (W). Les émetteurs WiFi émettent en général des ondes d'une puissance de l'ordre de 100 mW. On parle également en décibels de milliWatt, notés **dBm** (et plus rarement en décibels de Watt, notés **dBw**). Voici les formules pour convertir d'une unité à l'autre :

$$Puissance_{dBm} = 10 \times \log (Puissance_{mW})$$

Par exemple, un émetteur Wifi à 20 dBm est un émetteur à 100 mW [27].

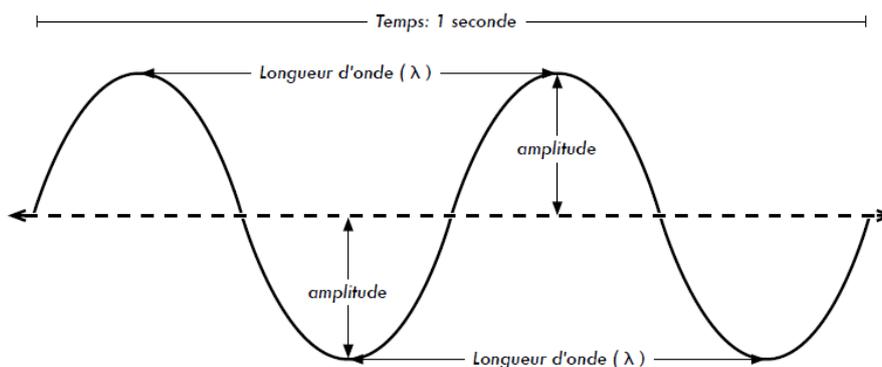


Figure 3.3. Une onde et ses grandeurs physiques [30]

3.2.2.2 Les règles de la transmission radio

Cette section identifie les paramètres pertinents qui définissent une transmission radio

1- Bruit, interférences et multipath

Le Rapport signal/bruit (RSB) est crucial pour bénéficier d'une bonne qualité de communication. Il s'exprime en décibels (dB) et correspond simplement à la différence entre la puissance du signal reçu et la puissance du bruit (exprimés en dBm) :

$$RSB = Puissance\ du\ signal\ recu_{dBm} - Puissance\ du\ bruit_{dBm}$$

Plus le RSB est importante, plus la réception est bonne et permet des débits importants. Par exemple, si le bruit est de -100 dBm et que le signal reçu est de -65 dBm, alors le RSB est de +35 dB. Parmi les sources de bruit, on trouve bien sûr les réseaux sans fil et tous les équipements radio situés à proximité, mais il y a également un bruit ambiant lié à l'activité humaine (industrielle, militaire, radios, télévision, antennes de téléphonie mobile...) et le bruit

électromagnétique naturel. La puissance du bruit naturel est en général de l'ordre de -100 dBm pour les fréquences du WiFi.

En outre, les interférences peuvent également provenir du signal lui-même : pour parvenir au récepteur, le signal peut parfois parcourir plusieurs chemins (on parle de multipath) du fait de multiples réflexions : par exemple, une partie du signal peut aller en ligne droite vers le récepteur et une autre peut rebondir sur un mur avant d'atteindre sa destination. Ce phénomène provoquera, au niveau du récepteur, à la fois des interférences radio (donc du bruit) mais aussi un signal étalé dans le temps : des symboles peuvent alors se superposer aux symboles suivants, ce qui provoque alors des erreurs de transmission. Dans ce cas, on parle d'interférence inter-symboles (Inter-Symbol Interference, ISI) [27].

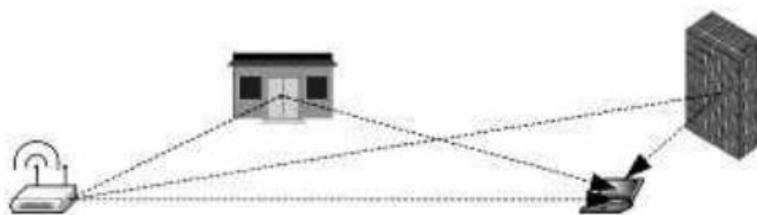


Figure 3.4. Chemins multiples (multipath) [27].

2- Le Débit

Pour obtenir un bon débit, il est nécessaire d'avoir un bon rapport signal/bruit. Puisque le RSB diminue lorsqu'on s'écarte de l'émetteur, on en déduit que le débit diminue avec la distance. De fait, avec un émetteur 802.11g à 15 dBm et un bon récepteur, on peut en théorie, en conditions idéales (pas de bruit ni d'obstacles), obtenir un débit de 11 Mb/s jusqu'à 100 mètres environ, mais au-delà le débit tombera à 5,5 Mb/s, puis à 2 Mb/s et enfin à 1 Mb/s jusqu'à plus de 300 m.

On peut comparer ce phénomène au trafic automobile sur une autoroute : le débit maximal est plus important sur une autoroute à trois voies que sur une autoroute à deux voies. Or, plus on se situe sur des fréquences élevées, plus on a « de la place » pour exploiter des bandes de fréquences larges, donc plus le débit peut être important.

Par exemple, Les radios 802.11b transmettent à **2,4 GHz** et envoient des données jusqu'à **11 Mbps** tandis que les radios 802.11a transmettent à **5 GHz** et envoient des données jusqu'à **54 Mbps**. En gros, il y a plus de canaux disponibles pour communiquer dans le 5 GHz que dans le 2,4 GHz, donc la capacité totale du 802.11a est plus importante.

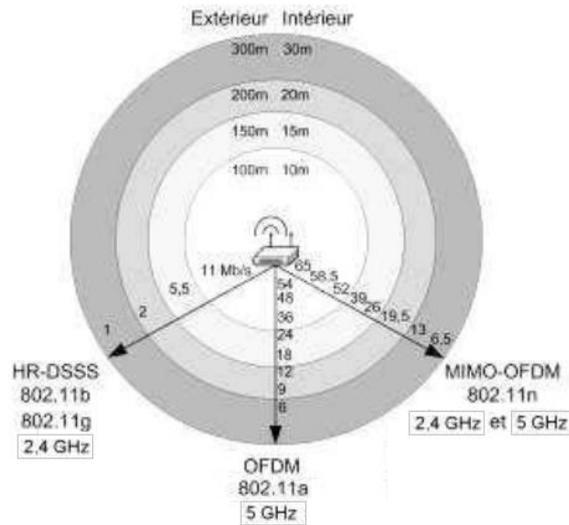


Figure 3.5. Débit théorique maximal du signal en fonction de la distance [27]

Il existe une formule, établie par Claude Shannon, mathématicien et père de la fameuse théorie de l'Information. Cette formule permet de trouver le débit maximal en fonction du rapport signal sur bruit et de la largeur de la bande de fréquence utilisée

$$C = H \times \log_2 \left(1 + \frac{Ps}{Pb} \right)$$

- **C** est la capacité maximale du canal de communication, en bits par seconde.
- **H** est la largeur de la bande de fréquence utilisée, en hertz.
- **PS** est la puissance du signal, en Watt.
- **PB** est la puissance du bruit, également en Watt.

La puissance, la fréquence, la largeur de bande, le RSB, la nature et la disposition des obstacles et la qualité des récepteurs décident donc en grande partie de la portée du signal et du débit que l'on peut atteindre [27].

3- La portée du signal

La portée du signal définit l'étendue où un paquet est reçu avec succès s'il n'y a pas d'interférences. Cette étendue dépend de la puissance de transmission et de l'atténuation du signal. Le réseau ad hoc minimal est constitué de deux stations dans la zone de couverture radio l'une de l'autre, mais il n'y a aucune limitation de la taille du réseau. Cette architecture est aussi appelée IBBS (Independent Basic Service Set).

Aussi, un terminal ne peut communiquer avec un autre terminal que si celui-ci se situe dans sa zone de couverture. Dans un réseau ad-hoc, la portée de l'IBSS est donc déterminée par la portée de chacun des terminaux le constituant [31].

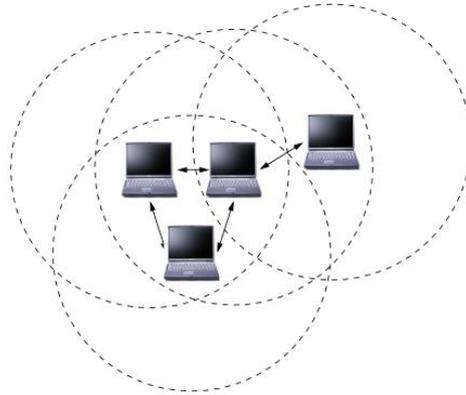


Figure 3.6. Zone de portée [31].

Un autre paramètre important est la modulation radio utilisée. C'est ce que nous allons aborder maintenant.

3.2.2.3 Les différentes techniques de modulation

La modulation est réalisée généralement dans le domaine électromagnétique en modifiant l'amplitude, la fréquence, ou la phase du signal transmis. IEEE 802.11 utilise quasi-exclusivement des modulations de **phase** et **d'amplitude**. L'ensemble des modulations utilisées par IEEE 802.11 repose sur un moduleur I (In phase)/ Q (Quadrature), illustré sur la figure 3.7, composé de deux multiplieurs, un signal de porteuse (ici, soit 2.4GHz, soit 5GHz), et un sommateur. L'intérêt de ce moduleur est de transmettre deux informations (I et Q) sur une seule porteuse et utilisant la bande passante d'un seul composant, I ou Q. Les principales modulations utilisées par IEEE 802.11 sont les suivantes :

- BPSK : 1 bit est transmis par symbole.
- QPSK : 2 bits sont transmis par symbole.
- QAM : 4 ou 6 bits sont transmis par symbole [28].

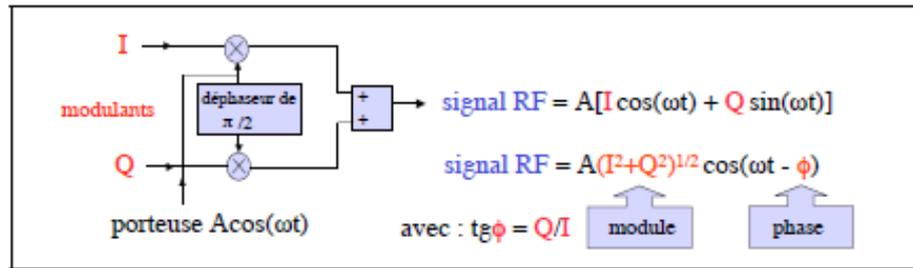


Figure 3.7. Modulateur I/Q : utilisé pour les modulations dans 802.11 [28]

Cependant, la transmission de ces symboles sur la bande publique peut-être altérée par d'autres signaux d'interférences. Afin d'améliorer la résistance, la norme IEEE 802.11 spécifie quatre techniques de transmission des signaux : trois dans la bande **2.4 GHz** (FHSS, DSSS, HR/DSSS). Et une couche physique pour la bande des **5GHz** : OFDM (Orthogonal Frequency Division Multiplexing) seulement pour 802.11a [31].

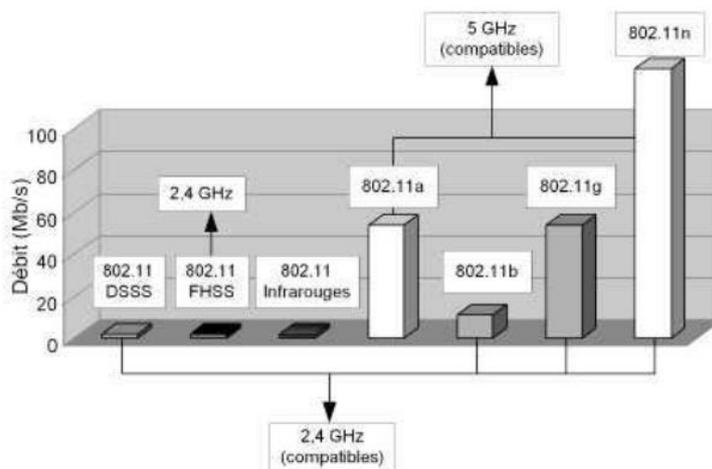


Figure 3.8. Couches physiques du WiFi : débit, fréquence et compatibilité [27].

1- Le FHSS (Frequency Hopping Spread Spectrum)

Cette technique de transmission utilise le principe de sauts de fréquences : l'émetteur passe d'une fréquence à une autre à intervalle régulier selon une règle de saut et un rythme spécifique. La bande des 2,4 GHz étant divisée en 75 sous-canaux de 1 MHz de largeur de bande, chaque sous-canal offre un débit d'au moins 1 Mbps, mais est limité à la vitesse de 2 Mbps. Ainsi, le principe de cette technique est basé sur le fait que l'émetteur et le récepteur s'accordent sur le choix d'un modèle de sauts de fréquences définissant la séquence de sauts de fréquence à réaliser afin d'envoyer les données successivement sur les différents sous-canaux. Ces modèles

sont conçus de telle sorte que la probabilité que deux émetteurs utilisent le même sous-canal simultanément soit minimisée [31]. (Voir Figure 3.9).

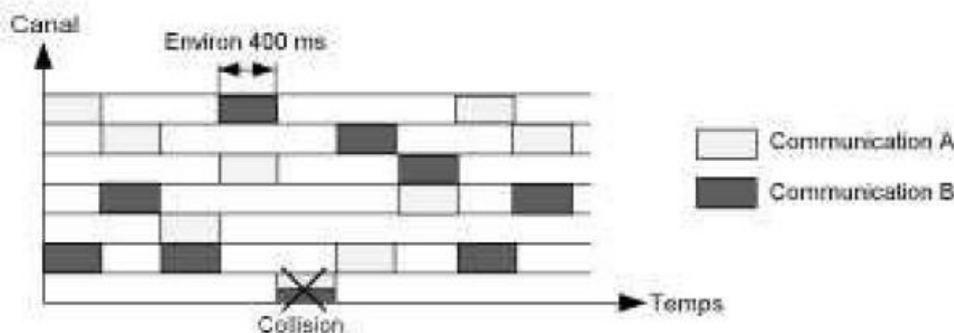


Figure 3.9. Exemple de partage des ondes avec le FHSS [27].

Le FHSS n'est utilisé que dans la première version du standard 802.11. Le **802.11a** repose exclusivement sur l'OFDM, le **802.11b** exclusivement sur le DSSS et le **802.11g** utilise le DSSS ou l'OFDM, en fonction du débit souhaité. Le 802.11n repose sur l'OFDM exclusivement lorsqu'on le règle sur un canal à 5 GHz, et sur le DSSS ou l'OFDM à 2,4 GHz.

2- Le DSSS (Direct Sequence Spread Spectrum)

DSSS (Direct Sequence Spread Spectrum), Cette technique repose sur l'élargissement de la bande utilisée par le signal utile sur une porteuse à l'aide d'une modulation de phase, afin de le rendre plus résistant [28].

Pour ce faire, l'émetteur envoie une séquence de plusieurs bits, appelés des chips, pour chaque bit d'information à transmettre. Par exemple, on peut choisir d'envoyer 11101 au lieu de 0 et son inverse (00010) au lieu de 1 : dans ce cas, si l'on veut transmettre l'information 010, alors on émettra les chips suivantes : 11101 00010 11101. Dans cet exemple, la séquence 11101 est ce qu'on appelle le « code d'étalement ». Plus ce code est long, plus le débit est artificiellement démultiplié, donc plus le spectre est étalé.

Le DSSS présente deux intérêts importants :

- tout d'abord, comme nous l'avons dit, le spectre de fréquences du signal est étalé, avec tous les avantages (et les inconvénients) que cela apporte, en particulier une meilleure résistance au bruit.
- le fait que l'on émette plusieurs chips pour chaque bit d'information signifie que l'on peut avoir une redondance importante, qui permet de corriger des erreurs de transmission [27].

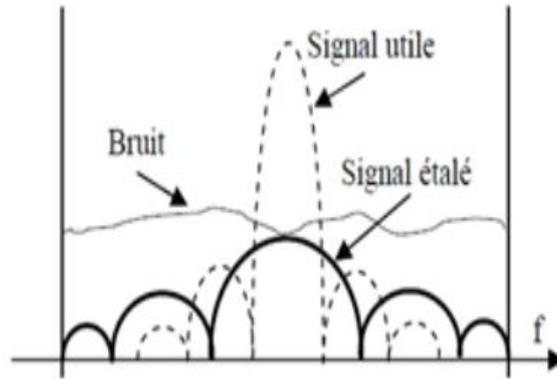


Figure 3.10. Etalement de spectre sur une porteuse [29]

3- L'OFDM

La modulation OFDM (Orthogonal Frequency Division Multiplexing), est sans doute la plus puissante des trois modulations du WiFi car elle permet à la fois les débits les plus importants, la meilleure résistance au multipath, mais aussi la plus grande capacité de partage du spectre : elle est donc particulièrement indiquée en intérieur avec une densité importante d'antennes WiFi. On la trouve à la fois dans le 802.11g, le 802.11a et dans le 802.11n. L'OFDM repose sur le principe du multiplexage : permettre la transmission simultanée de plusieurs communications sur une même bande de fréquences. Il existe le multiplexage par division des communications au cours du temps, qu'on appelle le Time Division Multiplexing (TDM) : chaque communication dispose de sa tranche de temps pour émettre des données et peut utiliser l'ensemble du spectre. Le multiplexage peut également se faire en partageant les différentes communications par fréquences : c'est le Frequency Division Multiplexing (FDM). Un spectre assez large est divisé en de multiples sous-porteuses (sub-carriers) et les données sont émises simultanément sur chaque sous-porteuse [28].

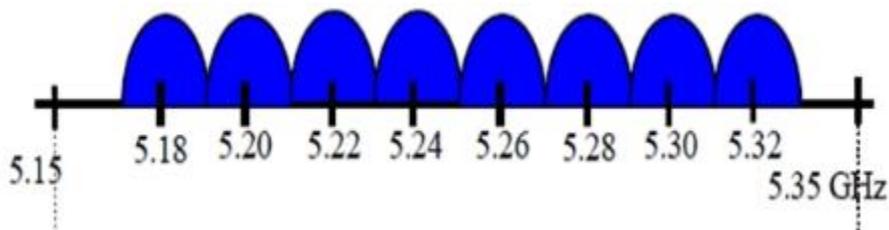


Figure 3.11. Canaux définis par OFDM [29]

3.2.2.4 Les trames 802.11

Le standard 802.11 définit trois types de trames, chaque type est subdivisé à son tour en plusieurs sous types auxquels il assigne une fonction spécifique. Ces principaux types sont :

- Les trames de données : Utilisées pour la transmission des données ;
- Les trames de contrôle : Utilisées pour contrôler l'accès au médium (RTS, CTS et ACK).
- Les trames de gestion : Utilisées pour l'échange d'informations de gestion au niveau MAC [29].

Lorsqu'un paquet de données doit être envoyé sur les ondes, le nœud commence par le traiter au niveau de la couche MAC. En bref, le paquet est éventuellement fragmenté et les fragments sont encapsulés dans des paquets appelés des *MAC Protocol Data Unit* (MPDU). La couche physique a donc pour responsabilité de transmettre sur les ondes les MPDU fournis par la couche MAC et inversement de fournir à la couche MAC les paquets reçus sur les ondes [27].

Au niveau de la couche physique, le MPDU est inclus dans une trame 802.11 dont la structure est la suivante :

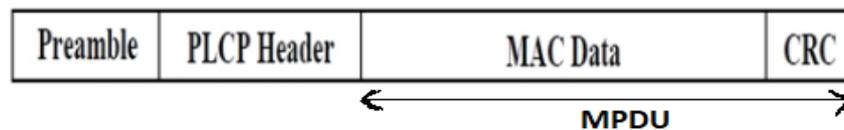


Figure 3.12. Format général d'une trame 802.11

Preamble : C'est l'entête protocolaire de la sous couche physique PMD, nommée généralement PLCP Preamble. Elle définit deux champs : **Synch** (pour la synchronisation avec le récepteur) et **SED** (contient la durée de transmission de la trame de données).

PLCP Header : C'est l'entête protocolaire de la sous couche PLCP, envoyée toujours à 1Mbit/s. Elle porte les informations indispensables pour la couche physique du récepteur afin qu'elle puisse décoder la trame. Elle est composée essentiellement des champs suivants :

Length (indique la taille de la trame), **Signal** (débit de transmission de la trame) et **CRC** (code de redondance cyclique).

MAC Data : La figure 3.13 montre le format d'une trame MAC. Elle est composée d'une entête protocolaire MAC, de données utile et un code de correction d'erreur.

CRC : C'est un champ codé sur 32 bits contenant le code de redondance cyclique [29].

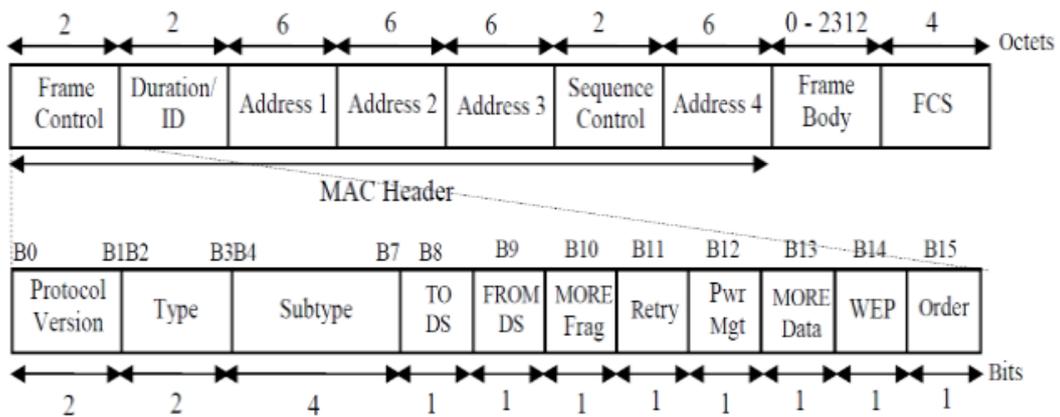


Figure 3.13 Format d'une trame 802.11 [29].

3.2.3 Couche de contrôle d'accès au medium (MAC) 802.11

Contrairement aux réseaux Ethernet qui sont pourvus d'une technologie CSMA/CD (Carry Sense Medium Access/Collision Detection) permettant aux machines d'écouter le support filaire pendant qu'elles émettent. Cette caractéristique permet de détecter les collisions. Le CSMA/CD ne peut être implanté dans un environnement radio, puisque les liaisons radio utilisées ne sont pas full-duplex (on ne peut écouter et émettre en même temps) et les WLAN n'ayant évidemment pas cette capacité. Le protocole CSMA/CA est conçu pour réduire la probabilité de collisions lorsque de multiples stations accèdent au médium. Alors la sous couche MAC de 802.11 offre une autre technique appelé CSMA/CA (Collision Avoidance) afin d'éviter les collisions et accéder au support sans fil tout en restant évolutive [22].

3.2.3.1 Méthodes d'accès au médium sans fil IEEE 802.11

Le protocole MAC de l'IEEE 802.11 incorpore deux méthodes d'accès basées sur la technique CSMA/CA. La méthode PCF (Point Coordination Function) est une méthode d'accès centralisée utilisée en mode infrastructure, elle utilise un point d'accès coordonnateur pour arbitrer le droit d'accès à tour de rôle aux stations. La méthode DCF (Distributed Coordination Function) utilisée en mode Ad hoc et fonctionne avec deux modes d'accès au médium : l'accès en mode basique (CSMA/CA) et celui en mode RTS/CTS. [22]. Dans ce qui suit nous explorons d'avantage le CSMA/CA utilisé pour le mode DCF sans et avec le mécanisme RTS/CTS.

1) Le Protocole CSMA/CA

CSMA/CA est la méthode d'accès de base du protocole MAC, son fonctionnement est décrit par les deux Figure 3.14 et 3.15. Lorsque la station A veut transmettre une trame de données, elle écoute d'abord le support. Si le support est libre pour un intervalle de temps DIFS (DCF Interframe Space), la station A transmet sa trame de données. Si la trame est reçue avec succès par la station B, ce dernier attend un temps SIFS (Short Interframe Space) et renvoie un accusé de réception Ack (Acknowledgement) à la station B.

Lorsque le support est occupé pendant l'intervalle DIFS (i.e., d'autres stations entrains de transmettre des données), la station 'A' utilise un temporisateur appelé NAV (Network Allocation Vector), lui permettant de différer ses transmissions. Ce NAV s'applique à toutes les stations et elles n'ont la capacité d'émettre qu'après la fin du NAV. Puis elle calcule un délai aléatoire appelé (Backoff) dans un intervalle $[0, CW]$ (CW étant la valeur de la fenêtre de contention) Quand le support est libre et après un nouveau DIFS, la station 'A' décrémente son temporisateur (Backoff) jusqu'à 0 puis la trame des données est transmise [22].

A la réception correcte d'une trame qui lui est destinée, une station doit envoyer un acquittement (ACK), si aucun acquittement n'est reçu par la station émettrice à l'expiration d'une temporisation interne, la station émettrice déduit qu'une collision a eu lieu. Une collision a principalement lieu lorsque la procédure de back-off aléatoire de deux stations du réseau expire en même temps. Ainsi, afin de réduire le taux de collisions, la valeur de la fenêtre de contention d'une station subissant une collision est doublée dans la limite de l'intervalle de contention $[CW_{min}, CW_{max}]$ (CW_{min} et CW_{max} étant des paramètres protocolaires décrits **ci-après**), la trame peut être retransmise jusqu'à atteindre la limite de retransmissions donnée m [26].

La Figure 3.14 représente un organigramme détaillant le déroulement de protocole CSMA/CA :

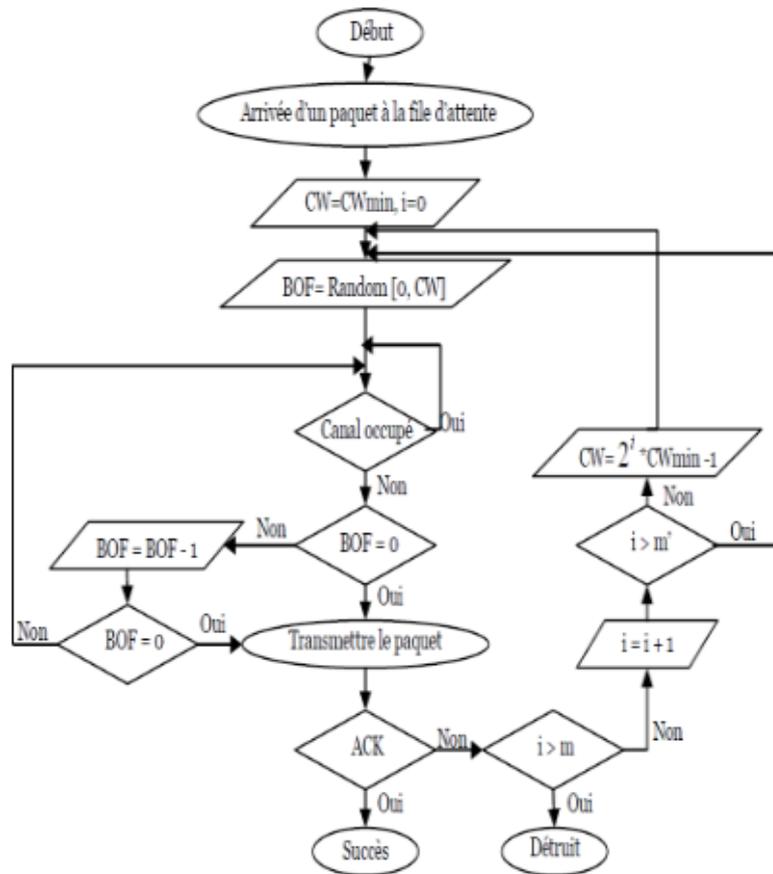


Figure 3.14 Organigramme du protocole CSMA/CA [29]

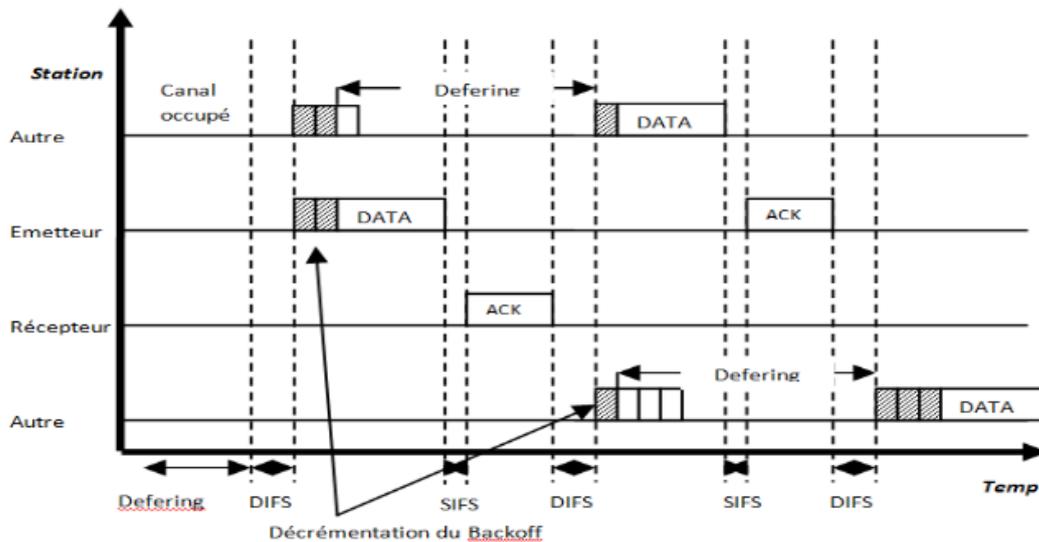


Figure 3.15. Déroulement du protocole CSMA/CA [29]

a) IFS (InterFrame Space)

Le standard définit cinq types d'espace entre trames (IFS) permettant d'instaurer des priorités entre les accès, ces temporisations sont utilisées par DCF et par les autres fonctions d'accès :

L'intervalle entre une transmission et son acquittement est fixé à **SIFS**, dont la valeur est fixée à 10µs pour 802.11-802.11b, et 16µs pour 802.11a. Dans le cas d'un mode de compatibilité pour 802.11g, **SIFS** est fixé à 10µs.

EIFS est l'intervalle correspondant à une expiration indiquant une erreur de transmission suite à non réception d'un acquittement. Cette valeur est calculée à partir de la formule suivante :

$$EIFS = SIFS + (8 * ACKSize) + Preamble + PLCPHeader + DIFS$$

L'intervalle de silence à respecter avant une tentative de transmission est fixée à **DIFS**. Le **DIFS** se calcule à partir de la durée d'une slot et de SIFS. Ainsi pour 802.11/802.11b,

$$DIFS = 2 * slot + SIFS = 2 * 20 + 10 = 50\mu s$$

Pour 802.11a : $DIFS = 2 * 9 + 16 = 34\mu s$.

Ainsi, après un intervalle DIFS (ou EIFS si la dernière trame était erronée) la station doit générer un Backoff aléatoire pour effectuer une attente supplémentaire [28]. La figure suivante montre les relations entre les différents types d'intervalle.

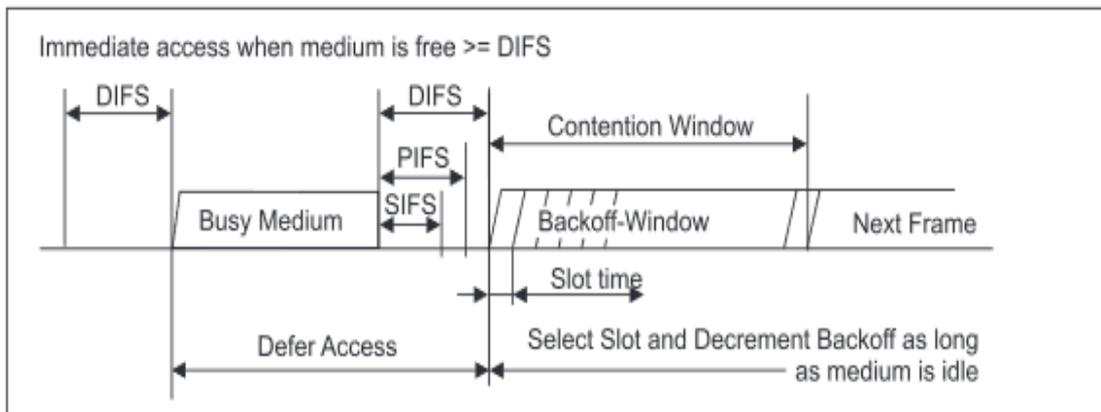


Figure 3.16 Relation entre les différentes unités temporelles IFS [28]

b) BEB (Binary Exponential Backoff)

Une procédure de backoff est mise en place suite à la détection par la fonction d'accès DCF d'un état de médium libre pour une durée supérieure à DIFS. Cette procédure permet aux stations de réduire la probabilité de collisions. Le Backoff correspond à l'attente pendant une durée aléatoire avant de procéder à l'envoi. Ce temps aléatoire (Temps De Backoff) est choisi de la façon suivante :

$$\mathbf{TempsDeBackoff = Random(0, CW) \times aSlotTime}$$

Random (a, b) étant la fonction de tirage aléatoire uniforme d'un entier dans l'intervalle [a,b]. CW étant la valeur en cours de la fenêtre de contention. La valeur de CW évolue dans l'intervalle [CWmin,CWmax] défini par le standard, la valeur de CW est initialisée à CWmin lorsqu'un paquet vient d'être envoyé avec succès ou lorsqu'un paquet est rejeté suite au dépassement de la limite de retransmissions.

Suite à une collision, la valeur de CW est augmentée de façon exponentielle jusqu'à atteindre la borne maximale CWmax afin de réduire le taux de collisions :

$$\mathbf{CW = min(CW \times 2 - 1, CWmax)}$$

Une fois la valeur du temps de Backoff tirée, elle est décrétementée de 1 à chaque slot libre observé par la fonction d'accès. Lorsque le temps de backoff atteint 0, et si le médium est toujours libre, la fonction d'accès tente l'envoi sur le médium.

Si en cours de décrémentation du temps de backoff le médium devient occupé, la valeur en cours du temps de backoff est mémorisée, la décrémentation reprendra au point où elle s'était arrêtée lorsque la fonction observera à nouveau un intervalle DIFS d'inoccupation du médium [26].

En 802.11/802.11b, CWmin est fixé à 31 slots, en 802.11a/g CWmin est fixé à 15 slots. La valeur de backoff maximale est fixé à CWmax=1024slot [28]. L'évolution des valeurs de CW est visible sur la figure 3.17.

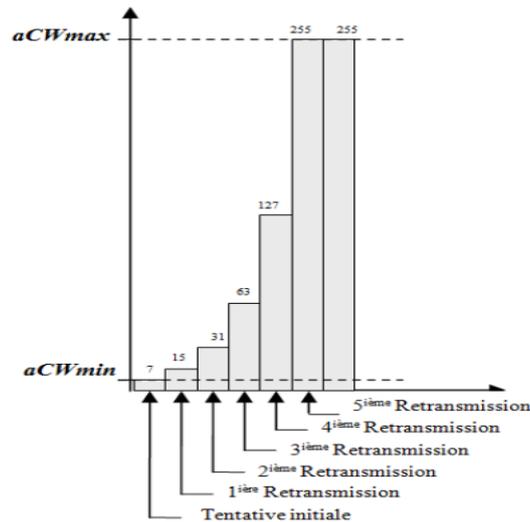


Figure 3.17 Evolution de la Contention Window CW selon le BEB [29].

2) Le mécanisme RTS/CTS

Le mécanisme est effectué à travers une opération de réservation du canal annonçant l'usage du médium aux autres stations. Un échange de trame Ready-To-Send (RTS) et Clear-To-Send (CTS) est effectué avant l'envoi des données. Les trames RTS/CTS contiennent un en-tête 802.11 indiquant la Durée (Duration) et l'Identification (ID) qui définit la période de temps pour laquelle le médium doit être réservé, ce qui correspond à l'envoi des données et à la réception de l'acquittement. Toutes les stations qui sont à portée de réception de la station émettrice (qui envoie le RTS) ou de la station réceptrice (qui envoie le CTS) peuvent ainsi obtenir l'information sur l'usage du canal. Dans ce cas, les stations mettent à jour un vecteur d'allocation réseau (Network Allocation Vector - NAV) qui maintient en mémoire la prédiction du trafic futur sur le réseau. L'échange RTS/CTS est détaillé sur la figure ci-dessous. [28].

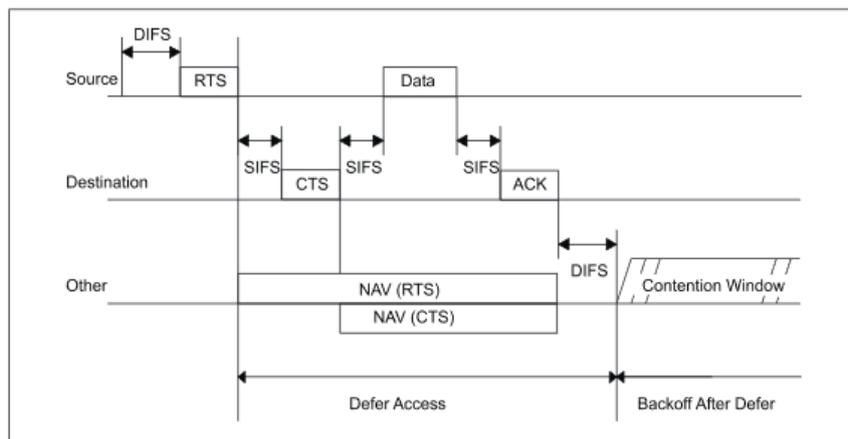


Figure 3.18 Mécanisme d'accès RTS/CTS [28].

Ce mécanisme RTS/CTS est plutôt utilisé pour l'envoi de longues trames qui feraient chuter les performances en bande passante en cas de collision. On peut justement calculer un seuil permettant d'utiliser ou non le mécanisme RTS /CTS (RTS Threshold). Une deuxième application, très utile pour l'univers des liaisons radio, est la détection des stations cachées et des stations exposés. [22].

a) Problème de la station cachée

Deux stations peuvent être hors de portée radio l'une de l'autre tout en étant dans le même IBSS. Si elles veulent émettre en même temps il va y avoir collision. [22].

- (1) 'A' transmis à 'B'
- (2) 'C' ne détecte pas la transmission de 'A', et transmis à 'B'
- (3) l'interférence se produit au niveau de 'B' : donc collision

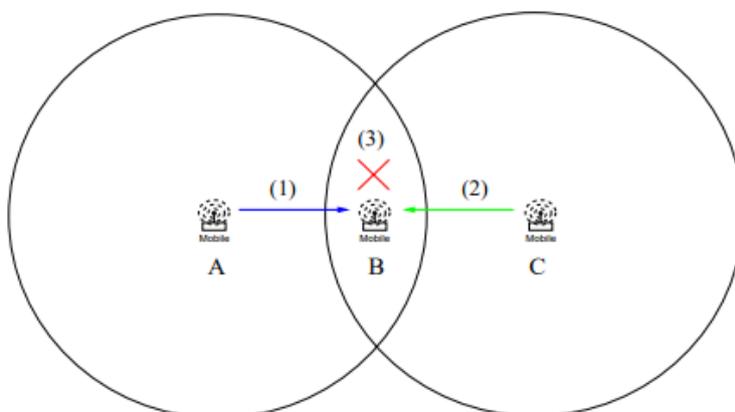


Figure 3.19 Problème de la station cachée [41].

b) Problème de la station exposée

Considérons maintenant une simulation qui est l'inverse de la précédente : le problème de la station exposée. Dans ce problème, une station s'abstient d'utiliser un canal alors qu'il est en fait disponible. Sur la figure 3.20, la station A transmet à la station B, et la station C a des données à envoyer à la station D qui peuvent être envoyées sans interférer avec la transmission de A à B. Cependant, la station C est exposée à la transmission de A, elle entend ce que A envoie et s'abstient donc d'envoyer. En d'autres termes, le nœud 'C' est trop conservateur et gaspille la capacité du canal [18].

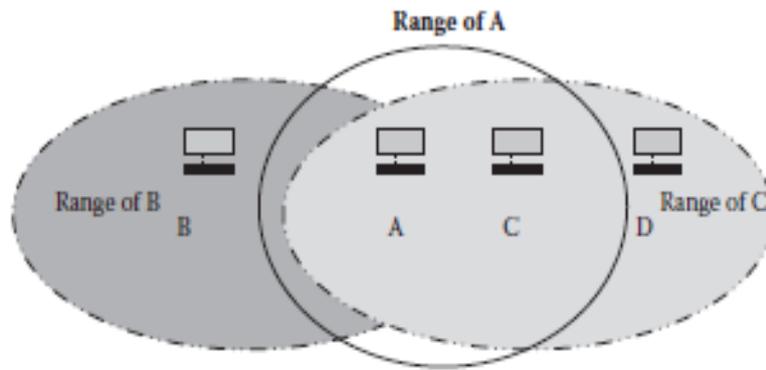


Figure 3.20. problème de la station exposée [18]

3.3 Conclusion

Dans ce chapitre, nous avons présenté les principaux prérequis nécessaires à notre travail, où nous avons présenté l'architecture protocolaire de 802.11, ses différentes couches physiques et essentiellement le mode de fonctionnement DCF du standard IEEE 802.11, la sous couche MAC ainsi que les différents protocoles d'accès au médium de transmission.

Le mode DCF que procure le standard IEEE 802.11 en fait que cette norme est très répandue commercialement et dans le domaine de la recherche scientifique. Dès lors, la modélisation et la simulation de ce type de d'architecture à savoir les réseaux Mobiles Ad hoc s'avèrent indispensables pour le dimensionnement et le déploiement réel des réseaux sans fil robustes et fiables. Le prochain chapitre fera l'objet d'une synthèse sur les principaux simulateurs utilisés pour les réseaux sans fils.

Chapitre 4 *Simulation et simulateurs réseaux*

4.1 Introduction

Les réseaux sans fil connaissent actuellement une très grande évolution. Le besoin, en termes de nouvelles méthodes d'évaluation de performances est par conséquent croissant. Actuellement, trois méthodes principales peuvent être adoptées pour analyser les performances des réseaux : les études analytiques, les expérimentations et la simulation, qui est la méthode la plus répandue et la plus pratique pour évaluer les performances des réseaux communication.

4.2 La simulation informatique

4.2.1 Définition de la simulation

La simulation par ordinateur est la discipline consistant à concevoir un modèle d'un système physique réel ou théorique, à exécuter le modèle sur un ordinateur numérique et à analyser le résultat de l'exécution [32].

[32] définit ainsi cette discipline comme un processus itératif non linéaire composé de 3 tâches fondamentales, fortement interdépendantes :

- Elaboration du modèle.
- Exécution du modèle sur ordinateur.
- Analyse de l'exécution du modèle et des résultats obtenus.

Pour détailler d'avantage le processus d'une simulation informatique, la Figure 4.1 distingue les étapes du processus d'une simulation :

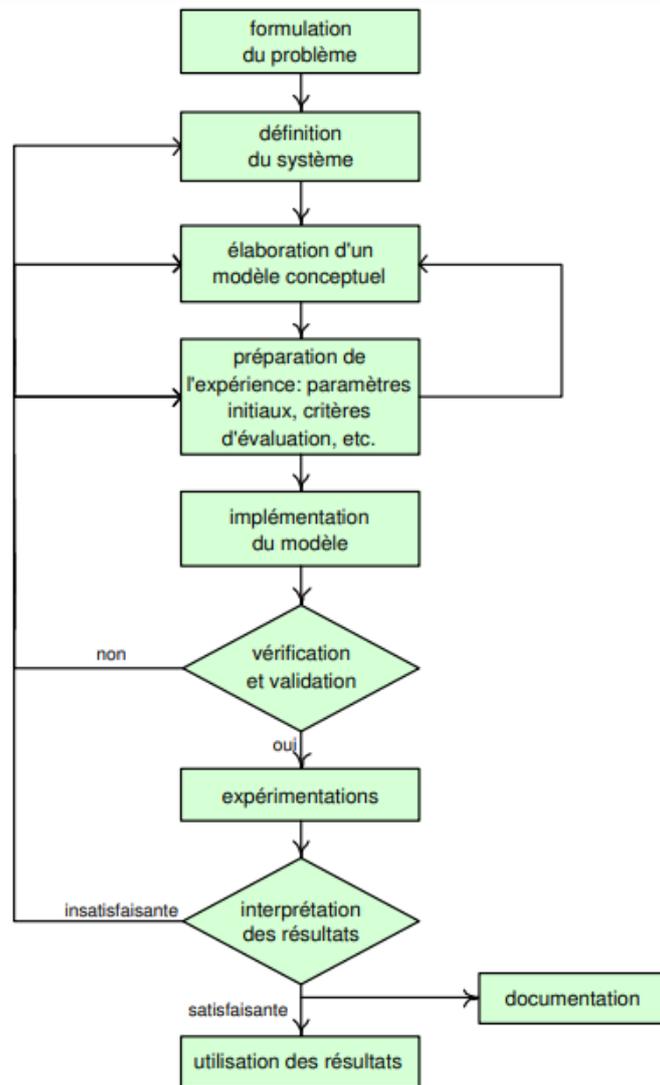


Figure 4.1 Etapes de conception d’une simulation informatique [32]

4.2.2 Types de la simulation

Lorsque la simulation s’avère nécessaire pour évaluer un système réel, quatre principales méthodes de simulation peuvent être utilisées en fonction de la nature du système cible :

- A. La Simulation continue :** Elle permet d’analyse de manière continue le comportement d’un système, représenté sous la forme d’équations différentielles, au cours du temps.
- B. La simulation discrète :** Permet d’évaluer le comportement d’un système au cours du temps grâce à la génération d’événements.

C. La Simulation analytique : Elle permet d'analyser des processus stochastiques à travers lesquels le système peut passer par différents états, comme par exemple les chaînes de Markov [33].

Un autre type de La Simulation est celui orienté agents : les modèles à base d'agents sont généralement caractérisés par de nombreux paramètres qui déterminent la dynamique globale du système.

4.2.3 Intérêts de la simulation

Parmi les intérêts de la simulation :

- Quand on ne peut pas facilement observer les états du système.
- Quand on désire analyser l'enchaînement des événements dans le système, ainsi que les relations de causes à effets.
- Quand on désire valider une solution analytique.
- Quand la complexité des interactions dans le système est telle qu'elle ne peut être étudiée qu'au travers de simulations.
- Quand on désire visualiser les états d'un système, afin d'étudier son comportement dans un contexte de surveillance ou d'aide à la décision.
- Quand on veut tester différentes optimisations pour améliorer un système déjà existant [42].

4.2.4 Définition de la simulation d'un réseau

La simulation d'un réseau de communication est une approche couramment utilisée dans la conception, la mise en œuvre, l'optimisation et l'évaluation des algorithmes et protocoles de réseau, modélise le comportement d'un réseau réel fonctionnant sous diverses configurations. Cela permet aux chercheurs d'exécuter des expériences de manière contrôlée et reproductible sans le temps et les frais de mise en place d'un banc d'essai de réseau réel avec des nœuds, des liens et des dispositifs réels pour mesurer les paramètres du réseau.

4.2.5 Contraintes et défis de la simulation dans les réseaux sans fil

La simulation des réseaux sans fil introduit de nombreuses problématiques liées à la modélisation des nœuds, du médium radio et de l'environnement physique. Ainsi, un simulateur doit prendre compte de nombreux paramètres afin que la modélisation s'approche le maximum du system réel. Les différentes fonctionnalités que doit offrir un simulateur pour réseaux sans fil sont les suivants :

- Une modélisation complète de l'architecture des nœuds, à s'avoir les différentes couches du modèle OSI (application, routage, MAC. . .).
- Une modélisation des composants matériels des nœuds son fil comme la modélisation d'une batterie ou d'un processeur.
- Une modélisation détaillée du médium radio afin de prendre en compte les différents phénomènes liés à la propagation, aux collisions et aux interférences.
- Une modélisation de l'environnement physique. Par exemple, simuler la présence d'obstacles [33].

4.2.6 Comparaison entre les simulateurs traditionnels des réseaux sans fils

Il existe de nombreux simulateurs de réseau dans le monde de la communication. Certains d'entre eux sont dédiés au réseau sans fil, certains sont dédiés au réseau filaire ou aux deux types de réseaux. En raison des grandes variations des systèmes d'exploitation, des exigences matérielles, des exigences des logiciels de programmation, des fonctionnalités de sortie et de l'évolutivité, il est très difficile de choisir un simulateur approprié pour un travail spécifique [34]. Dans cette section, nous avons étudié et comparé les caractéristiques détaillées de certains simulateurs traditionnels des réseaux à savoir : NS2, NS3 et OMNET++.

4.2.6.1 NS2

NS2 (Network Simulator Version-2) est un simulateur open source conçu spécialement pour la recherche sur les réseaux informatiques. Il est disponible sur les plates-formes UNIX, Free BSD et Windows OS. NS2 effectue une simulation pour explorer différents problèmes tels que l'interaction de protocole, le contrôle de la congestion, l'effet de la dynamique du réseau, l'évolutivité... [34]. Ce simulateur est plutôt utilisé pour la simulation du routage et de protocoles d'émission/réception et surtout pour la recherche dans les réseaux ad-hoc. En effet,

ns supporte plusieurs protocoles réseaux et permet la simulation de réseaux sans fils et câblés aussi [33].

Le simulateur NS2 est basé sur deux langages : un simulateur orienté objet, écrit en C ++, et un interpréteur OTcl (une extension orientée objet du langage Tcl) pour exécuter des scripts de commande [36]. NS2 fournit un grand nombre d'objets C ++ intégrés, utilisés pour configurer une simulation à l'aide d'un script de simulation écrit en Tcl. Il génère des résultats de simulation textuels ou animés. Pour interpréter ces résultats graphiquement et interactivement, des outils tels que NAM (Network AniMator), XGraph et Gnuplot sont utilisés [34].

- **Avantages**

- NS2 dispose d'un grand nombre de modèles disponibles, de modèles de mobilité réalistes, d'une configuration de script et de simulation puissante et flexible, d'une large communauté d'utilisateurs et d'un développement continu.
- NS2 fournit un modèle de circulation et de mouvement facile en incluant un modèle énergétique efficace.
- Les utilisateurs peuvent développer leurs propres objets C ++.

- **Limites**

- 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 tel que le langage OTCL
- L'inclusion d'un grand nombre de nœuds peut ralentir le processus de simulation.

4.2.6.2 NS3

Le Ns-3 est un simulateur réseau open source, construit en remplacement du simulateur de réseau bien connu ns-2, et le développement a commencé en 2006. L'objectif du projet était de mettre à disposition une plate-forme de simulation de réseau extensible pour l'enseignement et la recherche. Les scripts de simulation peuvent être écrits en C ++ ou Python. Actuellement,

ns-3 est activement développé et a un impressionnant nombre de modèles validés de protocoles. Ces derniers comprennent plusieurs modules de routage (AODV, DSDV, DSR et OLSR), module de modélisation de la consommation d'énergie, module pour les nœuds mobiles, modules pour la perte de propagation et le retard de propagation [35].

NS3 fournit une bibliothèque solide qui est utile à l'utilisateur pour faire son travail en éditant NS-3 lui-même. Pour la technologie filaire, le NS-3 fournit un modèle d'appareil d'un simple réseau Ethernet qui utilise CSMA / CD comme schéma de protocole avec une augmentation exponentielle du back-off pour lutter contre le support de transmission partagé. Il fournit également un ensemble de modèles 802.11 qui tentent de fournir une implémentation précise au niveau MAC de la spécification 802.11 et un modèle de niveau PHY de la spécification 802.11a [34].

- **Avantages**

- Grande modularité que son ancêtre NS2.
- Prise en charge de la simulation pour TCP, UDP, ICMP, IPv4, routage multidiffusion, protocoles P2P et CSMA.
- Beaucoup plus flexible que tout autre simulateur.

- **Limites**

- NS3 souffre toujours d'un manque de crédibilité.
- Les liaisons Python ne fonctionnent pas sur Cygwin.
- NS3 a besoin de beaucoup de mainteneurs spécialisés afin de profiter de ses avantages.

4.2.6.3 OMNET++

OMNeT ++ (Objective Modular Network Tested in C ++) est un outil de simulation d'événements discrets open source, extensible, modulaire, permet de simuler des réseaux câblés et sans fil. Il est principalement utilisé à des fins de recherche et d'enseignement dans la communauté scientifique mondiale. Des fonctionnalités spécifiques au domaine telles que la prise en charge des réseaux de capteurs, des réseaux ad hoc sans fil, des protocoles Internet, de la modélisation des performances, des réseaux photoniques, etc. y sont disponibles. Les

composants et les modules sont programmés en C ++, puis assemblés en composants et modèles plus grands à l'aide d'un langage de haut niveau (NED). Il offre un IDE basé sur Eclipse, un environnement d'exécution graphique et une foule d'autres outils [34].

- **Avantages**

- Modélise avec précision la plupart des matériels et inclut la modélisation des phénomènes physiques
- Il fournit également une puissante bibliothèque d'interfaces graphiques pour l'animation et la gestion du débogage [33].
- Architecture modulaire permettant l'intégration de nouveau modèle.

- **Limites**

- Le seul protocole MAC fourni pour les réseaux sans fil est 802.11.
- OMNeT ++ est un peu lent en raison de sa longue simulation et de sa consommation élevée de mémoire
- L'extension de mobilité pour OMNeT ++ qui prend en charge les simulations sans fil et mobiles serait assez incomplet [23].

Fonctionnalités	NS2	NS3	OMNET++
Langage	C++/OTCL	C++, Python	C++
Plateforme	Linux, Unix, Windows, Cygwin	Linux, Unix, Windows	Linux, Unix, Windows, MAC OS
Interface	Limité	Bon	Bon
Facilité d'utilisation	Difficile	Difficile	Facile
Architecture réseau pris en charge	Prend en charge la simulation filaire et sans fil des protocoles	Prend en charge la simulation filaire et sans fil des protocoles	Prend en charge la simulation filaire et sans fil des protocoles

Capacités de simulation rapides	Modéré	Modéré	Modéré
Mobilité	Oui	Oui	Non

Tableau 4.1 Tableau comparatif entre les simulateurs réseau traditionnels.

4.3 Simulation à base d'agents

4.3.1 Modélisation et la simulation (M&S) basées sur des agents

La modélisation basée sur les agents (MBA) est un type de modélisation informatique pour simuler les actions et les interactions des agents autonomes. Ces agents peuvent être des entités individuelles ou collectives telles que des organisations ou des groupes, allant des réseaux sociaux, des groupes aux systèmes auto-organisés et aux modèles bio-inspirés [37].

MBA peut être utilisé pour développer des agents proactifs, autonomes et intelligents. La proactivité fait référence à la capacité des agents à prendre l'initiative, leur permettant de communiquer avec d'autres agents et de prendre leurs propres décisions.

4.3.2 Structure d'un modèle basé sur un agent

Un modèle basé sur un agent comporte généralement quatre aspects :

- **Un ensemble d'agents autonomes** - chaque agent possède un ensemble d'attributs qui décrivent l'état de l'agent et un ensemble de comportements spécifiés (règles qui régissent le comportement de l'agent) qui définissent le comportement d'un agent en réponse aux changements de son environnement et, peut-être, vers un ensemble de buts ou d'objectifs
- **Un ensemble de relations d'agent** : chaque relation définit la façon dont chaque agent interagit avec les autres agents et son environnement. Cela implique également la façon dont chaque agent est «connecté» aux autres agents.
- **L'environnement de l'agent** - le «monde» dans lequel les agents existent, c'est-à-dire l'ensemble minimal de variables ou de structures «globales» nécessaires pour définir la façon dont les agents réagissent à leur environnement.

- Un «système» : il est composé de l'ensemble des agents, de l'environnement et de leurs relations. Un système a une frontière clairement définie avec des entrées et des sorties bien définies [40].

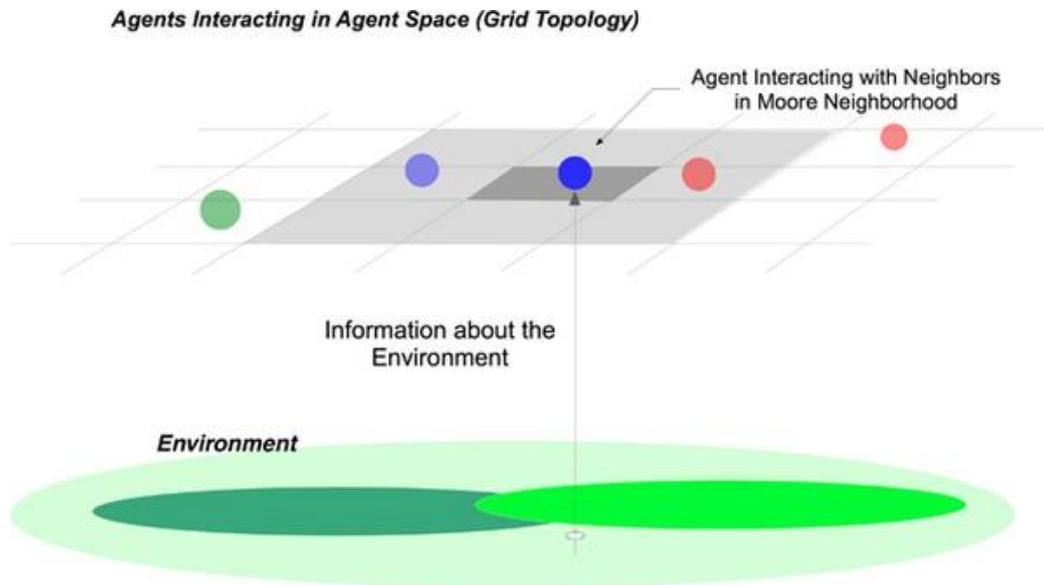


Figure 4.2. Structure d'un modèle basé sur un agent [40]

4.3.3 Conception du modèle d'agent

Lors du développement d'un modèle basé sur un agent, il est utile de poser une série de questions, dont les réponses conduiront à une conception initiale du modèle :

- Quel problème spécifique devrait être résolu par le modèle ? À quelles questions spécifiques le modèle devrait-il répondre ? Quelle valeur ajoutée la modélisation basée sur les agents apporterait-elle au problème que d'autres approches de modélisation ne peuvent pas apporter ?
- Quels devraient être les agents dans le modèle ? Qui sont les décideurs du système ? Quelles sont les entités qui ont des comportements ? Quelles données sur les agents sont simplement descriptives (attributs statiques) ? Quels attributs d'agent seraient calculés de manière endogène par le modèle et mis à jour dans les agents (attributs dynamiques) ?
- Quel est l'environnement des agents ? Comment les agents interagissent-ils avec l'environnement ? La mobilité d'un agent dans l'espace est-elle un élément important de considération ?

- Quels comportements des agents sont intéressants ? Quelles décisions les agents prennent-ils ? Quels comportements sont mis en œuvre ? Quelles mesures sont prises par les agents ?
- Comment les agents interagissent-ils entre eux ? Avec l'environnement ? Quelle est l'ampleur ou la concentration des interactions avec les agents ?
- D'où pourraient provenir les données, notamment sur les comportements des agents, pour un tel modèle ?
- Comment pourriez-vous valider le modèle, en particulier les comportements des agents ? Répondre à ces questions est un élément essentiel du processus de conception d'un modèle basé sur un agent. Il existe différentes approches pour concevoir et mettre en œuvre des modèles basés sur des agents [40].

4.3.4 Outils basés sur l'agent

Dans le domaine des outils basés sur des agents, un certain nombre d'outils populaires sont disponibles. Celles-ci vont des outils basés sur Java tels que Mason aux outils basés sur Logo tels que : StarLogo, NetLogo etc. Chacun de ces outils a des forces et des faiblesses différentes [37].

4.3.4.1 NETLOGO

NetLogo est un langage de programmation multi-agents basé sur le langage de programmation Logo. Dans sa version actuelle, NetLogo est distribué en open source. NetLogo a été conçu et développé dès le départ pour la modélisation et la simulation de systèmes complexes évoluant dans le temps. NetLogo a été utilisé pour modéliser et simuler divers systèmes complexes dans divers domaines tels que l'économie, la biologie, la physique, la chimie, la psychologie, la dynamique des systèmes et de nombreuses autres sciences naturelles et sociales [38].

Cela a inspiré certains chercheurs à simuler et analyser des systèmes distribués tels que les réseaux mobiles ad hoc et les réseaux sans fil modernes. Un package NetLogo est livré avec une bibliothèque de code intégrée contenant de nombreux exemples, avec un code bien documenté pour aider l'utilisateur à facilement modifier et construire sa simulation couvrant de nombreux sujets [39].

Le monde NetLogo se compose d'une interface composée de «patches». Les habitants de ce monde peuvent aller des tortues aux liens. De plus, on peut avoir des éléments d'interface utilisateur tels que des boutons, des curseurs, des moniteurs, des tracés, etc.

4.3.4.2 Pourquoi le SMA NETLOGO ?

En comparant le simulateur NETLOGO aux simulateurs traditionnels des réseaux utilisés dans le domaine de la recherche cités auparavant, nous constatant que les simulateurs réseaux existant sont conçus pour manipuler principalement les réseaux filaires statique, mais sont très limités dans la modélisation des systèmes complexe modernes comme les réseaux MANETs.

MANETs, contrairement aux réseaux filaires, leur conception est basée sur de multiples paramètres qui permettent de gérer des métrique complexe comme (la transmission radio, la mobilité, le routage et l'accès au canal radio dans un environnement mobile...). De plus, en raison de leur complexité dans les détails et terme de programmation, les simulateurs traditionnels ne fournissent pas un environnement convivial pour les étudiants et les chercheurs.

Contrairement aux simulateurs traditionnels, NETLOGO, lui, semble être plus adapté pour simuler des MANETs à grand échelle. Autre que sa facilité d'utilisation, NETLOGO est caractérisé par :

- **Plage de valeurs d'entrée :** NetLogo permet la création de paradigmes complètement nouveaux de modélisation de réseau, dans lesquels le spécialiste M&S peut se concentrer, par exemple, sur des aspects purement d'auto-organisation ou sur le développement de diagrammes d'antenne et de modélisation de propagation - directement dans NetLogo, une tâche relativement triviale en soi.
- **Gamme de statistiques :** NetLogo est assez flexible en termes de statistiques et de mesures. Toute variable d'intérêt peut être ajoutée en tant que variable globale et des statistiques peuvent être générées sur la base d'une ou de plusieurs exécutions. Des tracés peuvent également être générés automatiquement pour ces variables.
- **Gestion des métriques complexes :** Les mesures de termes complexes dans les programmes NetLogo ne sont limitées que par l'imagination du spécialiste M&S. Presque tous les concepts, qui peuvent être considérés comme importants, peuvent facilement être

ajoutés au modèle. Par défaut, NetLogo fournit une analyse en temps réel. Des variables ou des rapporteurs (fonctions qui renvoient des valeurs) peuvent être utilisés pour mesurer des paramètres en temps réel et l'analyste peut en fait avoir une session interactive avec le système modélisé sans modifier le code à l'aide de la «fenêtre de commande» [37].

4.4 Conclusion

Dans le contexte de notre travail, et d'après les limites aperçus dans les simulateurs réseaux traditionnels, la plupart des simulateurs d'événements discrets ne sont pas conçues pour traiter les modèles complexes car le nombre d'événements peut être augmenté de façon exponentielle, ce qui rend le modèle inefficace et difficile à analyser. La modélisation basée sur les agents fournit une description des systèmes complexes avec un niveau d'abstraction plus élevé avec une meilleure compréhension du paradigme de communications inter agents.

De ce fait, l'utilisation d'outils de modélisation basés sur des agents génériques et très flexibles peuvent apporter une valeur ajoutée dans la modélisation et simulation de réseaux de communication tels que le P2P, les réseaux sans fils, notamment les réseaux MANET. Cet avantage nous a motivé pour s'orienter vers ce type de simulation, et de choisir l'outil NETLOGO pour concevoir notre modèle et simuler des aspects de haut niveau des réseaux mobiles ad hoc

Chapitre 5 *Simulation d'un réseau MANET sur Netlogo*

5.1 Introduction

Après avoir fait la présentation de la première partie de notre travail qui touche aux aspects théoriques sur lesquels notre projet est basé, à savoir les systèmes multi agents, les réseaux Ad hoc, la norme IEEE 802.11 PHY/MAC ainsi que la simulation à base d'agents. Ce chapitre vise à décrire en premier lieu les concepts de programmation Netlogo, pour ensuite décrire les étapes qu'on a suivies pour la conception et la mise en œuvre de notre modèle MANET sur le simulateur cité au-dessus.

5.2 Présentation du modèle simulé

5.2.1 Objectifs du modèle

L'objectif principal de notre travail est de pouvoir simuler le comportement d'un réseau mobile Ad hoc MANET. Ces réseaux se caractérisent par une très grande taille, ils sont fortement hétérogènes en termes de technologies de communication (protocoles, services), et très dynamiques en raison des changements continus de la topologie, du nombre d'utilisateurs actifs etc.

Notre modèle vise à simuler en particulier les notions et les protocoles qui constituent les deux couches basses du modèle OSI, à savoir les couches physique et Mac utilisées pour les réseaux sans fils IEEE 802.11, en utilisant un simulateur multi agent NETLOGO, et d'adapter son code source en cas de non satisfaction de nos besoins.

D'une manière générale, le modèle devra contenir un ensemble d'agents de tortues qui peuvent agir soit comme des nœuds de réseau, soit comme des événements d'un réseau sans fil, les liens (entre tortues) modélisent la connectivité radio entre les nœuds du réseau, toute paire de nœuds connectés par le même lien peut communiquer entre eux. Chaque nœud possédera une portée radio (un rayon de transmission), envoie des ondes radios lors d'une transmission.

En outre, chaque nœud formant le réseau sera en capacité de générer un message pour l'envoyer soit en unicast (une destination spécifique), soit en broadcast.

Le protocole d'accès au canal radio CSMA/CA 802.11 DCF sera implémenter afin de gérer les transmissions simultanées des nœuds se trouvant dans la même portée radio et de résoudre le problème de collisions (envoies simultanés, station cachée).

Le modele doit aussi assurer le routage des messages d'un nœud source vers un nœud destination si les deux extrémités ne font pas partie d'une même IBBS (zone de couverture radio).

5.2.1.1 Conception du modèle simulé

Dans notre modèle, nous avons implémentés les paramètres pertinents qui définissent une transmission radio, et les protocoles nécessaires afin de gérer la compétition entre les nœuds pour accéder au canal ainsi que l'acheminement des messages entre les mobiles.

5.2.1.2 Paramètres de la transmission radio simulés

a) Portée radio : La portée radio d'un nœud désigne la zone de couverture dans laquelle il peut transmettre un signal et qu'un autre nœud peut le recevoir avec succès s'il n'y a pas d'interférences. Cette étendue aussi appelé rayon de transmission dépend de la puissance de transmission et de l'atténuation du signal. Dans notre modèle, la tache de la portée radio sera affectée à un Agent représentée sous forme d'un cercle.

Les nœuds qui se trouvent dans la même zone de couverture, des liaisons sans fil seront créés entre eux. Les liens seront représentés par des agents sous forme d'une droite.

b) Mobilité des nœuds (changement de topologie) : La topologie du réseau peut changer à tout moment (voir la figure 5.2), elle est donc dynamique et imprévisible ce qui fait que la déconnexion des agents (nœud) mobiles soit très fréquente. La figure 5.2 nous montre un exemple de changement de topologie entre deux périodes de temps t1 et t2.

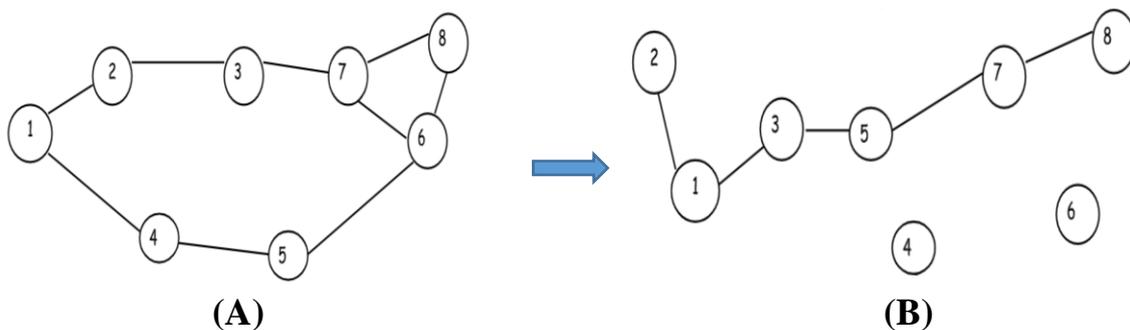


Figure 5.1 Changement de topologie dans réseau MANET

Dans notre modèle, la mobilité des Agents (nœud) sera représentée un déplacement aléatoire d'un agent dans l'environnement suivant une direction choisi aléatoirement entre 0° et 360°.

c) Les ondes radio : Les ondes radio sont des ondes électromagnétiques. Pour générer ou recevoir de telles ondes, on utilise un équipement appelé antenne. Le passage d'un courant dans l'antenne va générer une onde électromagnétique qui va se propager dans tout l'espace. Une antenne réceptrice va être en mesure de capter l'onde 'électromagnétique et de traduire l'onde reçue en un courant qui peut ensuite être numérisé.

Dans notre modèle, lors d'une transmission, les ondes radios seront représentées par un agent sous forme d'un cercle de taille 0, il prendra la même position d'un nœud et s'agrandira 'jusqu'à atteindre la limite du rayon de transmission de ce dernier.

5.2.1.3 Protocoles simulé dans le model

a) Protocole d'accès au medium MAC

Afin de gérer les transmissions simultanées des nœuds qui se trouve dans la même zone de couverture et d'éviter les collisions répétées menant à la perte de paquets, nous avons implémentés l'algorithme de CSMA/CA 802.11 DCF.

Les agents impliqués dans cette tache sont : **les nœuds, les zones de portées et les liens.**

L'algorithme CSMA/CA implémenté :

1. **tant que** (le mobile est allumé) **faire**
2. $r \leftarrow 0;$
3. $CW \leftarrow CW_{min};$
4. $B \leftarrow \text{random}[0, CW];$
5. Message_envoyé \leftarrow faux
6. **Tant que** (*buffer_paquets* non vide) **faire**
7. **Répéter**
8. **Tant que** ($B > 0$) **faire**
9. **Si** (le canal est libre) alors
10. $B \leftarrow B - 1;$
11. Tentative de transmission du 1^{er} paquet du *buffer_paquets* ;

```

12. | | | Si (aquittement recus) alors
13. | | |   Message_envoye ← vrais   ; buffer_paquet ← buffer_paquets -1 ;
14. | | |   Collision (plus d'un nœuds transmis simultanément, ou problème
15. | | |   station caché : paquet éronné, taux d'erreur élevé) ;
16. | | |   r ← 0;
17. | | |   CW ← CWmin
18. | | | Sinon
19. | | |   r ← r+1
20. | | |   B ← 2 * CW ;
21. | | |   Si (B >= CWmax) alors
22. | | |     CW ← CWmax
23. | | |     B ← random[0, CW]
24. | | | Jusqu'a (r = R) ou (message_envoyé = vrais);
25. | | | Si (r >= R) et (message_envoyé = faux) alors
26. | | |   Detruire le paquet ; buffer_paquets ← buffer_paquets - 1 ;
27. | | |   r ← 0 ;
28. | | |   CW ← CWmin ;

```

r indique le nombre de tentatives de retransmission, tandis que **R** est la limite maximale de tentatives de retransmission. Lorsque les retransmissions **R** sont atteintes, le paquet en attente de transmission est abandonné.

b) Le routage au plus court chemin

Pour routage, nous avons optés pour le routage au plus court chemin. C-a-d : les nœuds qui constituent le chemin sur lequel un message est routé d'une source à une destination sont les nœuds qui possèdent la plus courte distance vers le nœud destination, chacun dans par rapport aux autres nœuds avec qui il partage la IBBS.

Notre approche est inspirée de l'algorithme de Djikstra illustré dans la figure ci-dessous :

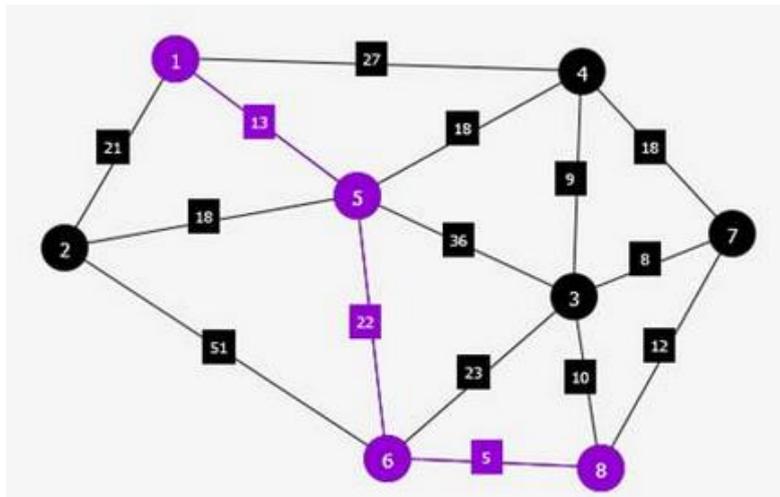


Figure 5.2 Dijkstra : le plus court chemin

Principe de l'approche

Remarque : Sachant que les nœuds du réseau sont en **mobilité** et le protocole CSMA/CA n'est pas en considération.

Debut

Nœud_source : génération d'un message

msg_envoyé \leftarrow false

Tant que (msg_envoyé = false) **faire**

Si (le nœud destination se trouve sa portée radio du **nœud_source**) **alors** :

Envoie direct du message ; **msg_envoyé** \leftarrow true ;

Sinon

Si (il existe d'autres nœuds dans la portée radio de **nœud_source**) **alors** :

Choisir le nœud qui se trouve à une destination la plus courte du destinataire. ;

Ce nœud sera designer comme un **nœud_Relay** (intermédiaire), le message lui sera envoyé, et deviendra à son tour le **nœud_source**, toute en gardant trace du nœud qui a générer le message la 1^{er} fois ;

Sinon

Le **nœud_source** garde le message dans le buffer et attend la détection d'un nœud qui rejoindra sa zone de couverture ;

Fin

5.3 NETLOGO

5.3.1 Concepts de programmation en NetLogo

Bien qu'il n'y ait pas de règles réelles pour créer un programme NetLogo, on pourrait concevoir un programme pour avoir un ensemble de procédures qui peuvent être appelées directement depuis le centre de commande. Cependant, dans la plupart des cas, il suffit d'avoir des boutons d'interface utilisateur pour appeler des procédures. Ceux-ci peuvent être insérés facilement en cliquant avec le bouton droit n'importe où sur l'écran d'interface et en sélectionnant des boutons, en pensant à écrire les noms des boutons dans les commandes et les affecter aux procédures. Voici les éléments qui nous permettent de créer un programme dans Netlogo [43] :

- **Les Agent**

Le simulateur Netlogo comporte quatre type d'agents, et sont les suivants :

- ✓ **Patches** : sont les composantes spatiales du «monde» modélisé. Un patch représenté par un seul endroit « pièce » où la tortue existe.
- ✓ **Les tortues** : Les habitants clés du monde Logo sont les tortues qui, de notre point de vue de conception de réseaux, peuvent être utilisées pour modéliser facilement des nœuds de réseau
- ✓ **Liens** : un type particulier d'agent qui relie deux agents et est représenté comme une ligne tracée entre ces agents. Ce lien peut être orienté ou non.
- ✓ **L'Observateur** : appelé l'observateur car il peut être utilisé dans une simulation interactive où l'utilisateur de la simulation peut interagir dans ce contexte ou dans un autre en utilisant la fenêtre de commande

- **Les variables**

Emplacement pour stocker des valeurs, telles que des nombres ou du texte. Il existe trois types de variables :

- **Variables globales** : une variable globale est accessible par tous les agents. Il doit être déclaré avant toutes les procédures. Exemple :

```
globals [  
    max_energy    ;; énergie maximale  
    total_sim_time ;; le temps de simulation total  
]
```

- **Variables d'agent** : chaque tortue et chaque patch a son propre ensemble de variables, nommé variables d'agent. La valeur d'une variable d'agent peut varier d'un agent à l'autre.

Exemple :

```
Turtles-own [  
    Name  
]
```

- **Variables locales** : Une variable est définie et accessible uniquement à l'intérieur d'une procédure. Exemple :

```
To permuter [ val1 val2 ]  
    let tmp val1 ;; déclaration de la variable local 'tmp' qui reçoit la valeur 'val1'  
    set val1 val2  
    set val2 tmp  
end
```

- **Variables d'agent prédéfinies** : mise à part les variables que les utilisateurs peuvent déclarer selon leurs besoin, les agents possèdent des variables prédéfinies :

Pour les tortues :

color ;; couleur	size ;; taille
heading ;; orientation en degrés	label ;; nom
xcor ;; coordonnée le long de l'axe x	shape ;; forme
ycor ;; coordonnée le long de l'axe y	who ;; identifiant

Pour les Patches :

pcolor ;; couleur	pxcor ;; l'axe x
plabel ;; nom	pycor ;; l'axe y

- **Les procédures et les fonctions**

- **Les procédures** (Commands) : une procédure est un bloc d'instructions qui définit l'action à mener par les agents et elle ne retourne pas de valeur.

Déclaration :

```
to <nom_procédure> [<parametre1> <parametre2> . . . ]  
  <Instructions>  
End
```

Appel : <nom_procédure> <argument1> <argument2>

- **Les fonctions** (Reporters) : Ensemble d'instructions pour calculer et retourner une valeur. La valeur est indiquée avec le mot-clé "rapport" (équivalent à "return" dans d'autres langages).

Déclaration :

```
To-report <nom_fonction> [<parametre1> <parametre2> . . . ]  
  <Instructions>  
  Report <valeur>  
End
```

Appel : <nom_fonction> <argument1> <argument2>

- **Asks**

Demander à un ou plusieurs agents (tortues, patches, liens) d'exécuter des commandes.

Syntaxe : ask turtles [<commands>] ;; demander à toutes les tortues

 ask turtle <id> [<commands>] ;; demander à une tortue

Exemple : ask turtles [set color red] ;; demander à toutes les tortues à devenir rouge

- **Agent set**

Définie un sous ensemble d'agents.

Syntaxe : `<agentset> with [<condition>]` ;; sous ensemble d'agents selon la condition
`<agentset> in-radius <distance>` ;; sous-ensemble d'agents situés à une distance de l'appelant

Exemple 1 : `turtles with [color red]` ;; créer un sous ensemble de tortues en rouge

Exemple 2 : `turtles in radius 3` ;; sélectionner les tortues à moins de 3 pas de l'appelant

- **Breeds**

Breed ou race, un ensemble d'agents dont le type ou l'espèce est autre que les tortues.

Déclaration des Breeds :

Syntaxe : `breed [<breeds> <breed>]`

La première valeur est le pluriel de la race. La deuxième valeur est le singulier de la race.

Exemple : `breed [chats chat]` ;; déclaration d'agent de type ou race chats

Remarque : Breed (race) est une variable de la tortue. Exemple :

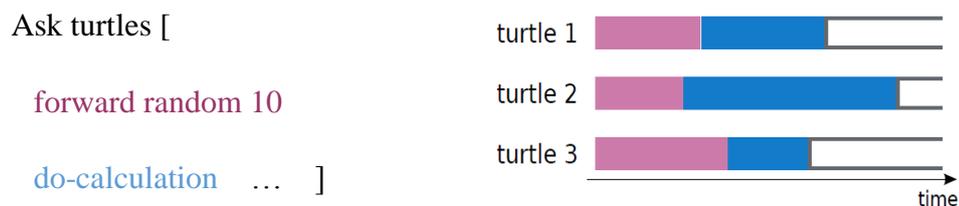
Demander à la tortue numéro 5 de permuter vers la race chien si elle est de race chat :

`ask turtle 5 [if (breed = chat) [set breed = chien]]`

- **Exécution parallèle des agents**

Les agents s'exécutent en parallèle (chaque agent est un thread indépendant).

Exemple : Exécution des commandes en parallèle par toutes les tortues.



- **Synchronisation des agents**

Les threads d'agent attendent et se rejoignent à la fin d'un bloc d'exécution.

Exemple : Toutes les tortues attendent que les premières commandes soient exécutées par toutes avant d'exécuter les secondes commandes :

Ask turtles [forward random 10]

Ask turtles [do-calculation] [43].

5.3.2 L'interface graphique de Netlogo

L'interface graphique de Netlogo comporte trois onglets :

Interface : visionneuse et outils d'interface utilisateur pour la simulation.

Info : informations textuelles sur le modèle simulé.

Code : code source NetLogo du modèle simulé.

La figure ci-dessous montre les différents composants de l'interface Netlogo :

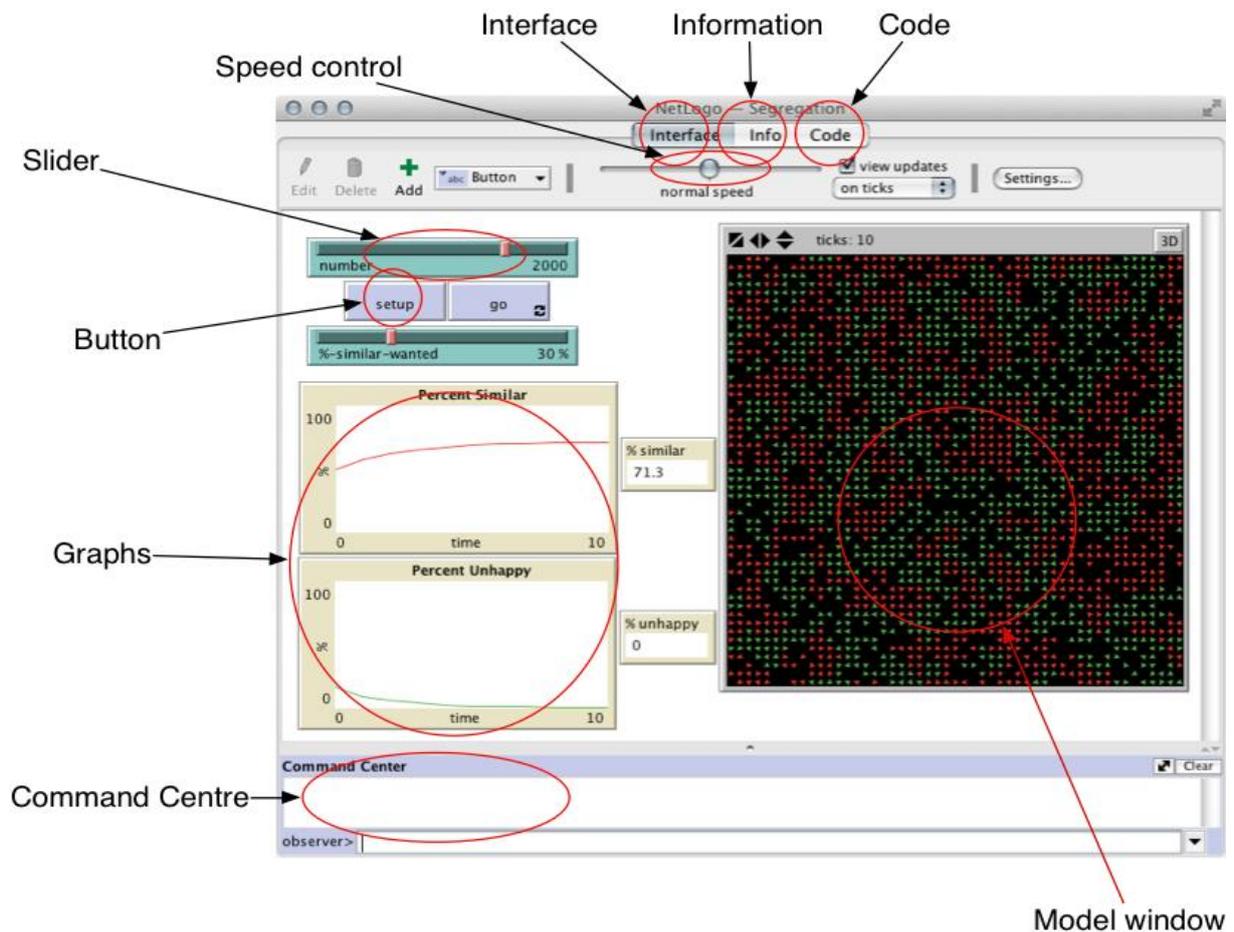


Figure 5.3 Composants de la fenetre Netlogo [44]

Explications : la définition de chaque composant de la fenêtre ci-dessus est comme suit :

- **Button** : Bouton permettant d'exécuter des procédures.
- **Slider** : Entrée qui permet à l'utilisateur de modifier les valeurs des variables.
- **Speed contrôle** : régler la vitesse d'exécution.
- **Interface** : l'onglet interface affiche la présente interface.
- **Info** : l'onglet info contient une interface contenant des informations fournis par l'utilisateur à propos du modèle simulé.
- **Code** : l'onglet code affiche une interface contenant une zone où l'utilisateur peut écrire le code en langage Netlogo.
- **Graphs** : consiste en une Présentation graphique des données résultantes de la simulation.
- **Command Center** : Permet à l'utilisateur de taper des commandes à exécuter par les agents.
- **Model window** : Un panneau qui affiche le déroulement de la simulation.

5.4 La mise en œuvre du modèle

Dans ce modèle, l'ensemble d'agents de tortues peut agir soit comme des nœuds de réseau, soit comme des événements d'un réseau sans fil, les liens (entre tortues) modélisent la connectivité radio entre les nœuds du réseau, toute paire de nœuds connectés par le même lien peut communiquer entre eux. Cependant, tous les nœuds partagent la même configuration radio. Le modèle considère que les liens sont symétriques et leur création dépend du rayon de transmission entré en paramètres.

5.4.1 Création et initialisation de l'environnement

5.4.1.1 Code

Ci-dessous, des parties de code Netlogo des procédures utilisées dans modèle :

- Initialisation de l'environnement : Création d'agents, initialisation des variables globales et les variables propre à chaque type d'agent,
- Mobilité des nœuds, portée radio, envoi d'ondes, génération de messages, contrôle d'accès CSMA/CA

Remarque : chaque instruction est expliquée par commentaire ;; .

1) Ce code permet d'initialiser la simulation et de créer les nœuds et leurs portées radio

```

physique & csmaCa - NetLogo (C:\Users\Hpc\Desktop\COURS M2\MEMOIRE FIN D'ETUDE)
File Edit Tools Zoom Tabs Help
Interface Info Code
Find... Check | Procedures |  Indent automatically

globals [num_noeud_source num_noeud_destination radius relaynode envoyé is totalrxfailed totaltxfailed totaltx totalrx r ]
breed [noeuds noeud]
breed [zones zone]
zones-own [location state]
noeuds-own [ state acklist deffercount nattempts CW txbuffer destination mytxlist packet_errors txfailed rxfailed txcount rxcount myheaderlist datarxcount]

; l'initialisation de la simulation, creation des noeuds
to setup
  __clear-all-and-reset-ticks
  create-noeuds nombre_noeuds ;création des noeuds
  [
    setxy random-xcor * 0.95 random-ycor * 0.95 ; positionnement des noeuds d'une maniere aléatoire
    set shape "circle" ;prendre la forme d'un cercle
    set color green ; prendre la couleur vert
    set label who ;afficher leurs numero d'identifiant
  ]

  ask noeuds [
    set radius (rayon_transmission * 7) / 20
    create-links-with other noeuds in-radius radius [set color blue] ;creation des liens sans fils avec les voisins qui sont dans la portée radio (le rayon de transmission)

    ;; initialisation des attribut des noeuds]
    ;initialiser tous les noeuds en état de reception
    set state "rx"
    ;le buffer des message à envoyer
    set txbuffer 0
    ;la liste des noeuds d'une transmission
    set mytxlist []
    ;initialiser la fenetre de contention avec la valeur donnée en parametre
    set CW Cwmin
    ;initialiser la liste des acquittements
    set acklist []
    ;liste des element de l'entete d'un paquet
    set myheaderlist []
  ]

  set myheaderlist []
  ;initialiser l'identifiant du destinataire à -1 (aucune génération de msgs pour l'instant)
  set destination -1
  ;nombre de tentatives
  set nattempts -1
  ;nombre de paquet erronés
  set packet_errors 0
  ;nombre de messages recus
  set datarxcount 0
]

setup1 ;la fonction setup1 pour initialiser les portées radio
if (visualise_portée = false) [ ask zones[set hidden? true] ; si le bouton visualise_ portée est "off" alors les agents zone seront invisibles
]
end

; cette procedure permet de créer des agents 'zone' qui prendront en charge la representation de la portée radio et de l'affecter à chaque noeud
to setup1
  create-zones nombre_noeuds ; créer autant de zones (portée) que de noeuds
  set-default-shape zones "portee"

  let i nombre_noeuds
  while [i < nombre_noeuds * 2][
    ask zone i [ set size rayon_transmission ;la portée aura la taille du rayon de transmission entré en parametre
      set location noeud (i - nombre_noeuds) ; chaque agent 'zone' se positionnera à la meme position d'un noeud
      move-to location]
    ask noeud (i - nombre_noeuds) [ create-link-with zone i [ tie set color white ] ]
    set i i + 1
  ]
end

```

Figure 5.4 Code d'initialisation de la simulation

Les tortues utilisées

1) Les agents *noeuds* : sont les nœuds mobiles constituant un réseau MANET

- Les variables de chaque nœud : *noeuds-own*

Etat : l'état de chaque nœud : TX ou RX.

Acklist : une liste d'acquittements.

Deffercount : le compteur Backoff.

Nattempts : nbr tentatives d'envoi.

packet_errors : nbr paquets erronés.

txbuffer : nbr de paquets générés.

Mytxlist : liste de nœuds destination.

Destination : le nœud destination.

CW : fenêtre de contention.

txfailed : nombre transmission échoués.

Rxfailed : nbr réception échouées

Txcount : nbr de transmissions effectuées.

Datarxcount : nbr de paquets reçus (destination final).

rxcount : nbr de paquets reçus (données relayées).

myheaderlist : tableau à deux dimension des entêtes de paquets générés.

2) les agents *zone* : **représentent la portée radio de chaque noeud**

- Les variables de chaque nœud : *zones-own*

Location : chaque zone sera positionnée à la même position d'un nœud.

Les variables globales : '*globals*', ces variables concernent l'ensemble de la simulation, ils peuvent être utilisés par d'autres agents :

num_noeud_source : numéro (identifiant) d'un nœud générateur d'un message.

num_noeud_destination : numéro (identifiant) d'un nœud destinataire d'un message.

radius : la portée radio des nœuds calculée à partir du rayon de transmission.

relaynode : numéro (identifiant) d'un nœud relay (intermédiaire).

envoyé : booléen qui prend la valeur true si paquet est livré avec succès, sinon faux.

is : une liste des nœud qui se trouve à la portée d'un nœud.

totalrxfailed : nombre totale de réceptions de paquets échouées dans la simulation.

totaltxfailed : nombre totale de transmissions de paquets échoués dans la simulation.

totaltx : nombre totale de transmissions de paquet réussis dans la simulation.

totalrx : nombre totale de réceptions de paquet réussis dans la simulation.

R : nombre de nœuds relais (intermédiaires) qu'une transmission d'un paquet a eu besoin pour aller d'un nœud source vers un le nœud destination.

2) Cette partie de code permet de réaliser les mobilités des nœuds

```
; les noeuds se déplacent aleatoirement ce qui genere un changement de la topologie et liens entre les noeuds
; la mobilité des noeuds
to mobilité

ask n-of random nombre_noeuds noeuds
[
  ; leur donner une orientation aléatoire
  set heading random 360

  ;; les disperser
  fd random 8 set xcor int(xcor) set ycor int(ycor)

  ;let ticks_per_sec (1 / ( 0.01 + (message_size_bytes * 8) / (bit_rate_kbps * 1000)))
  ; forward (100 / ticks_per_sec)

  ; regenerer les lien avec les noeuds qui sont dans la portée
  ask my-links with [color = blue] [ die ]

  create-links-with other noeuds in-radius radius [set color blue]
]
end
```

Figure 5.5 Code de mobilité

3) Cette partie de code permet de simuler l'envoi d'ondes radio

```
; pour simuler une onde radio
to ondes
  ;;demander à l'agent noeud emetteur de trouver l'agent zone (portée) à qui il est lié
  ;;une fois la zone corespondante est trouvée, on lui demande de prendre la taille 0
  ;;puis d'augmenter sa jusqu'a atteindre la limite du rayon de Transmission

  ask noeud num_noeud_source[

    let z (min-one-of (other zones with [ link-neighbor? myself ])
      [distance myself])

    ask z [
      set hidden? false
      set size 0

      let i 0
      let j 1
      while [i < rayon_transmission][set size j
        set i i + 1
        set j j + 1
      ]

      if( visualise_portée = false)[ set hidden? true]
    ]
  ]
end
```

Figure 5.6 Code d'ondes radio

4) Cette partie de code permet de générer des messages multiples pour CSMA/CA

```
;; une procedure qui s'exécute en boucle
to generer_messages
  ;; un noeud est choisi aléatoirement pour generer un message, qui à son tour choisi un noeud de destination aléatoirement
  ask noeuds [set label who]
  ask one-of noeuds [

    ;;choisir une destination de message
    let ds [who] of one-of other noeuds
    print sentence who ds ;; print permet de nous afficher en commentaire

    ;;faire appel à la procédure handle_message pour générer l'entete et le mettre dans le buffer
    handle_message 1 1 who ds 1 who
  ]
; csmaCa_dcf
end
```

Figure 5.7 Code génération messages multiples pour CSMA/CA

5) Cette partie de code concerne la procédure qui prend en charge le CSMA/CA

```

to csmaCa_dcf

ask noeuds with [state != "dead" and state != "sleep" ] [

set state "rx" ;; tout les noeud sont initialisés en état reception
set acklist [] ;; toujours initialiser le 'acklist'

;Règles BACKOFF de 802.11
; Lorsqu'un message a été livré avec succès ou que toutes les tentatives de retransmission ont été
; épuisés, et la station a un message suivant à transmettre, alors cette station doit effectuer une procédure de Backoff

;deffercount c'est le backoff tiré
;nattempts c'est le nombre de tentative

;;tirer un backoff pour une nouvelle transmission après epuisement de nmbr de tentatives d'envoi d'un message
if ([deffercount] of self = 0) and (nattempts = -1)
[
set deffercount random (CW) ;; initialiser le backof of 802.11
]

;; deffercount c'est le compteur de backoff
if ([deffercount] of self = 0)[
if (txbuffer > 0) ;;si le buffer n'est pas vide
[

print sentence who " je devien en etat lst : prêt a transmettre" ;; le backoff = 0 et le buffer n'est pas vide
set state "lst"

;mytxlist : la liste des noeuds pour qui une source veut envoyer un msg
set destination first mytxlist ;;ajouter la destination du message à la liste mytxlist
print sentence "la destination est le noeud : " destination
]
]

; si le buffer denvoi est vide, alors reste en etat RX et sera colorée en gris
if ([state] of self = "rx")[set color gray]
]

]

;; #### BACKOFF procedure ####
ask noeuds with [state != "dead" and state != "sleep" ] [

ask self [

;le nombre de noeud voisin (dans la portée), -1 c'est pour enlver le lien avec sa zone 'portée'
ifelse ([count link-neighbors - 1] of self >= 1)[
;; essayer la transmission seulement si le backoff est fini
ifelse ([deffercount] of self <= 0)
[
;si il est en état de vouloir transmettre
if ([state] of self = "lst") [

;; si un autre noeud qu est dans la portée est aussi dans un état de transmission alors le canal est occupé
ifelse (any? link-neighbors with [state = "tx"]) ;; le canal est occupé - probleme de noeud Exposé
[
set deffercount random (CW + 1) ;; on tire un autre backoff en augmentant la fenetre de 1
set state "rx" ;le noeud redevien en état de rx pour laisser le canal libre
set color white ;; transmission retardée par le beckoff
]
]
; sinon
[
show (sentence "je suis le noeud : " who " mon backoff est fini, je commence la transmission")
print ""
;; demarer la transmission
set state "tx"
set color red
]
]
]

;le backoff n'est pas encore fini, si le canal est libre on decremente le backoff
if (count link-neighbors with [state = "tx"] <= 0) ;; le canal est libre, donc on decremente le backoff
[
set deffercount (deffercount - 1) ;; decrementer jusqu'a 0
]
]

[
if ([deffercount] of self > 0) [set deffercount (deffercount - 1) ] ]]] ;; decrementer jusqu'a 0 ]

```



```

;##### ACKs confirmation procedure #####
ask noeuds [
;print "je suis dans aquittement"
  ask self[
    if ([state] of self = "tx") [

      let dest [destination] of self

      ;;pour le broadcast aucun aquitement n'est envoyé
      ifelse (dest != "broadcast") [

        ;compter le nombre de noeud qui sont les destinataires de ce message unicast ou multicast
        ifelse (count link-neighbors with [who = dest] >= 1) [

          ;prendre le numero de premier noeud qui dans la liste des aquitement du distinaire
          let nodex -1

          if(not empty? acklist)[
            set nodex [first acklist] of one-of link-neighbors with [who = dest] ]

          ;si ce numero (nodex) est le meme avec le who de cet emetteur alors, c'est reussi
          ifelse (nodex = who)
          [
            set color blue ;; réussite de la transmission confirmée par la réception de l'Ack

            set txbuffer (txbuffer - 1)
            set txcount (txcount + 1)

            set totaltx (totaltx + 1)

            set mytxlist remove-item 0 mytxlist
            set myheaderlist remove-item 0 myheaderlist
            set natempts -1
            set CW CWmin
            set destination -1

            show (sentence " aquittement reçu par le noeud : " who " avec succes")
            print ""
          ]

        ]

        ;sinon, Si le STA source ne reçoit pas de msg d'accusé de réception, il tentera de retransmettre le msg
        [

          ;augmenter le compteur des tentatives d'envoi
          set natempts (natempts + 1)

          ; on double la fenetre de contention
          set CW (CW * 2)

          ;si le cw depasse le Cwmax definie en parametre, alors la CW prend Cwmax comme valeur
          if (CW >= Cwmax) [set CW Cwmax]

          ;le noeud tire un backoff aléatoire de CW
          set deffercount random (CW + 1)

          ; si nombre de tentative à dépassé le nombre de tentatives maximales definie en parametre
          ;alors le message sera deteruit
          if (natempts >= nombre_tentative)
          [
            ;;on mis à jour les compteurs et on retire l'entete de message de la liste des entetes générés

            set txbuffer (txbuffer - 1)
            set txcount (txcount + 1)
            set mytxlist remove-item 0 mytxlist
            set myheaderlist remove-item 0 myheaderlist
            set natempts -1
            set CW CWmin
            set destination -1
            set txfailed (txfailed + 1)
            set totaltxfailed (totaltxfailed + 1)

            set color pink ;; paquet deteruit par le noeud
          ]
        ]
      ]
    ]
  ]
]

[ ;; dans ce cas, la destination du paquet (dans le buffer) n'est pas active donc le paquet est deteruit
;NOUS AVONS pas encore programmer l'etat sleep et dead qui lié à la consommation d'énergie dans le temps

set txbuffer (txbuffer - 1)
set txcount (txcount + 1)
;;set totaltx (totaltx + 1)

```



```
;; gerer le message quand il est générer pour la 1ere fois ou quand il est reçu pour être retransmis au prochain saut
;; cette procedure est appelée par deux procedure : to generer_message et par to receive_message avec les atributs de l'entete
to handle_message [typeofmsg acdelay sourceadd msgdest hopcount msgrelaysrc]

  show ("je suis dans handle")
  ;; Types de messages: 0: données relayées, 1:nouvelles données, 2: msg de contrôle relayé, 3: nouveau message de contrôle
  let distancetodest 0
  ;calculer la distance vers la destination final]
  set distancetodest distance noeud msgdest
  show ( sentence "la distance jusqu'au destinataire final est " distancetodest)

  ;; si la distance >0
  if (distancetodest > 0)[
    let rd [] ;mettre tous les noeuds qui se trouve dans la portée radio (rayon de transmission) de cet emetteur dans une liste
    set rd noeuds in-radius radius

    ;; si la destination de ce message fait parti de la liste des voisins c-a-d se trouve dans portée alors l'entete sera créer et
    ;; le msg sera envoyer après avoir executer la procedure de csmaCA_dcf
    ifelse (member? noeud msgdest rd)[
      print "je rempli la liste myheaderliste avec le noeud destination finale"
      print ""
      set txbuffer (txbuffer + 1)
      ;ajouter le prochain saut (noeud) dans la liste des noeud à qui on va transmis
      set mytxlist lput (msgdest) mytxlist
      ;; création de l'entete et l'ajouter à la liste des entetes
      set myheaderlist lput (sentence who msgdest acdelay sourceadd typeofmsg hopcount) myheaderlist
    ]

    ;;sinon, si la destination ne se trouve pas dans la portée radio de l'emetteur, trouver parmi les voisins celui
    ;; qui possede la plus courte distance vers la destination final pour l'utiliser comme un noeud relay

    ;;sinon, si la destination ne se trouve pas dans la portée radio de l'emetteur, trouver parmi les voisins celui
    ;; qui possede la plus courte distance vers la destination final pour l'utiliser comme un noeud relay
    let relay min-one-of other noeuds in-radius radius [distance noeud msgdest]

    ifelse (relay != nobody)[ ;; il existe un noeud pour devenir un relay alors

      ;;le noeud relay devient le destinataire intermediaire de ce message
      set destination [who] of relay
      ;; augementer le compteur de nombre de transmission du noeud
      set txbuffer (txbuffer + 1)
      ;ajouter le prochain saut (destination) dans la liste des noeud à qui on a transmis
      set mytxlist lput (destination) mytxlist

      show "je rempli la liste myheaderliste avec le noeud relay"
      show (sentence "le prochain noeud relay est le : " relay)
      print ""
      ; construire le l'entete de message
      set myheaderlist lput (sentence who msgdest acdelay sourceadd typeofmsg hopcount) myheaderlist
    ]

    ;aucun noeud voisin se trouve dans la portée radio
    [show (sentence msgdest ": aucun chemin n'est trouvé.")]
  ]
]
end
```

Figure 5.8 Code CSMA/CA et Routage

5.4.1.2 Interface

L'interface de la simulation comporte un ensemble de boutons, des inputs, des sliders, des switches et le panneau affichant le déroulement de la simulation (actions et déplacements des agents dans leur environnement).

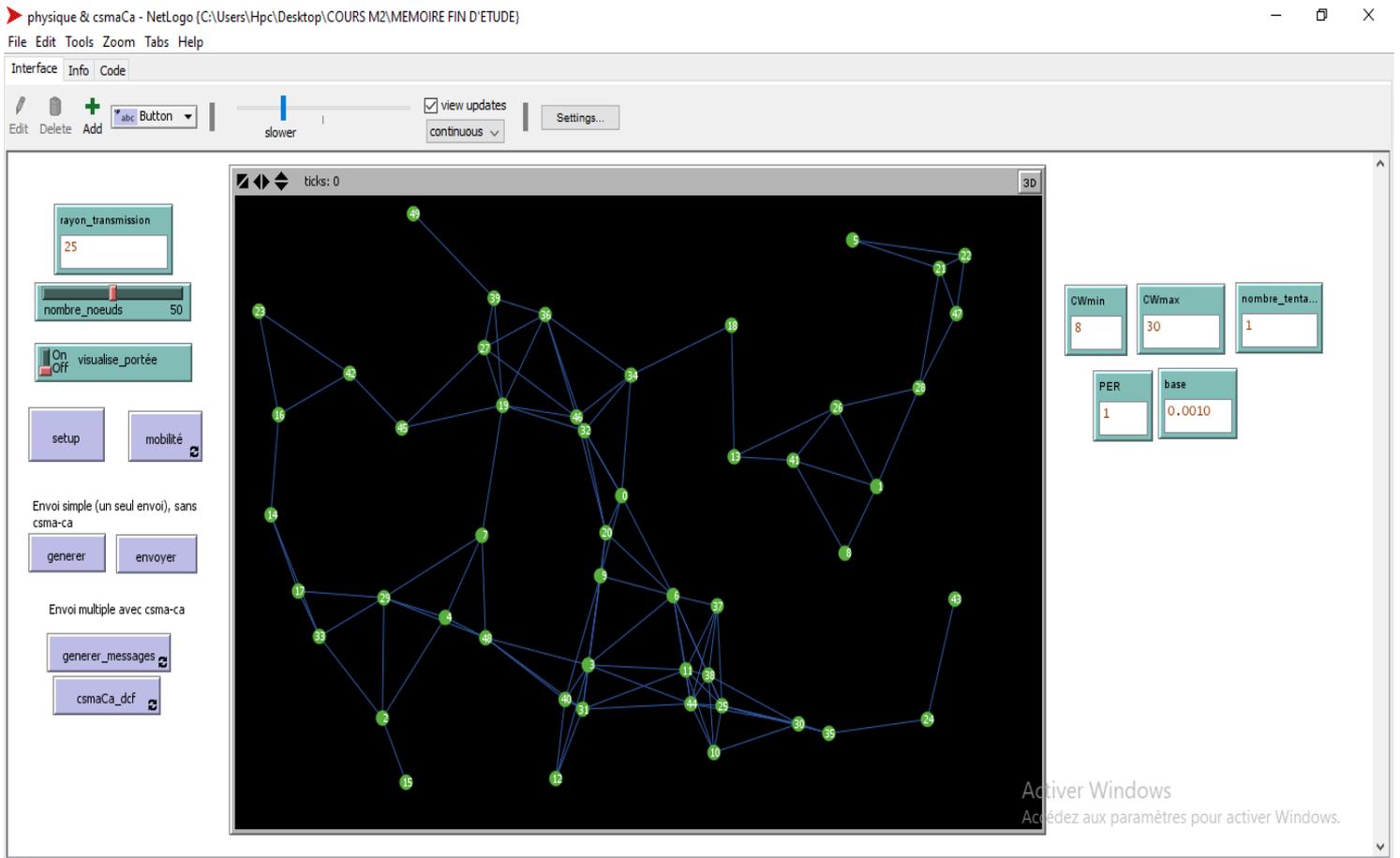


Figure 5.9 Interface de la simulation

1- Les paramètres d'entrés :

- **Nombre_noeuds** : un slider qui permet à l'utilisateur d'entrer le nombre de nœuds de ce réseaux.
- **Rayon_transmission** : l'utilisateur indique la valeur du rayon de transmission, le cercle de la portée prendra comme taille cette valeur, et des liens seront créés avec les nœuds voisin qui se trouve à l'intérieur de cette étendu.
- **Visualise_portée** : activer ou désactiver la visualisation du cercle de la portée des nœuds, dans ce cas de figure, il est désactivé. Dans le cas contraire, la topologie sera comme suit :

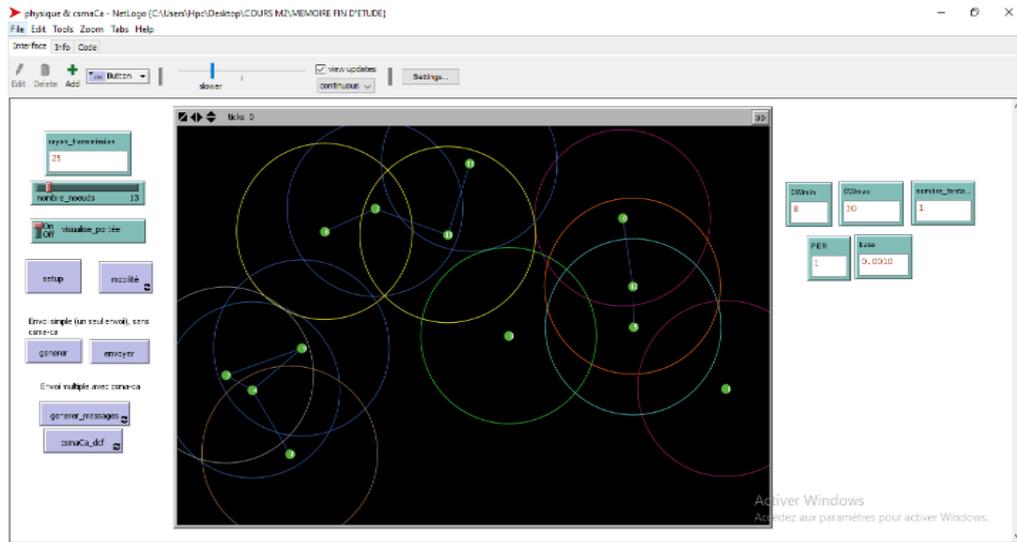


Figure 5.10 visualisation de la portée radio

- **CWmin** et **CWmax** : l'utilisateur entre la valeur de l'intervalle de la fenêtre de contention.
- **Nombre_tentative** : l'utilisateur indique la valeur de nombre de tentative de retransmission d'un paquet avant qu'il soit détruit.
- **PER**: le taux d'erreur de paquet (PER) est le nombre de paquets de données incorrectement reçus divisé par le nombre total de paquets reçus. ce taux inclue BER (bits error rate), bruit thermique, Multipath et Fading.

Pour calculer la probabilité d'erreur, on utilise les entrées **Base** et **PER** pour indiquer le taux de probabilité d'erreur. Par exemple, pour avoir un taux = 0,035, il faut définir le PER sur 3,5 et la base à $1.0E-2 = 0.01$.

2- Les boutons

Chaque bouton, une procédure écrite dans la partie code lui est affecté. L'exécution de ces procédures désigne les actions des agents.

- **Setup** : le bouton qui exécute la procédure To Setup, et qui permet de créer les agents, initialiser les variable, créer les liaisons entre les nœuds voisins,
- **Mobilité** : exécuté en boucle : les nœuds se déplace dans l'environnement d'une manière aléatoire, par conséquent, la topologie du réseau est en changement continue, les liens disparaissent, et d'autres se créent, selon la portée radio de chaque nœud.
- **Generer** et **envoyer** : concernent un envoie de message simple, c-a-d : un seul nœud génère un message et le transmet au nœud destinataire, soit à un saut unique, soit à sauts multiples (en exécutant l'algorithme de routage).

- **Generer_message** et **csmaCa_dcf** : dans ce cas de simulation, les deux boutons s'exécute en boucle. Le premier permet à chaque agent nœud de générer des paquets de données, ces derniers sont insérés dans le buffer des transmissions. Les agents nœuds envoient leurs paquets en exécutant l'algorithme CSMA/CA.

5.4.2 Envoie de message simple

Cette partie décrit la procédure d'envoi d'un message simple généré par un seul agent nœud à destination d'un autre agent nœud.

1- à un saut :

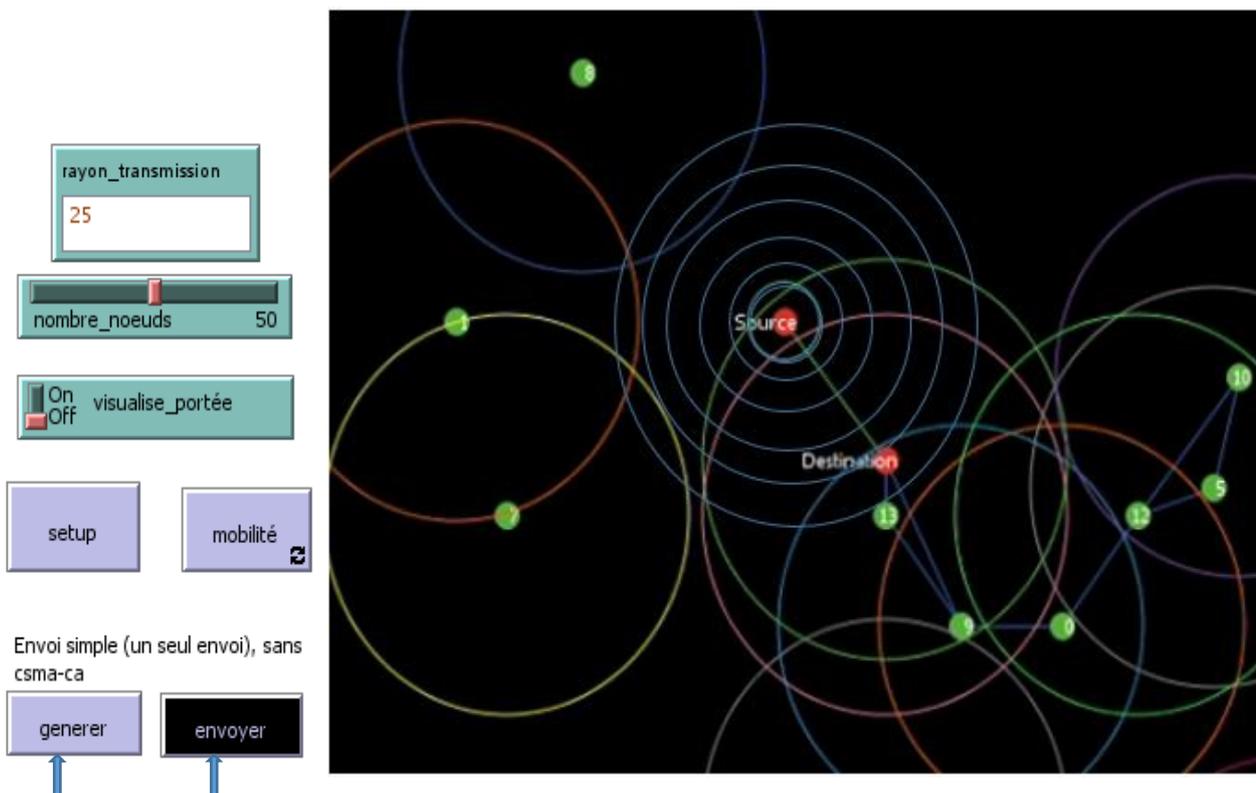


Figure 5.11 Envoi simple à un saut

Dans ce cas de figure, le nœud source génère un message et le transmet au nœud de destination qui se trouve dans portée radio.

2- à sauts multiples (routage) :

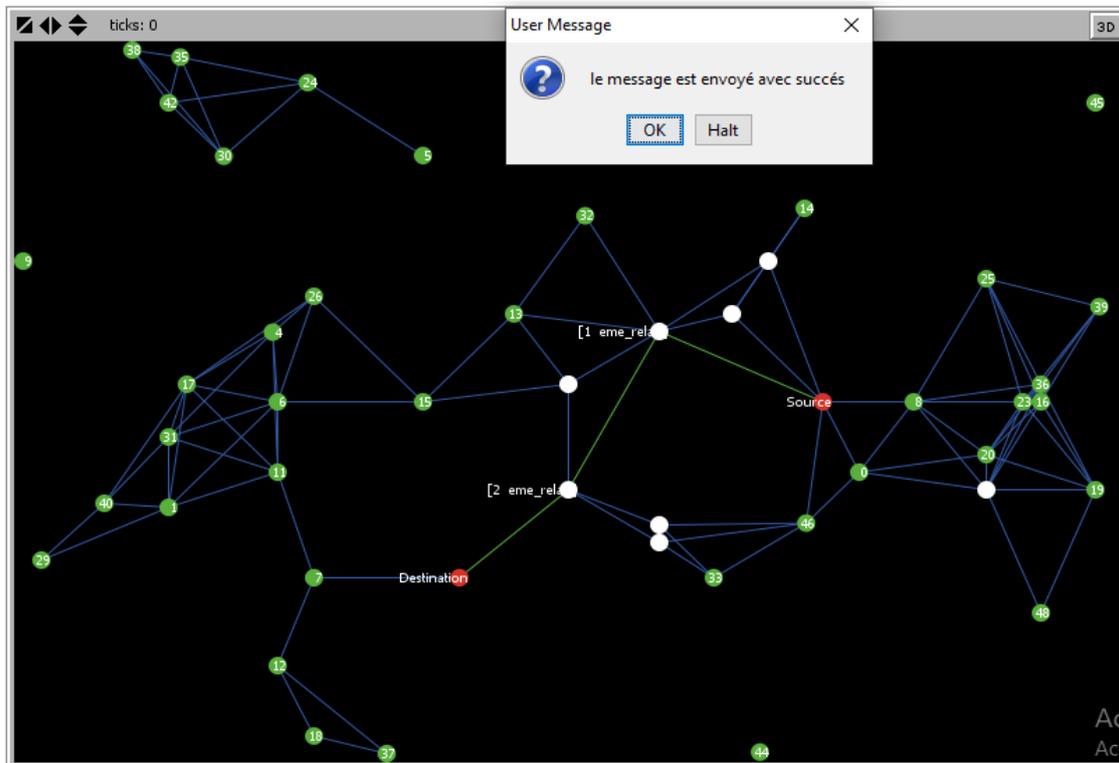
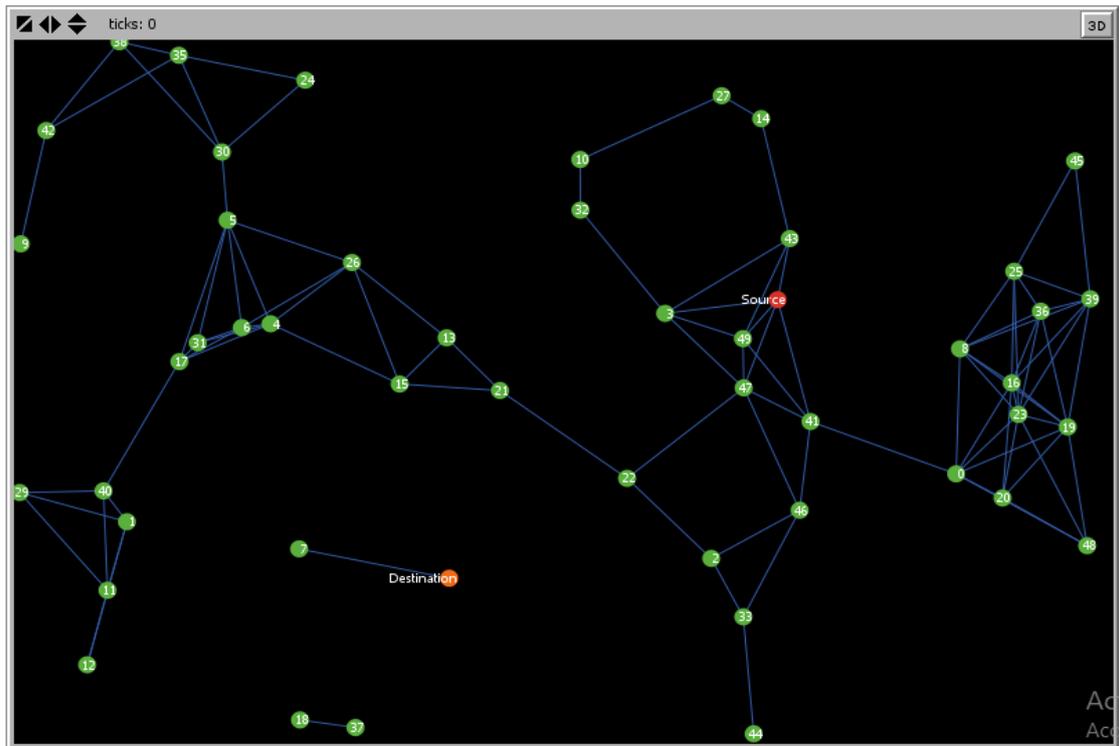


Figure 5.12 Envoi simple à sauts multiple (routage)

- : Les nœuds source et destination
- : Les nœuds ayant reçus le message et l'ont détruits
- [ieme relay] : les nœuds intermédiaires qui ont acheminés le message.
- : Le chemin que le message à suivi pour arriver à destination.

Dans ce cas de figure, les agents nœuds sont en mobilité, la topologie du réseau change, de liaisons entre les nœuds changent aussi.

Le nœud source génère un message pour une destination qui se trouve hors de sa portée radio. Il transmet le message au nœud voisin le plus proche de la destination pour devenir un nœud Relay et se charger à son tour d'acheminer le message jusqu'au bout. Le nœud Relay vérifie si la destination est dans sa portée alors il lui transmet directement le message, sinon il exécute à son tour la procédure de routage, et ainsi de suite jusqu'à atteindre le nœud final.

5.4.3 Envoi de messages multiples avec CSMA/CA

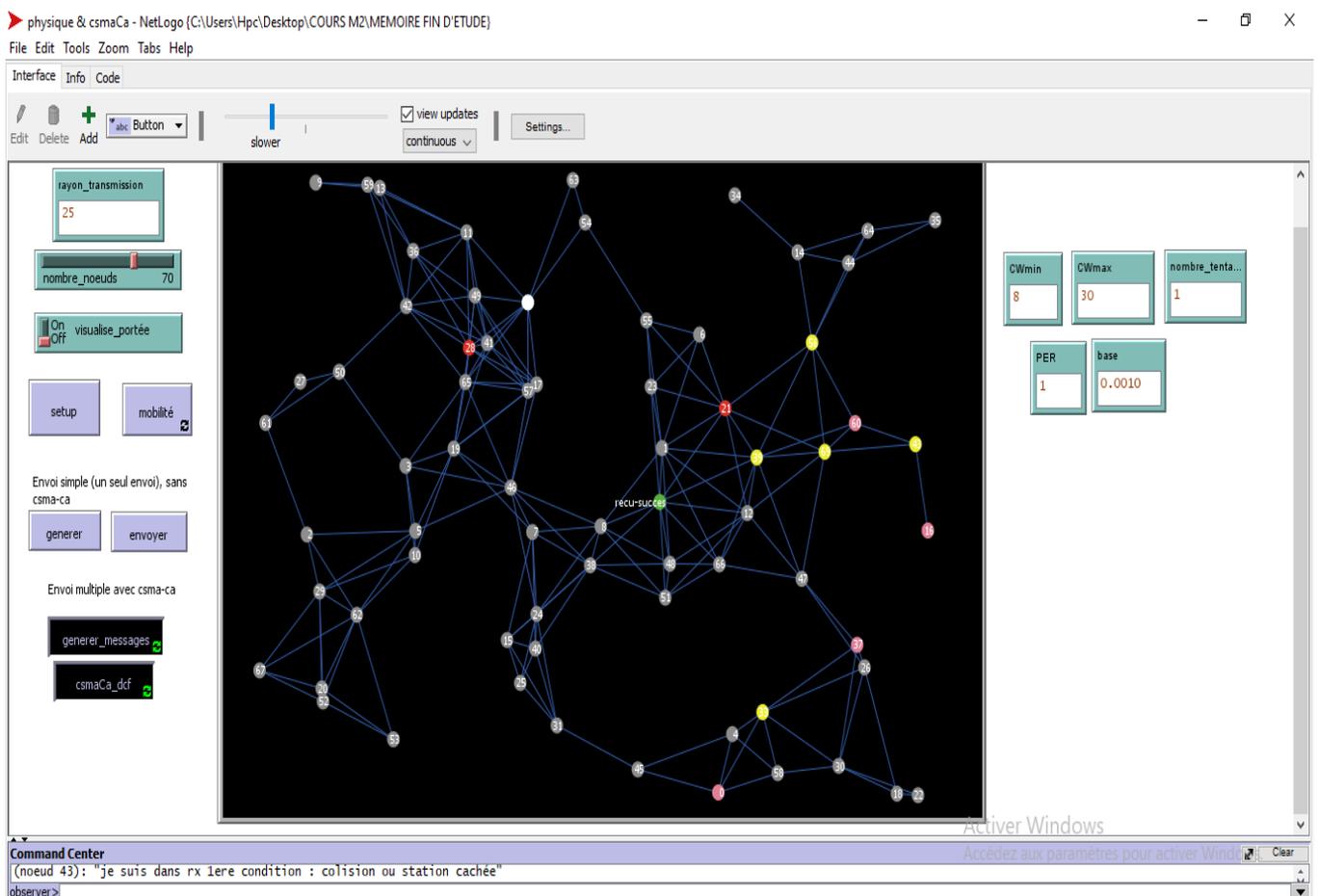


Figure 5.13 Envoi de messages avec CSMA-CA

-  : Génération de message par un nœud.
-  : Réception d'un message par un nœud destinataire.
-  : Collision produite au niveau d'un nœud récepteur.
-  : Un nœud qui détruit un message à cause de l'épuisement du nbr_tentatives.

Déroulement :

En appuyant sur le bouton « **générer_messages** », les nœuds génèrent des messages et créent des entêtes propres aux messages (type de message, identifiant de la source, identifiant de la destination, nombre de sauts, identifiant de nœud relay).

Ensuite, en appuyant sur le bouton « **csmaCa_dcf** », le protocole CSMA-CA se met à s'exécuter afin gérer les transmissions des messages dans les IBSS. Ce dernier s'exécute en coopération avec l'approche de routage définie auparavant.

A l'instant de la capture on :

Les nœuds  (soit des nœuds source ou des nœuds relays) ont finis de décrémenter le backoff, ils se mettent en état de transmission et transmettent leurs messages au prochain saut. Si le message est reçu par le destinataire final  avec succès, alors un acquittement lui sera retourné afin de confirmer l'envoi et de mettre à jour les buffers et les compteurs.

Dans le cas contraire, c-a-d : échec d'envoi, le nœud  change de couleur et devient  qui est de destruction du message à cause d'absence d'acquiescement après avoir retransmis le message autant de fois que **nombre_tentative**.

Dans le cas où un nœud reçoit plusieurs messages à la fois, soit à cause de la station cachée, soit deux nœud et plus qui ont transmis en même temps (tirage de même backoff), ce qui est synonyme de collision, alors ce nœud devient  , et aucun acquittement ne sera délivré à l'émetteur du message.

En outre, Les sortie de la simulation sont sous forme de commentaires qui s'afficheront au niveau de **command center** au bas de la fenêtre, et qui décrivent les actions effectuées par chaque agent et les résultats de ces actions.

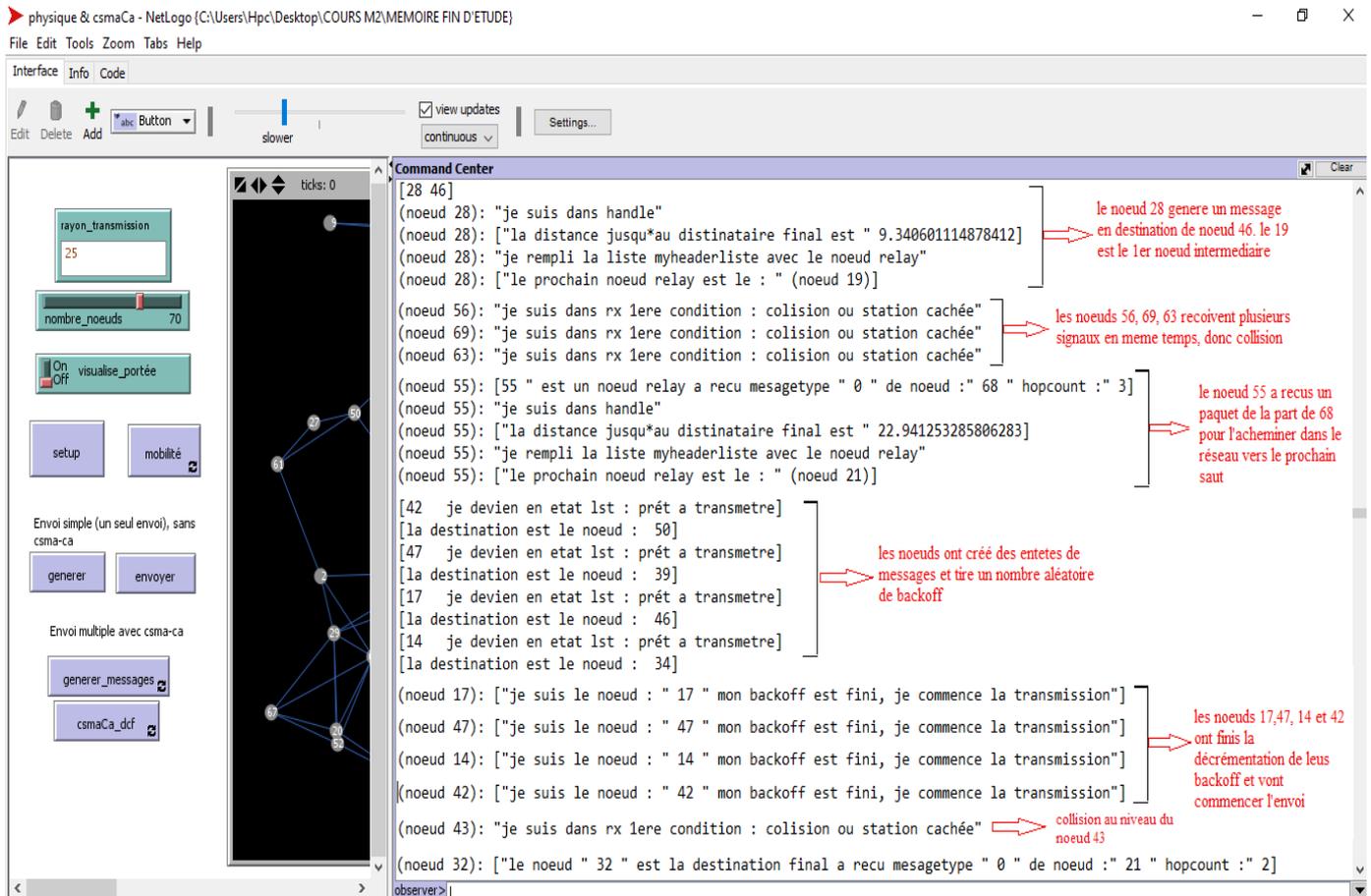


Figure 5.14 Commentaires de sortie détaillant la simulation

5.5 Conclusion

Dans ce chapitre nous avons expliqué la démarche que nous avons suivie pour implémenter et mettre en œuvre notre modèle de réseau MANET sur le simulateur multi Agent NETLOGO. Nous avons expliqué l'objectif du modèle en détaillant chaque paramètre et protocole implémenté. Et enfin, nous avons déroulé le modèle en expliquant son fonctionnement et les évènements qui se produisent à chaque étape de la simulation.

Conclusion et Perspectives

Les réseaux à grande échelle sont difficiles à simuler dans les simulateurs de réseau traditionnels, car ils peuvent nécessiter des ressources de calcul élevées. Dans un besoin d'outils alternatifs flexibles équipés de techniques efficaces pour modéliser les réseaux modernes à mesure qu'ils évoluent vers des systèmes complexes tel que les réseaux Ad hoc, nous avons exploré le domaine des systèmes multi agents utilisés dans de tels environnements à complexité élevée afin de répondre aux besoins de modélisation et de simulation proche de la réalité d'un réseau Ad hoc. Nous avons procédé à l'utilisation du simulateur multi agent NETLOGO pour concevoir un modèle d'un réseau MANET. Notre modèle permet de simuler les principaux protocoles et aspects liés aux deux couches PHY et MAC de la norme 802.11 utilisés pour les réseaux Ad hoc (la transmission radio et le mécanisme d'accès au canal CSMA/CA DCF), la mobilité des nœuds, ainsi que le mécanisme de routage à court chemin qui assure l'acheminement des paquets dans le réseau.

En utilisant le simulateur NETLOGO, nous avons constaté une certaine facilité et une simplicité dans son utilisation et sa manipulation par rapport aux simulateurs réseaux traditionnels, notamment la facilité d'apprentissage de son langage de programmation ainsi que la simplicité d'usage de son interface graphique et de surfer entre ses composants. Pour son utilisation, NETLOGO permet de modéliser et de simuler toutes sortes de systèmes complexes évoluant dans le temps par les comportements collectifs produits par les interactions de plusieurs entités (agents) autonomes, dynamiques et auto-organisées.

Ayant remarqué dans notre modèle un taux de collisions élevée et un retard dans la délivrance des messages dû au manque d'une modélisation proche de la réalité du concept de **détection**, qui permet d'arrêter la décrémentation du backoff à l'instant de la détection d'un signal, nous pourrions envisager dans la continuité du travail que nous avons présenté, la perspective suivante :

- Etendre le code source du simulateur NETLOGO et ajouter une extension qui permettrait d'offrir à un agent la capacité de la sensibilité à détecter un événement (signal plus précisément une onde) aux alentours de sa zone.

Bibliographie

- [1] K.Benhamza, « Conception d'un système multi -agents adaptatif pour la résolution de problème », Université B. Mokhtar-annaba, 2016.
- [2] T.Marir, « *Les Systèmes Multi-Agents* », Université de Oum El Bouaghi 2017.
- [3] J. Ferber. « *Les systèmes multi-agents : vers une intelligence collective* », InterEdition, 1995.
- [4] D. David, « *Théorie des SMA* », M1 STIC, 2010.
- [6] S. Abbas, H. Sawamura, « *Innovations in Multi-Agent Systems and Applications – 1* ». *Studies in Computational Intelligence*, 113–147, 2010.
- [5] N. Boukhechem, « *Routage dans les réseaux mobile Ad hoc par approche à base d'agents* », Mémoire Présenté en vue de l'obtention du diplôme de Magister en informatique, Université de Constantine, 2008.
- [7] S.Alam « *Approche multi agents pour contrôler l'inondation dans un réseau de capteurs* », mémoire de fin d'étude pour l'obtention du diplôme d'ingénieur d'état en informatique, Ecole nationale supérieure d'informatique, Oued-Smar, Alger, 2009.
- [8] M-P. gleizes, « *Vers la résolution de problèmes par émergence* », habilitation à diriger des recherches de l'Université Paul Sabatier, 2004.
- [8] C. Bernon, « *Émergence fonctionnelle par auto- organisation dans les systèmes artificiels* », Institut de Recherche en Informatique de Toulouse, 2007.
- [9] S. Chouha, S. Zerouala, « *Adaptation de protocole de routage AODV dans un réseau de robots* », projet de fin d'études pour l'obtention du diplôme d'ingénieur d'état en informatique, université de Batna, 2010.
- [10] J.P. Macker, W. Chao, R. Mittu, & M. Abramson, « *Multi-agent systems in mobile ad hoc networks* ». *Proceedings - IEEE Military Communications Conference MILCOM*, 2005.
- [11] A. Kut, S. Labed, & S. Chikhi, « *Netlogo Agent based tool for Modeling and Simulation of Routing Problem in Ad hoc Networks* », 154–160, 2015.
- [12] H. Labiod, « *Wireless Ad Hoc and Sensor Networks* », 2010.

- [13] S. Kassab, M. Oularbi, « *élaboration d'un protocole de routage efficace en énergie pour les réseaux de capteurs sans fil* », mémoire de fin d'études Pour l'obtention du diplôme d'Ingénieur d'Etat en Informatique, École nationale Supérieure d'Informatique (ESI), 2010.
- [14] M. VAN DER MEERSCHEN, « *Hybridation entre les modes ad-hoc et infrastructure dans les réseaux de type Wi-Fi* », Mémoire de fin d'études présenté en vue de l'obtention du grade d'Ingénieur Civil Informaticien en Sciences Appliquées, Université Libre de Bruxelles, 2006.
- [15] BS. HAGGAR, « *Les protocoles de routage dans ad hoc* », Rapport de stage, Université de Reims, 2007.
- [16] R. HEKMAT, « *Ad-hoc Networks : Fundamental Properties and Network Topologies* », Delft University of Technology, The Netherlands, springer, 2018.
- [17] M. TAHAR ABBES, « *Proposition d'un protocole á économie d'énergie dans un réseau hybride GSM et AD HOC Devant* », mémoire doctorat, Université d'Oran, 2012.
- [18] M. Smith., D. Cook, & B. Smith, « *Dairy Science & Technology* », Second Edition , CRC Taylor & Francis Group, 2001
- [19] K. DRIRA, « *Topologie Dynamique Virtuelle Pour Le Routage Dans Les Réseaux Mobiles Ad hoc* » Rapport de projet de fin d'études, école supérieure des communications de TUNIS, 2005.
- [20] P.MOHAPATRA , S. V. KRISHNAMURTHY, « *AD HOC NETWORKS Technologies and Protocols* », University of California, springer, 2018.
- [21] S. DJAHEL, « *Sécurité des Protocoles de Routage et d'Accès au Médium dans les Réseaux Sans Fil Multi-Sauts* », Université des Sciences et Technologies de Lille, December 2010.
- [22] A.Amamra. « *Techniques d'estimation de la bande passante disponible de réseaux sans fil. Informatique mobile* », Université Blaise Pascal - Clermont-Ferrand II, 2008.
- [23] L. Hogie, « *Mobile Ad Hoc Networks : Modelling, Simulation and Broadcast-based Applications. Modeling and Simulation* », Université du Luxembourg, Faculté des

- Sciences, de la Technologie et de la Communication [FSTC], Université du Havre, 2007.
- [24] Wu, P., & Lee, C, « *MAC Protocol for Ad-Hoc Network* », 621–625, (2006).
- [25] R. KUMAR, « *A Comparative Study of MAC Layer Protocols for Mobile Ad-Hoc Networks* ». Bhagwan Parshuram Institute of Technology Rohini, New Delhi, India. 2014.
- [26] L. GADOUM, S. HAOUARI, « *Modélisation Analytique de la norme IEEE 802.11e mode EDCA Bloc ACK avec les chaines de Markov* », Mémoire de fin d'études Pour l'obtention du diplôme de master en informatique, Université de Bejaia, 2016.
- [27] A.GERON, « *Wifi professionnel : la norme 802.11, le déploiement, la sécurité* ». 3eme édition, DUNOD, 2009.
- [28] Y. Grunenberger. « *Réseaux sans fil de nouvelle génération : architectures spontanées et optimisations inter-couches* ». Réseaux et télécommunications [cs.NI]. Institut National Polytechnique de Grenoble - INPG, 2008.
- [29] L. BOURICHE, L. MAMERI, « *Evaluation des performances d'un mécanisme de fragmentation amélioré avec RTS/CTS dans un environnement bruité* », Mémoire de fin d'études Pour l'obtention du diplôme de master en informatique, Université de Bejaia, 2012.
- [30] R. Flickenger, « *Réseaux sans fil dans les pays en développement* », Deuxième édition, 2009.
- [31] C. Marchand, « *Mise au point d'algorithmes répartis dans un environnement fortement variable, et expérimentation dans le contexte des pico-réseaux* », Réseaux et télécommunications [cs.NI]. Institut National Polytechnique de Grenoble - INPG, 2004.
- [32] F.MICHEL, « *Formalisme, outils et éléments méthodologiques pour la modélisation et la simulation multi-agents* ». Système multi-agents [cs.MA]. Montpellier II, 2004.
- [33] N. HELAILI, S. MEKHNACHE, « *Simulation du Routage dans les Réseaux de Capteurs Sans Fils* », Mémoire de Fin de Cycle, Université de Bejaia, 2017.

- [34] M. KABIR, I .SYFUL, H. JAVED, & H. SAZZAD, « *Detail comparison of network simulators* », International Journal of Scientific & Engineering Research, Volume 5, Issue 10, October-2014.
- [35] J. HELKEY, J. HOLDER, & B.SHIRAZI, « *Comparison of simulators for assessing the ability to sustain wireless sensor networks using dynamic network reconfiguration* », Sustainable Computing: Informatics and Systems, 9, 1–7, 2016.
- [36] A. KUMAR, S.K. KAUSHIK, R. SHARMA, & P. RAJ, « *Simulators for wireless networks: A comparative study* ». International Conference on Computing Sciences, 2012.
- [37] M. NIAZI, A. HUSSAIN, « *Agent-based tools for modeling and simulation of self-organization in peer-to-peer, ad hoc, and other complex networks* ». IEEE Communications Magazine, 47(3), 166–173, 2009.
- [38] M. BABIS, P. MAGULA, P, « *NetLogo - An alternative way of simulating mobile ad hoc networks* », Wireless and Mobile Networking Conference, 2012.
- [39] H. ALHARBI, A. HUSSAIN, « *An Agent-Based Approach for Modelling Peer to Peer Networks* », 17th International Conference on Computer Modelling and Simulation, 2015
- [40] Simon J. E. Taylor, « *Agent-Based Modeling and Simulation* », Palgrave Macmillan UK, 2014.

Références webographiques

- [41] <https://www.cs.purdue.edu/homes/park/cs536-wireless-3.pdf> [Consulté le 05/09/2020]
- [42] <http://www-public.int-evry.fr/~gauthier/> (VINCENT, GAUTHIER : support de cours simulation) [Consulté le 03/09/2020]
- [43] http://www.ciad-lab.fr/wp-content/uploads/Galland_Teaching_NetLogo.pdf (Stéphane GALLAND) [Consulté le 23/08/2020].
- [44] https://www.geog.leeds.ac.uk/courses/level3/geog3150/practicals/practical1/2_basics.php [Consulté le 16/09/2020]

Résumé

Nous avons analysé le simulateur multi agents NETLOGO et identifié plusieurs caractéristiques et avantages de NETLOGO par rapport aux simulateurs de réseaux traditionnels qui prédéterminent l'application NETLOGO pour la modélisation et la simulation de réseaux ad hoc, en particulier pour simuler les aspects les plus élevés des réseaux mobiles ad hoc à grande échelle (MANETs). Dans ce projet, nous avons conçus et mis en œuvre un modèle de réseau MANET qui permet de simuler en particulier les principaux protocoles et aspects liés aux deux couches PHY et MAC de la norme IEEE 802.11 utilisées pour les réseaux Ad hoc (transmission radio et contrôle d'accès au canal CSMA/CA DCF), la mobilité des nœuds, ainsi que le mécanisme de routage à court chemin qui assure l'acheminement des paquets dans le réseau.

Mots-clés : NETLOGO, simulation et modélisation, réseau mobile ad hoc (MANET), IEEE 802.11, Transmission radio, CSMA/CA, mobilité, routage.

Abstract

We analyzed the NETLOGO multi agent simulator and identified several features and advantages of NETLOGO over traditional network simulators that predetermine the NETLOGO application for modeling and simulating ad hoc networks, in particular for simulating the higher aspects of Large-scale mobile ad hoc networks (MANETs). In this project, we have designed and implemented a MANET network model which makes it possible to simulate in particular the main protocols and aspects related to the two PHY and MAC layers of the IEEE 802.11 standard used for Ad hoc networks (radio transmission and channel access control CSMA / CA DCF), the mobility of nodes, as well as the short-path routing mechanism that ensures the routing of packets in the network.

Keywords : NETLOGO, modeling and simulating, mobile ad hoc networks (MANET), IEEE 802.11, radio transmission, CSMA/CA, mobility, routing.