ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

THÈSE PRÉSENTÉE À
L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

COMME EXIGENCE PARTIELLE
À L'OBTENTION DU
DOCTORAT EN GÉNIE
PH.D.

PAR
EL BAZZAL, Zouhair

LA CLUSTÉRISATION DES RÉSEAUX SANS FIL *AD HOC* : UNE APPROCHE EFFICACE ET PERFORMANTE

MONTRÉAL, LE 16 OCTOBRE 2008

CETTE THÈSE A ÉTÉ ÉVALUÉE

PAR UN JURY COMPOSÉ DE :

M. Michel Kadoch, directeur de thèse
Département de génie électrique à l'École de technologie supérieure

M. Pierre Bourque, président du jury
Département de génie logiciel et des TI à l'École de technologie supérieure

M. François Gagnon, membre du jury
Département de génie électrique à l'École de technologie supérieure

M. Chadi El-Zammar, membre du jury
CMLabs Technologies

ELLE A FAIT L'OBJET D'UNE SOUTENANCE DEVANT JURY ET PUBLIC

LE 7 OCTOBRE 2008

À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

REMERCIEMENTS

Je souhaite exprimer ma plus profonde gratitude à mon directeur de recherche le professeur Michel Kadoch de m'avoir encadré en partageant son expertise scientifique avec moi et en m'insufflant le goût de la recherche. Outre ses conseils avisés, je le remercie spécifiquement de m'avoir soutenu moralement pendant les moments difficiles que j'ai vécus durant la période de cette thèse.

Je remercie tout particulièrement les membres de mon jury de thèse, de m'avoir fait l'honneur d'y participer, de juger ces années de travail et d'en être les rapporteurs. J'adresse mes très sincères remerciements au professeur Pierre Bourque d'avoir présidé ce jury. Je remercie vivement le professeur François Gagnon d'avoir accepté de faire partie du jury et pour le temps qu'il a consacré à des discussions si fructueuses. Je tiens également à remercier Docteur Chadi El-Zammar d'avoir accepté être le membre externe du jury, j'ai beaucoup apprécié sa participation aussi bien que ses remarques constructives.

Mes autres remerciements iront au Docteur Basile L. Agba qui a pris part à de nombreuses discussions et enrichi mes connaissances concernant la couche physique des réseaux sans fil. Un grand merci pour tous les conseils constructifs.

J'ai une pensée particulière pour mes collègues de longue date du laboratoire de recherche LAGRIT à l'École de technologie supérieure (*Kais, Hachem, Wafic, Mohamad, Achour, Ousama, Con, Nabila*), avec qui j'ai partagés ces années dans une ambiance pleine d'amitié et de collaboration.

J'ai aussi une pensée très émue pour ma chère épouse, sans qui cette thèse n'aurait jamais vu le jour. Elle a partagé avec moi chaque moment de bonheur et de malheur. Je te remercie Lydia de ta patience, ton humanisme et tes encouragements qui m'ont été très précieux. Je dédie cette thèse à toi et à mon petit ange, Karim, le fruit de notre mariage et qui, avec ses

beaux sourires, était ma source d'espoir et m'incitait dès sa naissance à penser à améliorer chaque lendemain.

Finalement, je dédie ce travail à mes chers parents, mes frères, mes sœurs et tous les amis. Je vous remercie profondément de mon cœur pour les encouragements, la confiance et le soutien moral malgré la distance. Je vous suis infiniment reconnaissant pour le rôle essentiel que vous avez eu pour la réussite de cette thèse.

Merci à tous.

LA CLUSTÉRISATION DES RÉSEAUX SANS FIL *AD HOC* : UNE APPROCHE EFFICACE ET PERFORMANTE

EL BAZZAL, Zouhair

RÉSUMÉ

Un réseau sans fil *Ad hoc* consiste en un groupe de nœuds mobiles facilement déployables sans l'existence d'une infrastructure préexistante. Chaque nœud joue le rôle d'un hôte/routeur et utilise le canal radio comme le seul moyen de communication. Le routage classique s'appuie souvent sur une vue à plat du réseau *Ad hoc* et utilise un mécanisme d'inondation des paquets de contrôle afin de maintenir les routes. Cette inondation crée des problèmes d'utilisation inefficace des ressources en termes de traitement, de capacité, d'énergie et de bande passante. Aucun protocole de routage ne permet le déploiement à grande échelle du réseau *Ad hoc*; la raison principale est le manque d'une hiérarchie. Dans cette thèse, nous proposons de découper le réseau en clusters afin de réduire la complexité de la topologie et de bâtir un réseau mis à l'échelle.

Chaque cluster comprend un chef qui joue le rôle d'un point d'accès mobile. Nous pouvons ainsi imaginer une dorsale virtuelle dynamique servant d'une part à acheminer le trafic de contrôle entre les chefs d'une manière fiable et efficace et d'autre part, de limiter l'effet des changements de topologie et des conditions des nœuds dans les clusters adjacents. Le choix du nœud-chef est sujet à plusieurs critères en termes de performances (stabilité, mobilité, énergie, capacité, etc.). Ces critères constituent des métriques à considérer dans les calculs analytiques et les simulations.

Pour que les clusters puissent s'adapter dynamiquement aux changements de l'environnement, nous avons proposé un mécanisme de maintenance distribuée permettant de gérer les adhésions, les *handoff* des nœuds et la réélection des nœuds-chefs. Parallèlement, un modèle analytique élaboré a été proposé pour faire les estimations nécessaires des paramètres de qualité de service. Ce qui nous a également permis d'intégrer un mécanisme de contrôle d'admission sur les nœuds-chefs, permettant d'éviter les congestions et toutes les perturbations intraclusters. Ceci nous a amené à balancer la charge entre les différents clusters et d'optimiser l'utilisation des ressources du réseau *Ad hoc*. Les résultats numériques obtenus nous ont aidés à réajuster les paramètres utilisés dans la clustérisation afin de fournir une meilleure qualité de service aux applications utilisées.

Finalement, pour valider nos résultats, nous avons défini quelques paramètres utiles à l'étude des performances du modèle. Les résultats des simulations ont clairement montré le niveau de stabilité des clusters en présence de mobilité et en termes de nombre de clusters formés, nombre de ré-affiliations, nombre de transitions sur les chefs, niveau de qualité de

service globale, équilibrage de charge et passage à l'échelle. Nous avons aussi remarqué une meilleure connectivité et un niveau de stabilité élevé en comparant notre modèle avec un autre de la littérature et en présence de mobilité.

Mots clés : réseaux *Ad hoc*, clusters, maintenance, qualité de service, mise à l'échelle.

CLUSTERING IN WIRELESS *AD HOC* NETWORKS: AN EFFICIENT AND PERFORMANT APPROACH

EL BAZZAL, Zouhair

ABSTRACT

In recent years, most research is focusing on clustering approaches for *Ad hoc* networks because of its effectiveness in building a virtual backbone formed by a set of suitable *clusterhead* (CH) to guarantee the communications across clusters. Clustering has been proven to be a promising approach for mimicking the operation of the fixed infrastructure and managing the resources in *Ad hoc* networks. An *Ad hoc* network is characterized by a collection of wireless nodes which communicate with each other using high-frequency radio waves. These nodes arbitrarily and randomly change their locations and capabilities without the aid of any fixed infrastructure.

We propose a clustering algorithm in order to elect suitable nodes' representatives, i.e. CH and to store minimum topology information by reducing the propagation of routing information which facilitates the spatial reuse of resource and increase the system capacity. We use the node degree, the remaining battery power, the transmission power, and the node mobility for the CH's election. Each CH will act as a temporary base station within its zone or cluster and communicates with other CHs. Thus, packets for route finding may only spread among CHs instead of flooding among all nodes. On the other hand, the topology change information caused by movement of some nodes is limited in adjacent clusters, not in the whole network.

The clusters must adapt dynamically to the environment changes, we proposed a distributed maintenance procedure that allows managing nodes' adhesion, nodes' handoff and the re-election of the CHs. In parallel, an elaborate analytical model was proposed to estimate the quality of service parameters (local allowed saturation throughput, delay and packet error rate). Based on the results from the analytical model, we implement a new admission control algorithm in order to determine the number of members inside a cluster that can be accommodated while satisfying the constraints imposed by the current applications. In this matter, the estimated knowledge of the number of members sharing a cluster might effectively drive congestion avoidance on the CH and interclusters load-balancing to achieve better network resource utilization. The obtained results will help us to readjust the used parameters of the clustering algorithm in order to provide better maintenance and quality of service guarantees depending on the used applications.

Through numerical analysis and simulations, we have studied the performance of our model and compared it with that of other existing algorithms. The results demonstrate the superior

performance of the proposed model in terms of number of clusters, number of re-affiliations, number of transitions (state change) on CH, quality of service, load balancing and scalability. We also observed how the connectivity and the stability are maximized when the number of nodes increases in presence of mobility.

Keywords : *Ad hoc* networks, clusters, maintenance, quality of service, scalability.

TABLE DES MATIÈRES

Page

LISTE DES FIGURES

Page

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

3hBAC	3-hop Between Adjacent Clusterheads
ABR	Associativity-Based Routing Protocol
AC	Access Category
ACK	Acknowledgment
AEDCF	Adaptive Enhanced DCF
AMC	Adaptive Multi-hop Clustering
AMPS	Advanced Mobile Phone System
AODV	*Ad hoc* On-Demand Distance-Vector
AQR	Adaptive Quality of service Routing
BCN	Backbone Capable Node
BER	Bit Error Rate
BGP	Border Gateway Protocol
BPL	Battery Power Level
BSIG	Bluetooth Special Interest Group
BSS	Basic Service Set
CAP	Contention Access Period

CBRP	Cluster Based Routing Protocol
CDMA	Code Division Multiple Access
CDS	Connected Dominating Set
CEDAR	Core Extraction Distributed *Ad hoc* Routing
CFP	Contention Free Period
CH	Clusterhead
CLI	Cross-Layer Interface
CP	Contention Period
CPU	Central Processor Unit
CRC	Cyclic Redundancy Check
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
CTS	Clear To Send
CW	Contention Window
D-AMPS	Digital Advanced Mobile Phone System
DBASE	Distributed Bandwidth Allocation/Sharing/Extension
DCA	Distributed Clustering Algorithm

DCF	Distributed Coordination Function
DCS 1800	Digital Cellular System 1800
DDCA	Distributed Dynamic Clustering Algorithm
DDR	Distributed Dynamic Routing
DFS	Distributed Fair Scheduling
DHCP	Dynamic Host Configuration Protocol
DIFS	Distributed Inter Frame Space
DLBC	Degree Load Balanced Clustering
DMAC	Distributed Mobility Adaptive Clustering algorithm
DNS	Domain Name Server
DS	Distribution System
DSCP	Differentiated Service Code Point
DSDV	Destination Sequenced Distance Vector
DSL	Digital Subscriber Line
DSR	Dynamic Source Routing
DSSS	Direct Sequence Spread Spectrum
EAP	Extensible Authentification Protocol

ECA	Efficient Clustering Algorithm
ECGRID	Energy Conserving GRID
ED	Elected Degree
EDCF	Enhanced DCF
EDGE	Enhanced Data for GSM Evolution
ESS	Extended Service Set
FCC	Federal Communications Commission
FDMA	Frequency Division Multiple Access
FHSS	Frequency Hopping Spread Spectrum
FQMM	Flexible QoS Model for Mobile ad hoc Networks
FSK	Frequency Shift Keying
GPRS	General Packet Radio Service
GPS	Global Positioning System
GPSR	Greedy Perimeter Stateless Routing
GSM	Global System Mobile
HCF	Hybrid Coordination Function
HSCSD	High Speed Circuit Switched Data

HSDPA	High Speed Downlink Packet Access
HyperLAN	High performance Local Area Network
HyperPAN	High performance Personal Area Network
IBSS	Independent Basic Service Set
IDSR	Improved DSR
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IFS	Inter Frame Space
IGP	Interior Gateway Protocol
INORA	INSIGNIA Temporally Ordered Routing Algorithm
INSIGNIA	In-Band Signaling Support for QoS
IP	Internet Protocol
IS-95	Interim Standard-95
ISM	Industrial, Science, Medical
LAN	Local Area Network
LANMAR	Landmark *Ad hoc* Routing
LAR	Location-Aided Routing

LCC	Least Clusterhead Change
LLC	Logical Link Control
MAC	Media Access Control
MACA/PR	Multiple Access Collision Avoidance with Piggyback Reservation
MACAW	Multiple Access Collision Avoidance protocol for Wireless LANs
MAN	Metropolitan Area Network
MANET	Mobile *Ad hoc* NETwork
MBWA	Mobile Broadband Wireless Access
MCC	Maximum Connectivity Clustering
MCDS	Minimum Connected Dominating Set
MIMO	Multiple Input Multiple Output
MOBIC	Mobility Based Metric for Clustering in Mobile *Ad hoc* Networks
MPR	Multipoint Relay
MSDU	MAC Service Data Unit
NACK	Negative Acknowledgment
NAV	Network Allocation Vector
NMT	Nordic Mobile Telecommunication

OFDM	Orthogonal Frequency Division Multiplexing
OLMQR	On demand Link-state Multipath QoS Routing
OLSR	Optimized Link State Routing
OPNET	OPen NETwork
OQR	On-demand QoS Routing
OSI	Open System Interconnection
OSPF	Open Shortest Path First
PA-VBS	Power-Aware Virtual Base Stations
PAN	Personal Area Network
PCF	Point Coordination Function
PDA	Personal Digital Assistant
PDC	Personal Digital Cellular
PHY	Physical
PIFS	PCF Inter Frame Space
PLBQR	Predictive Location-based QoS Routing
PN	Pseudo-random Noise
PPM	Pulse Position Modulation

PRTMAC	Proactive Real Time MAC
PSK	Phase Shift Keying
QoS	Quality of Service
RAN	Regional Area Network
RAS	Remote Activated Switch
RC4	Ron's Code 4
RIP	Routing Interior Procotol
RPGM	Reference Point Group Mobility Model
RREP	Route Reply
RREQ	Route Request
RSA	Rivest, Shamir, Adleman
RSVP	Resource Reservation Protocol
RTS	Ready To Send
RWM	Random Waypoint Mobility
SCFD	Self-Clocked Fair Queuing
SDM	Spatial Division Multiplexing
SIFS	Shortest Inter Frane Space

SNR	Signal to Noise Ratio
SWAN	Stateless Wireless *Ad hoc* Networks
TACS	Total Access Communication System
TBP	Ticket Based Probing
TC	Traffic Category
TCP	Transport Control Protocol
TDMA	Time Division Multiplexing Access
TDR	Trigger-based Distributed Routing
TTL	Time To Live
TZRP	Two Zone Routing Protocol
UDG	Unit Disk Graph
UIT	Union Internationale de Télécommunication
UMTS	Universal Mobile Telecommunications System
UNII	Unlicensed National Information Infrastructure
UTRAN	UMTS Terrestrial Radio Access Network
UWB	Ultra Wide Band
VCS	Virtual Carrier Sense

VMAC	Virtual MAC
VS	Virtual Source
WAN	Wide Area Network
WCA	Weighted Clustering Algorithm
WCDMA	Wideband CDMA
WCDS	Weakly Connected Dominating Set
WEAC	Warning Energy Aware Clusterhead
WEP	Wired Equivalent Privacy
WiMax	Worldwide interoperability for Microwave Access
WIFI	Wireless Fidelity
WPA	Wifi Protected Access
ZRP	Zone Routing Protocol

INTRODUCTION

« Les espèces qui survivent ne sont pas les espèces les plus fortes, ni les plus intelligentes, mais celles qui s'adaptent le mieux aux changements », Charles Darwin[1].

Un réseau sans fil *Ad hoc*, appelé généralement MANET[2], consiste en un réseau *prêt à l'emploi* et comprenant un groupe, relativement dense, de nœuds mobiles qui se déplacent à une vitesse aléatoire et de façon autonome dans un territoire de taille quelconque. Ces réseaux étant rapidement déployables, utilisent les ondes hertziennes comme le seul moyen de communication, et n'exigent pas l'existence d'une infrastructure préexistante ou d'une entité d'administration centralisée.

Un nœud mobile peut être un téléphone cellulaire, un PDA (*Personal Digital Assistant*), un ordinateur portable, ou tout autre équipement permettant une communication sans fil. Cette gamme de nœuds, ayant chacun des contraintes énergétiques, des capacités de traitement et des modèles de mobilité complètement différents, rend le réseau très hétérogène et difficile à être contrôlé de façon simplifiée.

La mauvaise qualité des liens radio et la mobilité des nœuds entraînent une grande quantité d'informations de mise à jour pour la maintenance des routes, ce qui se traduit par une sous utilisation dans un réseau qui souffre déjà du manque de bande passante et d'énergie comparativement aux réseaux filaires où les nœuds (terminaux) sont constamment branchés par le biais des câbles à des sources d'alimentation électrique et à des prises réseau.

Par conséquent, toute la philosophie des réseaux classiques doit être repensée afin d'adapter les réseaux *Ad hoc* à de telles exigences. Alors qu'auparavant, toutes les fonctions

---

[1] Naturaliste anglais (1809-1882)
[2] Groupe de recherche Mobile Ad hoc NETworks, http://www.ietf.org/html.charters/manet-charter.html

intelligentes dans le réseau étaient à la charge de quelques équipements dédiés tels que les routeurs, qui se chargeaient de l'allocation des adresses IP (*Internet Protocol*) par le moyen des serveurs DHCP (*Dynamic Host Configuraton Protocol*), de l'acheminement des paquets et de l'échange des tables de routage; un nœud dans un réseau *Ad hoc* est à la fois, station et routeur; il doit être capable de s'autoconfigurer sans l'aide d'un serveur DHCP centralisé de façon à éviter tout conflit d'adresses.

En outre, la portée de transmission étant limitée, il est fréquent que les communications ne puissent pas se faire de point à point, le relai est souvent nécessaire et tous les nœuds doivent collaborer pour localiser la destination recherchée afin de lui acheminer les paquets.

D'une façon générale, les applications ayant recours aux réseaux *Ad hoc* couvrent un très large spectre, incluant l'organisation d'événements tels que des conférences et des manifestations. Ces réseaux sont également exploités dans des environnements militaires et de tactique comme les opérations de secours (incendies, tremblement de terre, catastrophe naturelle, etc.) et les missions d'exploration. Les réseaux hybrides constituent une autre application civile prometteuse qui permet d'étendre le concept des réseaux sans fil classiques pour créer des réseaux d'accès cellulaires multisauts : en élargissant la couverture radio, il serait possible d'autoriser tant les connexions vers l'Internet qu'entre les paires de nœuds mobiles.

Problématique de recherche

Tout d'abord, nous allons faire un survol des problématiques dans le domaine des réseaux sans fil *Ad hoc* en les comparant à celles des réseaux filaires.

Dans les réseaux filaires, chaque couche du modèle OSI (*Open System Interconnection*) offre ses propres services (les services de transport, de routage, de liaison de données, etc.). L'existence des câbles et des entités fixes intelligentes dans le réseau simplifie les problématiques rencontrées au niveau de différentes couches. Dans ce contexte, la qualité

des liens est prédictible et maîtrisable. Elle dépend des caractéristiques matérielles du câble et du trafic y circulant. Ainsi, le routage devient une tâche moins compliquée étant donné que les métriques utilisées sont facilement mesurables (bande passante, charge, délai, nombre de sauts, taux de perte, etc.). Il suffit d'appliquer un algorithme sur un routeur fixe en spécifiant les métriques que nous jugeons intéressantes en vue de construire une table de routage pertinente, offrant des routes ayant des contraintes de qualité de service.

Dans les réseaux sans fil *Ad hoc*, le modèle de service est complètement bouleversé. Ceci impose des changements importants au niveau de chaque couche de la pile protocolaire, voire à la notion de la qualité de service elle-même. En effet, ceci est fortement dépendant de la qualité du réseau où les nœuds *Ad hoc* sont très mobiles et jouent le rôle des hôtes et des routeurs simultanément. Ces nœuds doivent s'adapter aux conditions du canal radio, aux trafics et aux différents mouvements pouvant intervenir au sein de leur entourage.

La question qui se pose constamment dans les communautés de recherche et qui continue à être débattue à l'IETF[3] (*Internet Engineering Task Force*) concerne le routage avec l'intégration de la qualité de service dans les réseaux *Ad hoc*. Y a-t-il des ressources dans un environnement suffisamment proche pour permettre une communication de bonne qualité et comment peut-on utiliser ces ressources d'une façon optimale ? Vaut-il la peine de maintenir assez fréquemment les tables de routage qui changent constamment dans un contexte *Ad hoc* ou n'est-il pas plus judicieux de déterminer la table de routage au moment de l'arrivée des paquets ?

En effet, l'échange des tables de routage, suite aux variations des routes, est une vraie problématique dans les réseaux *Ad hoc*. D'une part, le fait de garder une connectivité maximale nécessite un envoi continu des tables de routage au détriment de la bande passante du médium radio et des ressources des stations (batterie, capacités de traitement,

[3] Organisme de normalisation des standards Internet

mémoire, etc.). [Tseng *et al.* (2002)] ont déjà étudié ce problème sous le nom de tempête d'inondation (*Broadcast Storm Problem*). D'autre part, le fait de réagir lorsque le flux des paquets est prêt à être émis, exige un délai d'attente supplémentaire pour les applications temps réel, mais améliore énormément les performances du réseau tant au niveau du trafic de contrôle qu'au niveau des ressources des stations.

La plupart des propositions existantes essayent de calquer les approches des réseaux filaires, bien qu'elles soient peu optimisées pour les réseaux sans fil à multisauts comme *Ad hoc*. En effet, les interférences entre les liens et la mobilité des nœuds, rendent difficile la détermination des chemins sur plusieurs sauts ayant un certain niveau de qualité de service. Ceci montre un fort contraste comparativement aux réseaux filaires où les algorithmes de routage optimaux sont polynomiaux.

Dans le cadre de nos travaux de recherche, nous nous sommes intéressés aux problèmes engendrés par le routage, la mobilité, la gestion de la topologie et la qualité de service des applications. Dans le but de limiter la quantité de trafic de contrôle échangé entre les différentes stations pour mettre à jour la carte de la topologie, nous proposons un modèle basé sur la clustérisation (ou bien le regroupement en clusters). Le cluster permet de regrouper les nœuds afin de créer de véritables zones de services limitées dans la taille en vue d'y appliquer les différentes ressources de réseaux et de faciliter les mécanismes de gestion, de contrôle et de routage.

Ainsi, la mobilité sera traitée localement au sein de chaque cluster, ce qui limite l'effet des changements topologiques au cluster lui-même. Les clusters, avec des chefs attitrés (*clusterhead*), peuvent prendre en charge des fonctionnalités de localisation des mobiles, de notification (*paging*) et de diffusion des informations de topologie. Nous proposons un mécanisme d'élection des chefs basé sur l'attribution d'un poids assigné aux nœuds. Ce poids dépend du niveau de stabilité d'un nœud dans le réseau; il est assez flexible de façon à s'adapter à son environnement et permettra de sélectionner le nœud qui se conforme aux exigences du *clusterhead*.

Ces clusters dynamiquement contrôlés permettent de développer de nouvelles solutions pour les réseaux *Ad hoc*. Le choix du nœud-chef doit subir plusieurs critères en termes de performances, de stabilité, de mouvement, d'énergie, etc. Ces chefs, en tant que points d'accès virtuels mobiles, permettront d'appliquer tout modèle de routage entre les différents clusters et d'acheminer les paquets de bout en bout tout en limitant les échanges et la taille des tables de routage.

Dans les réseaux *Ad hoc*, la qualité de service soulève plusieurs problématiques; elle doit s'appliquer au flux, au lien et au nœud. Les caractéristiques de ces réseaux font de l'intégration de la qualité de service un vrai défi, parmi lesquelles nous citons les suivantes :

- variation dynamique de la topologie du réseau *Ad hoc*, ce qui provoque la rupture des liens et crée des reconfigurations très fréquentes;
- impossibilité de maintenir localement et d'une façon précise les informations sur les états des liens et des flux : les nœuds maintiennent des informations sur les états par lien et par flux. L'état par lien indique la bande passante, le délai, la gigue, le taux de perte, le taux d'erreur, la stabilité, le coût et les valeurs de distance de chaque lien. L'état par flux inclut l'identificateur de la session, l'adresse source, l'adresse destination et les besoins du flux en termes de QoS (*Quality of Service*) (comme la bande passante maximale et minimale, le délai maximal et la gigue maximale); ces états sont imprécis suite aux changements de la topologie du réseau et des conditions du canal;
- manque d'un point de coordination centrale, taux d'erreur élevé sur le canal radio, problème des stations cachées/exposées ainsi que la disponibilité des ressources (bande passante, batterie, espace de stockage et capacité de traitement des nœuds) engendrent tous des vrais défis.

Ainsi, la qualité de service nécessaire pour les applications est fortement dépendante de la qualité du réseau : Il faut contrôler un ensemble de paramètres pour adapter l'application aux conditions du réseau pendant que les données sont acheminées.

Comme nos clusters sont utilisés pour des fins de routage, il est essentiel de traiter la problématique de la qualité de service dans un réseau *Ad hoc*. Aujourd'hui, IEEE 802.11 WLAN (*Wireless Local Area Network*) est considéré comme un Ethernet sans fil en vertu de son service *best effort* basé sur le protocole de contrôle d'accès MAC similaire à celui d'Ethernet, mais il n'offre aucune notion de qualité de service. Avec le modèle *best effort* ordinaire, les performances dépendent étroitement du nombre d'utilisateurs et des risques d'interférences par recouvrement de cellules.

Les solutions proposées actuellement ne satisfont pas les contraintes de la qualité de service surtout pour les applications temps réel. Un aspect important à respecter est d'adapter les clusters au niveau de service désiré. Par exemple, les applications multimédia ont de fortes contraintes en termes de bande passante, de délai et de gigue. Les applications militaires ont des contraintes de sécurité. Les applications d'urgence et de secours requièrent la disponibilité continue du réseau. D'autres applications comme les communications d'un groupe dans une salle de conférence requièrent un minimum de consommation d'énergie et de batterie. Évidemment, la taille d'un cluster en termes de nombre de stations actives influence directement le niveau de QoS offert dans le cluster. Le canal radio est de nature partagée, ce qui relève des problèmes de contention pour y gagner accès. Ceci occasionne des délais d'attente pour un canal libre. Il est ainsi primordial d'optimiser le nombre de stations dans un cluster de façon à offrir un débit, un délai et un taux de perte acceptables par les utilisateurs.

Nous proposons dans ce contexte un modèle analytique élaboré permettant de faire les estimations nécessaires des paramètres de QoS dans un cluster. Ce qui nous a également permis d'implémenter un mécanisme de contrôle d'admission sur les nœuds-chefs, permettant ainsi d'éviter les congestions et toutes les perturbations intraclusters. Ceci nous a amenés à balancer la charge entre les différents clusters construits dans le réseau *Ad hoc* pour optimiser l'utilisation des ressources réseau tout en intégrant un mécanisme de maintenance efficace permettant de gérer de façon transparente l'adhésion, la fusion, le *handoff* des nœuds dans les clusters.

Méthodologie et objectifs de recherche

Présentement, il existe une multitude d'approches essayant de remédier aux problématiques que nous venons de décrire. Il existe une large gamme de protocoles de routage pour les réseaux *Ad hoc* standardisés au sein de l'IETF; d'autres sont encore dans un état embryonnaire. Après avoir étudié toutes ces approches, nous avons conclu que la majorité des recherches se concentrent actuellement sur la hiérarchisation de ces réseaux afin de réduire leur complexité étant donné qu'un routage à plat ne permet pas d'étendre les solutions à venir. C'est de là qu'est née notre vision qui consiste à découper le réseau en des clusters en vue de réduire la complexité d'une large topologie et de bâtir un réseau mis à l'échelle.

Ainsi, nous pouvons imaginer une dorsale (*backbone*) virtuelle dynamique servant à acheminer le trafic de contrôle entre les chefs de chaque cluster d'une manière fiable et efficace. Sur la dorsale, un regroupement de nœuds en clusters sert à gérer leur mobilité. Une telle structure devra s'adapter dynamiquement aux changements de l'environnement et permettra donc, de déployer sur ces réseaux certaines fonctionnalités des réseaux filaires.

Vu que la solution proposée tient sur la clustérisation, nous débutons ce projet par une étude bibliographique approfondie sur les différentes approches touchant notre domaine de recherche. Nous avons fait une étude élaborée sur les fonctionnalités de chaque approche, leurs champs d'application, leurs avantages et leurs inconvénients. Nous avons finalement conclu que chacune de ces approches était implémentée pour répondre aux besoins spécifiques de leurs champs d'application. Les contributions de ces approches étaient vulnérables à de fortes mobilités des nœuds.

Nos travaux sur la clustérisation prenant en considération la QoS apportent des solutions à ces problèmes tout en gardant une bonne performance du réseau. En effet, ces travaux consistent à diviser la topologie en plusieurs clusters; chaque cluster comprend un chef (*clusterhead*) qui joue le rôle d'un point d'accès mobile. Le choix du nœud-chef est sujet à

plusieurs critères en termes de performance (stabilité, mobilité, énergie, capacité, etc.). Ces critères constituent des métriques à considérer dans nos calculs analytiques et nos simulations.

Étant donné que les nœuds-chefs sont beaucoup plus performants et stables que les nœuds ordinaires, alors ces chefs auront la responsabilité de maintenir une carte topologique (table de routage) et d'acheminer le trafic vers l'extérieur de sa zone. Ce sont seulement ces nœuds qui échangeront les tables de routages. Des routes seront alors créées entre les nœuds-chefs afin d'étendre le réseau et de rejoindre les zones lointaines.

Les réseaux *Ad hoc* étant peu déployés, le seul outil que nous détenons pour les étudier est l'utilisation des simulateurs. Le choix d'un simulateur est un facteur primordial. Nous avions le choix entre une multitude de simulateurs à la portée des chercheurs. OPNET[4] semblait au début un bon choix en raison de sa forte réputation comme un simulateur très puissant dans les réseaux classiques. Cependant, nous avons opté pour le simulateur de Agba *et al.* (2006a).

Ce simulateur permet de simuler la réalité en générant des scénarios très réalistes. Il tient compte de différents modèles de mobilité que nous rencontrons dans la littérature pour simuler le mouvement des nœuds, ainsi que d'une carte topographique en trois dimensions (3D) implémentant le relief, les obstacles (murs, *building*, objets, etc.). Agba *et al.* (2006b) ont aussi exposé l'avantage de ce simulateur sur OPNET, surtout au niveau de la couche physique. En effet, la couche physique d'OPNET n'est pas optimisée pour les réseaux sans fil. Elle a été parfaitement conçue pour des réseaux câblés sans tenir compte de différentes perturbations qui peuvent avoir lieu sur les canaux sans fil (effet multi-trajets, *path loss, fading*, atténuation, etc.).

---

[4] OPen NETwork, un simulateur réseau

Le choix du modèle de mobilité était aussi assujetti à plusieurs discussions. Étant donné que nous ciblons des champs d'applications dans des salles de conférence, dans des zones ayant une taille limitée, nous avons utilisé le modèle *Random Waypoint Mobility* (RWM) qui est pertinent pour simuler ces situations. Ce modèle suppose que le nœud choisit une destination pour s'y rendre à une vitesse déterminée. Rendu à la destination, le nœud s'arrête pendant un laps de temps avant de choisir une autre destination et une autre vitesse.

Ayant franchi cette étape, nous avons maintenant les outils pour bâtir un réseau *Ad hoc*. La prochaine étape est la clustérisation. La clé magique de la clustérisation est la méthode d'élection des chefs ainsi que les critères et les contraintes qu'un nœud doit posséder pour qu'il soit éligible à jouer ce rôle. Nous avons proposé une approche de clustérisation efficace pour les réseaux mobiles *Ad hoc*, nommée ECA (*Efficient Clustering Algorithm in Mobile Ad hoc Networks*). Cette approche a la flexibilité d'assigner des poids aux nœuds. Le nœud ayant le meilleur poids sera élu comme *clusterhead*.

Pour garantir le bon fonctionnement et la stabilité de nos clusters durant leur durée de vie, une procédure de maintenance s'avère nécessaire. Citons un simple exemple d'un *clusterhead* ayant des contraintes énergétiques limitées, il est essentiel que ce dernier résilie son rôle avant qu'il ne gaspille complètement sa batterie à cause d'une haute utilisation.

Ainsi, notre modèle développé diffère des autres existants dans la littérature par les aspects suivants :

- simuler un vrai canal radio tenant compte de toutes ses caractéristiques;
- assigner des poids aux nœuds, le poids peut être une combinaison de plusieurs métriques comme la puissance de transmission d'un nœud, son degré de mobilité (vitesse), son niveau de batterie restante et sa capacité en termes de nœuds membres;
- maintenir les structures construites pour s'adapter aux changements dans les conditions des nœuds et du réseau ainsi que pour éviter la surcharge des *clusterhead* en effectuant au besoin la réélection de nouveaux *clusterhead* et en étalant l'utilisation des ressources

(surtout les batteries) sur plusieurs nœuds afin de réaliser un balancement de charge dans tout le réseau;

- former des clusters stables avec un nombre réduit de *clusterhead* tout en minimisant le nombre d'exécutions des algorithmes de formation et de maintenance et maximisant la durée de vie des nœuds (batteries);
- exécuter la réélection là où il faut. Ceci ne se fait plus à des intervalles de temps assez fréquents comme dans la plupart des approches existantes. Nous le ferons seulement quand le poids change suite à des conditions établies dans l'algorithme. Ce qui optimise l'*overhead*, le traitement à faire au niveau des nœuds et les ressources à utiliser;
- optimiser la consommation de l'énergie, de mémoire et de CPU (*Central Processor Unit*) des nœuds en limitant les tâches lourdes seulement au *clusterhead* qui sera responsable de l'échange des tables de routage et des messages de contrôle avec les *clusterhead* avoisinants;
- utiliser le facteur de mobilité relative des nœuds durant la procédure de maintenance pour maximiser le niveau de stabilité. Ceci est basé sur l'historique de puissance des signaux reçus;
- proposer un modèle analytique permettant de limiter le nombre de nœuds dans chaque cluster en vue de respecter les paramètres de la qualité de service des applications tels que le débit, le délai et le taux de perte. Le fait d'imposer des limites sur le nombre de nœuds actifs dans un cluster permet de restreindre la contention sur le canal et de ne pas dégrader les fonctionnalités de la couche MAC;
- permettre de réajuster à la demande les paramètres utilisés dans les algorithmes pour satisfaire les besoins des clients d'un cluster dépendamment des types d'applications qu'ils utilisent;
- effectuer un contrôle d'admission des requêtes d'adhésion sans altérer le niveau de service des clients déjà admis;
- distribuer des codes CDMA, un code unique par cluster pour limiter les interférences interclusters et éviter la dégradation locale de la qualité de service;
- possibilité d'étendre le modèle pour profiter des techniques MIMO et permettre à un *clusterhead* d'émettre et de recevoir simultanément sur multiples antennes. Ceci lui

permettra de communiquer simultanément avec sa zone (sur un code CDMA) et avec les autres clusters (sur d'autres codes CDMA) sans se soucier des délais d'attente supplémentaires qui peuvent avoir lieu dans un mode de communication sans MIMO.

Dans le but de simplifier la maintenance et l'allocation des codes CDMA, surtout dans des environnements denses et très mobiles, nous ciblons la formation des *1-clusters* où le *clusterhead* est à un seul saut de tous ses membres. En plus, nous nous posons dans le cas où les nœuds sont, du point de vue du réseau, identiques et interchangeables.

Finalement, pour valider nos résultats, nous avons établi quelques paramètres utiles à étudier les performances de notre modèle. Nous avons choisi une des meilleures approches de la littérature basée aussi sur la notion d'assignation des poids aux nœuds. Les résultats des simulations ont clairement montré le niveau de stabilité de la nôtre sur une longue période de temps et sous plusieurs scénarios en termes de nombre de clusters formés, nombre de ré-affiliations, nombre de transitions qui arrivent sur les *clusterhead*, niveau de qualité de service globale, équilibrage de charge et passage à l'échelle (*scalability*).

Plan de la dissertation

La présente dissertation est composée de cinq chapitres. Dans le premier chapitre, nous survolons les différentes solutions dans les réseaux sans fil, précisément celles des réseaux à portée limitée comme le WLAN et le *Bluetooth*.

Dans le deuxième chapitre, une revue de littérature approfondie dans les réseaux *Ad hoc* est présentée. Nous élaborons diverses approches de la littérature tant au niveau physique qu'au niveau liaison de données et routage. Finalement, nous évoquons d'autres axes de recherche tels que le *Cross-Layer* et la qualité de service avant de terminer par une conclusion ouvrant les portes vers d'autres thèmes et perspectives de grand intérêt.

Dans le troisième chapitre, nous présentons l'approche proposée pour remédier aux problématiques décrites dans l'introduction de cette dissertation. Avant d'aller plus loin dans notre approche, nous mettons plus d'emphase sur la hiérarchisation des réseaux *Ad hoc* servant de fil conducteur à cette thèse, nous présentons une revue de littérature des approches basées sur les dorsales virtuelles ainsi que celles utilisant des clusters; nous faisons une étude des principes fondamentaux, des propriétés de telles structures ainsi que de toutes les carences rencontrées.

Quant à notre proposition, elle est constituée d'un modèle complet de formation de clusters, servant de collecteur de trafic, de limiteur d'inondations et de premier élément pour hiérarchiser le réseau. Sur les clusters, viennent se greffer des stations libres de se mouvoir. La stabilité des clusters permet d'optimiser la longévité des routes. Ce qui entraîne une réduction au niveau du trafic de contrôle, du traitement à faire par les nœuds ainsi qu'une meilleure économie des ressources énergétiques et de bande passante. Il est essentiel de considérer la qualité de service qui est un élément clé de l'évaluation de toute proposition dans les réseaux *Ad hoc*. En effet, le canal radio présente une faible bande passante, constituant ainsi une forte contrainte à optimiser. Ainsi, nous essayons, dans ce même chapitre, de répondre à une question qui vient constamment à l'esprit : l'introduction des clusters nuit-elle à la capacité d'un réseau ? Nous proposons un modèle analytique dans l'objectif d'optimiser le nombre de nœuds par cluster de sorte que le réseau puisse équilibrer la charge sur plusieurs clusters tout en garantissant un niveau de QoS acceptable pour les applications utilisatrices.

Dans le quatrième chapitre, nous détaillerons les mécanismes de maintenance de l'architecture proposée, nous y décrivons les différentes procédures utilisées pour faire la gestion des clusters (adhésion, départ, changement de cluster ou *handoff*) ainsi que le mécanisme de réélection des nœuds-chefs les plus appropriés.

Dans le cinquième chapitre, nous procédons à l'implémentation de notre modèle. Comme nous le savons, il est indispensable d'expérimenter réellement les clusters tout en évitant les

problèmes de simplification inhérents à toute modélisation. Le canal radio est largement sous modélisé dans les simulations rencontrées dans la littérature. Nous utiliserons un simulateur très réaliste pour refléter les problématiques de ce canal. La robustesse, la stabilité et la qualité de service sont des propriétés fondamentales pour prouver l'efficacité des clusters; les résultats de simulations ont apporté une corroboration quantitative de ces propriétés, nous observerons des performances très encourageantes surpassant celles des modèles existants en termes de passage à l'échelle, de stabilité et de QoS.

Enfin, nous terminons par une conclusion dressant le bilan et l'originalité des travaux achevés. Et nous exposerons quelques suggestions et visions sur des perspectives futures et potentielles.

CHAPITRE 1

GÉNÉRALITÉS SUR LES RÉSEAUX SANS FIL

1.1 Introduction

L'industrie informatique a connu une forte croissance avec l'arrivée des équipements mobiles sans fil, tels que les ordinateurs portables, les assistants personnels (PDA) et les cellulaires. L'idée des communications sans fil est née aux débuts du XIX siècle. En 1865, James Clerk Maxwell[5] a identifié et prévu le phénomène de propagation des ondes électromagnétiques dans l'espace lors du mouvement des électrons. Vingt-deux ans plus tard, Heinrich Hertz[6] a pu observer ces ondes pour la première fois. En 1901, Guglielmo Marconi[7] a expérimenté une transmission télégraphique sans fil transatlantique depuis un navire vers la côte au moyen du code Morse[8]. Les systèmes modernes offrent sans doute de meilleures performances, mais l'idée reste toujours la même.

Dans son livre, Tanenbaum (2003) prévoit que seuls deux modes de communication persisteront dans le futur : la fibre optique pour les équipements non mobiles et le sans-fil pour les équipements portables et mobiles. Beaucoup d'utilisateurs sont désireux de pouvoir joindre la jungle d'Internet, lorsqu'ils sont en déplacement ou lors de trajets. Puisqu'une connexion filaire dans ces situations où le déploiement du câble est très contraignant, voire impossible, les réseaux sans fil suscitent un très vif intérêt dans notre vie quotidienne et de nombreuses applications ont commencé à se développer rapidement pour répondre aux exigences du marché technologique.

---

[5] Physicien anglais (1831-1879)
[6] Physicien allemand (1857-1894)
[7] Physicien italien (1874-1937)
[8] Méthode de transmission télégraphique inventée par Samuel Morse (1791-1872)

En premier temps, les réseaux sans fil peuvent servir pour interconnecter les éléments d'un système quelconque comme dans le cas d'un ordinateur et ses périphériques (souris, clavier, casques, scanners, etc.) placés dans son rayon de portée et formant un réseau PAN (*Personal Area Network*). Dans sa forme la plus simple, ce type d'interconnexion utilise le modèle maître/esclave baptisé *Bluetooth* où c'est l'ordinateur qui assure le rôle de maître et qui communique avec les divers périphériques « les esclaves ». Il leur fournit les adresses à utiliser, les créneaux de transmissions et la durée de transmission allouée, etc.

En deuxième temps, ces réseaux peuvent servir pour créer des LAN (*Local Area Network*), des MAN (*Metropolitan Area Network*), des WAN (*Wide Area Network*) et des RAN (*Regional Area Network*) sans fil comme dans le cas des communications entre des terminaux équipés d'un modem radio et d'une antenne grâce auxquels ils peuvent s'échanger les données. Il existe des standards pour ce type des réseaux chapeautés par l'organisme de normalisation des standards IEEE (*Institute of Electrical and Electronics Engineers*) que les systèmes de nos jours implémentent de plus en plus.

En troisième temps, ces réseaux peuvent aussi servir pour permettre des communications longue distance comme dans le cas des réseaux cellulaires qui ont déjà connu quatre générations allant de la transmission de voix analogique et numérique à faible bande passante à la transmission du multimédia numérique à large bande.

Les standards IEEE peuvent fonctionner à des débits théoriques atteignant 320 Mbit/s (IEEE 802.11n) pour un rayon de couverture de plusieurs dizaines de mètres, alors que les systèmes cellulaires 3G n'offrent qu'un débit inférieur à 1 Mbit/s quand la distance entre le terminal et la station de base se compte en kilomètres, ce débit atteindrait des centaines de Mbit/s avec l'émergence des réseaux 4G dans les années à venir (2010 à 2015). [Pujolle (2005)].

1.2 Survol des normes Européennes et Américaines

Dès l'apparition des premiers dispositifs portables, nombreux sont ceux qui ont commencé à s'imaginer pouvoir être connectés sans fil à l'Internet en équipant ces dispositifs de transmetteurs radio leur permettant d'émettre sur le canal radio. Diverses sociétés se sont alors lancées dans leur commercialisation sans avoir le moindre souci des problèmes de compatibilité entre les différentes marques. Il était temps de définir des normes pour remédier à ces dilemmes; cette tâche a été prise en main par des comités du IEEE et UIT-T (*Union Internationale des Télécommunications*), et des pionniers de normalisation des standards dans le monde des télécommunications.

La figure 1.1 illustre les différentes catégories des réseaux sans fil selon leur étendue et leur débit. Dans cette section, nous décrivons brièvement ces catégories. Par contre, nous allons mettre l'emphase sur les normes permettant un mode *Ad hoc* comme le *Bluetooth* et les réseaux locaux sans fil WLAN IEEE 802.11 (section 1.4 et 1.5).

Pour les petits réseaux personnels PAN d'une portée de dizaine de mètres, la principale norme est IEEE 802.15 où trois groupes ont normalisé des gammes de produits en parallèle :

- IEEE 802.15.1 qui prend en charge le Bluetooth et qui est largement commercialisé;
- IEEE 802.15.2 qui assure une coexistence entre LAN et PAN sans fil dans la bande de fréquence non licenciée de 2.4GHz;
- IEEE 802.15.3 qui définit la norme UWB (*Ultra Wide Band*) qui permet des débits énormes (1 Gbit/s vers 2010) à une puissance extrêmement faible dans un rayon de 10 mètres;
- IEEE 802.15.4 qui prend en charge le *ZigBee* qui offre des débits faibles, mais à un coût très bas.

En Europe, le groupe HyperPAN (*High performance PAN*) n'est pas assez développé pour que nous puissions avoir une idée assez précise sur ces caractéristiques.

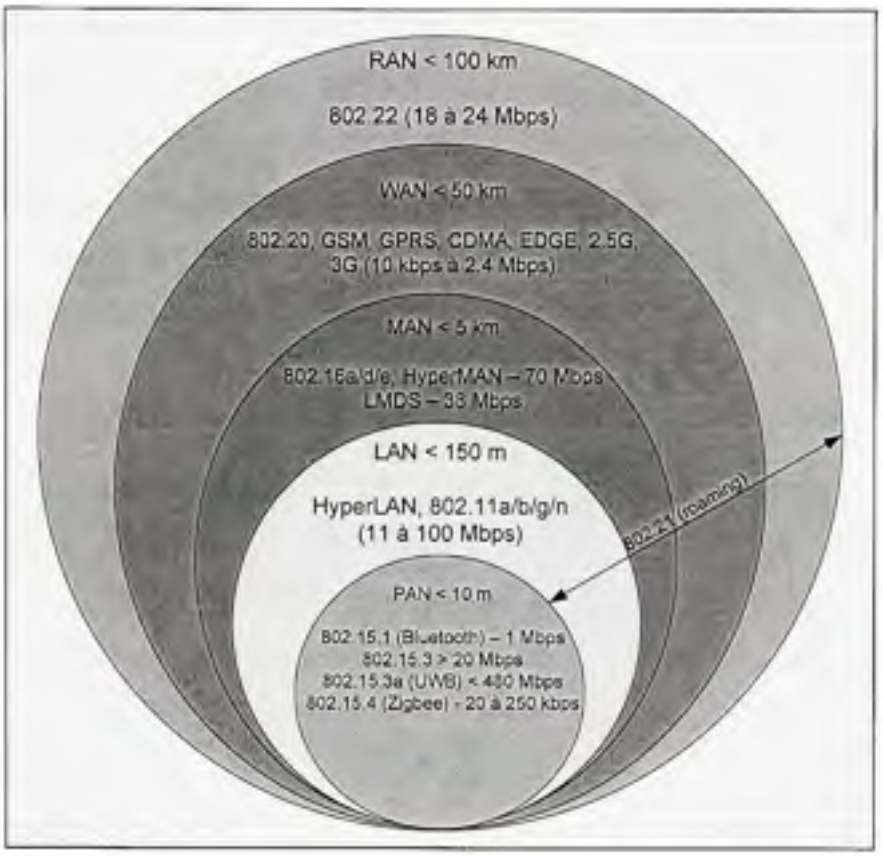

Figure 1.1 ***Étendues et débits de différentes catégories des réseaux sans fil.***

(Tiré de Cordeiro et al., 2005)

Source : Cette figure a été tirée de Cordeiro *et al.* (2005) « IEEE 802.22: The First Worldwide Wireless Standard based on Cognitive Radios » et correspond à la « Figure 2 - 802.22 wireless RAN classification as compared to other popular wireless standards » présentée en page 4 dans le document original. (La référence complète du document est présentée dans la liste de références).

Pour les réseaux locaux LAN, la principale norme est IEEE 802.11 ou WIFI (*Wireless Fidelity*). Il existe aujourd'hui quatre propositions, dont les débits sont de 11 Mbit/s (IEEE 802.11b), 54 Mbit/s (IEEE 802.11a/g) et 320 Mbit/s (IEEE 802.11n). Quant à lui, HyperLAN (*High performance Local Area Network*), pendant Européen de la norme IEEE 802.11, permet des débits atteignant une cinquantaine de Mbit/s dans un rayon de portée de l'ordre de 100 mètres.

Pour les réseaux métropolitains MAN, la principale norme est IEEE 802.16 connu sous le nom de WiMax (*Worldwide interoperability for Microwave Access*) qui avait pour objectif

de remplacer les modems DSL (*Digital Subscriber Line*) que l'on trouve dans les réseaux téléphoniques fixes. À la différence d'IEEE 802.11, WiMax fournit à l'utilisateur final des débits extrêmement importants dépendant de la bande des fréquences utilisées et des rayons de portée qui peuvent atteindre une dizaine de kilomètres. En outre, il est moins sensible aux effets multi-trajets et pénètre mieux à l'intérieur des bâtiments. Ce que lui permet un passage à l'échelle sur de grandes surfaces.

Pour les réseaux WAN à large étendue, les principales normes sont : GSM (*Global System Mobile*), GPRS (*General Packet Radio Service*), EDGE (*Enhanced Data for GSM Evolution*), UMTS (*Universal Mobile Telecommunications System*) et CDMA2000 (*Code Division Multiple Access 2000*). La norme concurrente provenant de la famille IEEE est IEEE 802.20, connu sous le nom de MBWA (*Mobile Broadband Wireless Access*), dont l'objectif est de concurrencer les standards des opérateurs de téléphonie mobile avec des coûts très avantageux.

Pour les réseaux régionaux RAN, la principale norme récemment proposée est IEEE 802.22, dont l'objectif est de fournir un accès large bande dans les zones rurales éloignées. Cette norme opère dans la bande de fréquences des canaux TV non utilisés (54 à 862 MHz).

1.3 Survol des réseaux cellulaires

L'histoire et l'évolution des réseaux cellulaires de la première à la quatrième génération sont discutées dans cette section. Le développement de la première génération 1G a commencé dans les années 70 par « *Bell Labs* » où les premiers terminaux étaient dotés d'une mobilité restreinte et de services limités. La seule technique de transmission utilisée était analogique. Une cellule n'est qu'une zone géographique couverte par une antenne de transmission, les terminaux mobiles sont en mesure de passer d'une cellule à autre (*handoff*) sans coupure de communication. Chaque cellule utilise une bande de fréquences qu'on ne retrouve dans aucune des cellules adjacentes afin d'éviter tout type d'interférences. Le premier réseau cellulaire opérationnel, l'AMPS (*Advanced Mobile Phone System*) est né aux

États-Unis. En revanche, les Européens ont lancé une génération assez similaire, le NMT (*Nordic Mobile Telecommunication system*) dans les pays scandinaves et le TACS (*Total Access Communication System*) au Royaume-Uni.

À la différence des systèmes analogiques de la première génération, ceux de la deuxième génération, conçus dans les années 80, ont permis la transmission numérique de la voix. La précipitation concurrentielle pour concevoir et déployer les systèmes numériques cellulaires a mené à une variété de normes différentes et incompatibles telles que GSM, principalement en Europe et fonctionnant à 900 MHz, DCS 1800 (*Digital Cellular System 1800*) équivalent au GSM, mais fonctionnant à des fréquences plus élevées (1800 MHz), D-AMPS (*Digital AMPS*) version numérique de AMPS, IS-95 (*Interim Standard 95*) et CDMA (*Code Division Multiple Access*) aux États-Unis ainsi que PDC (*Personal Digital Cellular*) au Japon.

Bien que le débit des systèmes 2G soit très limité, ces derniers représentent toujours la technologie d'aujourd'hui la plus déployée. Leurs spécifications définissent le segment radio, le segment réseau et toutes les interfaces entre les éléments du même système offrant une bonne qualité de voix numérique, une sécurité de haut niveau ainsi qu'une itinérance (*handoff*) internationale. Ces systèmes étant numériques, ils permettent la transmission des données non vocales, mais à des débits très faibles (moins de 10 Kbit/s), ce qui a rendu illusoire l'intégration des services multimédias dans ce type des systèmes et a donné naissance à la génération 2.5.

Un système 2.5G, vu en tant qu'un intérim entre 2G et 3G, n'est qu'une évolution des technologies précédentes qui permet des débits plus élevés de l'ordre de 384 Kbit/s. Un aspect très important de ces systèmes est que les canaux sont optimisés pour la transmission des paquets de données. Ceci permet un accès à l'Internet depuis des dispositifs mobiles comme des téléphones et des PDA. Les deux normes principales sont : HSCSD (*High Speed Circuit Switched Data*) et GPRS. Le HSCSD autorise des débits de données commutées de l'ordre de 77 Kbit/s, adapté aux transferts de fichiers volumineux et aux applications

multimédias comme la vidéo mobile. Le GPRS, contrairement au HSCSD, utilise la commutation de paquets, permettant aux ressources radio de n'être utilisées que pendant la transmission des paquets. Ce qui s'avère idéal pour les données se présentant sous forme de rafales. Ainsi, des débits théoriques de 120 Kbit/s peuvent être atteints. En réalité, ce débit plafonne à une trentaine de Kbit/s. Les Américains ont développé des solutions différentes de celles des Européens. L'IS-95B permet des débits beaucoup plus élevés. Cette solution consiste à allouer à l'utilisateur toute la bande de fréquence en se basant sur la technique d'étalement de spectre CDMA.

Le système 3G quant à lui, permet des canaux voix de plus haute qualité, aussi bien la possibilité de transmission de données à large bande, variant de 64 à 144 Kbit/s en forte mobilité, 384 Kbit/s en mobilité moyenne et de 2 Mbit/s à 10 Mbit/s en faible mobilité. L'UMTS et son dérivé HSDPA (*High Speed Downlink Packet Access*) sont les principales normes qui incluent W-CDMA (*Wideband CDMA*), CDMA2000 et EDGE. Le segment radio dans UMTS est nommé UTRAN (*UMTS Terrestrial Radio Access Network*).

W-CDMA, proposé par la société *Ericsson*, se fonde sur la technique d'étalement de spectre par séquence directe DSSS (*Direct Sequence Spread Spectrum*), il dispose d'une bande passante de 5 MHz et permet des *handoff* sans interruption entre une cellule W-CDMA et une autre GSM, mais il n'est pas compatible avec ce dernier. De son côté, CDMA2000 (extension d'IS-95), proposé par la société *Qualcomm*, s'appuie également sur la même technique d'étalement de spectre et dispose de la même bande passante de 5 MHz. Par contre, il ne permet aucun *handoff* vers des cellules GSM.

Malgré l'évolution significative dans les systèmes 3G, une forte demande de bande passante continue à accroître pour les applications d'aujourd'hui. Les technologies de la quatrième génération 4G sont pressenties comme une convergence entre les réseaux 3G et certaines technologies radio, avec pour objectif de fournir une continuité de service de haute qualité sans interruption et des débits importants.

Selon les indications historiques, une révolution de génération se produit une fois chaque décennie, le fabricant *Samsung* a pris de l'avance concernant les réseaux cellulaires de la quatrième génération 4G, ce dernier ayant réussi à établir une connexion à 100 Mbit/s dans un bus se déplaçant à 60 km/h. L'opérateur Japonais *DoCoMo* a par ailleurs réussi à transférer des données à 5 Gbit/s lors du test de son futur réseau 4G, soit 1300 fois plus rapide que le 3.5G lancé au Japon en 2006 [Wikipedia (2008)].

Les premiers téléphones compatibles 4G devraient apparaître d'ici 2009. Les premiers réseaux 4G quand à eux seront déployés d'ici 2010. Les infrastructures du futur 4G se composeront d'un ensemble de divers réseaux utilisant le protocole IP. Ils entoureront tous les systèmes des divers réseaux : publics à privés, larges bandes et réseaux *Ad hoc*. La figure 1.2 illustre un cas d'implémentation d'un réseau 4G.

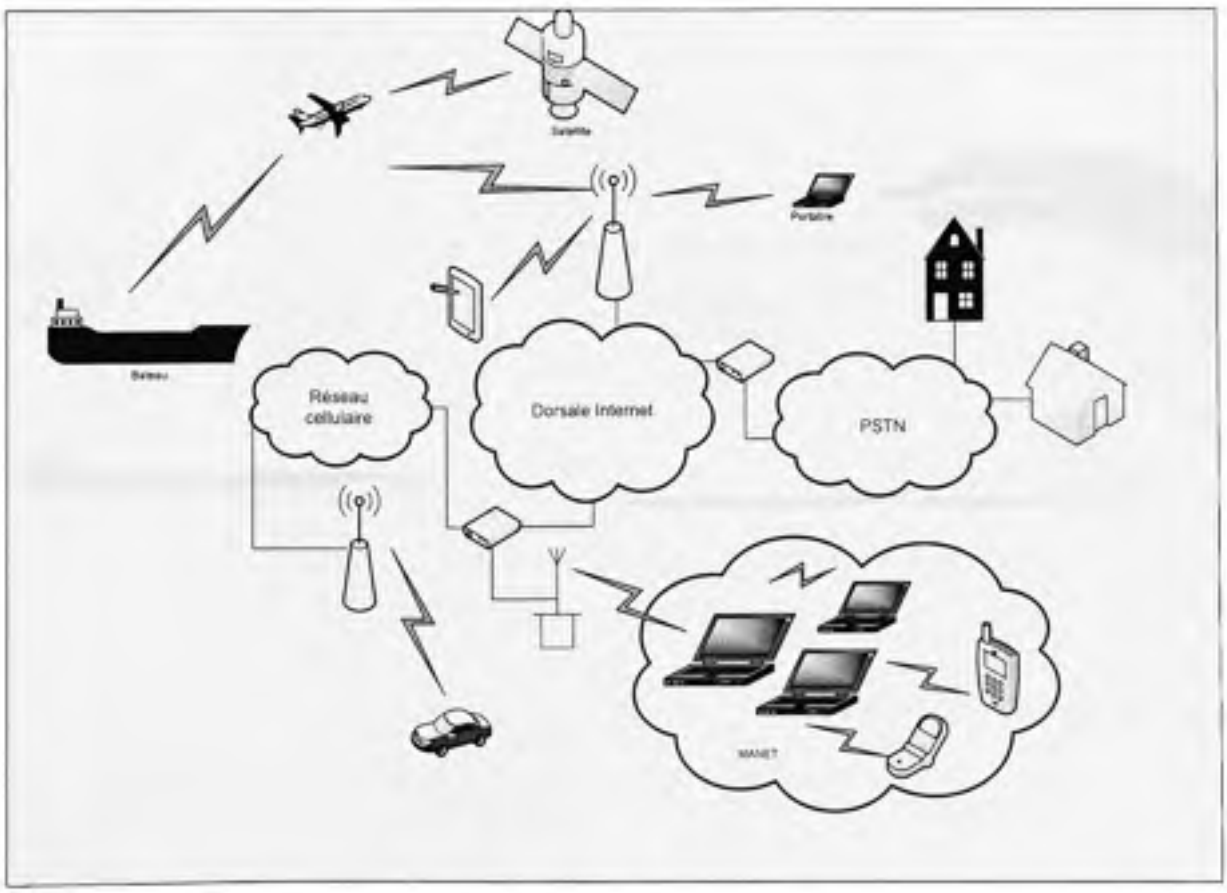

Figure 1.2 ***Exemple d'un réseau 4G.***

Du côté du marché actuel, la majorité des opérateurs mondiaux continue à installer et mettre à jour leurs réseaux pour satisfaire l'évolution rapide des abonnés mobiles sans fil, la demande pour des équipements grimpe rapidement. En 2006, 8 % des ventes et des achats d'équipements visaient des équipements fournissant l'accès radio. En revanche, nous remarquons sur la figure 1.3 une baisse des revenues à partir de l'année 2006. 14 % de cette perte étaient causés par la décroissance du marché des réseaux GSM en faveur du CDMA. Les opérateurs ont déjà commencé à remplacer leur réseau GSM par des réseaux W-CDMA et CDMA2000 en raison des débits importants que ces technologies offrent pour les nouvelles applications (multimédia, vidéo et télévision mobile).

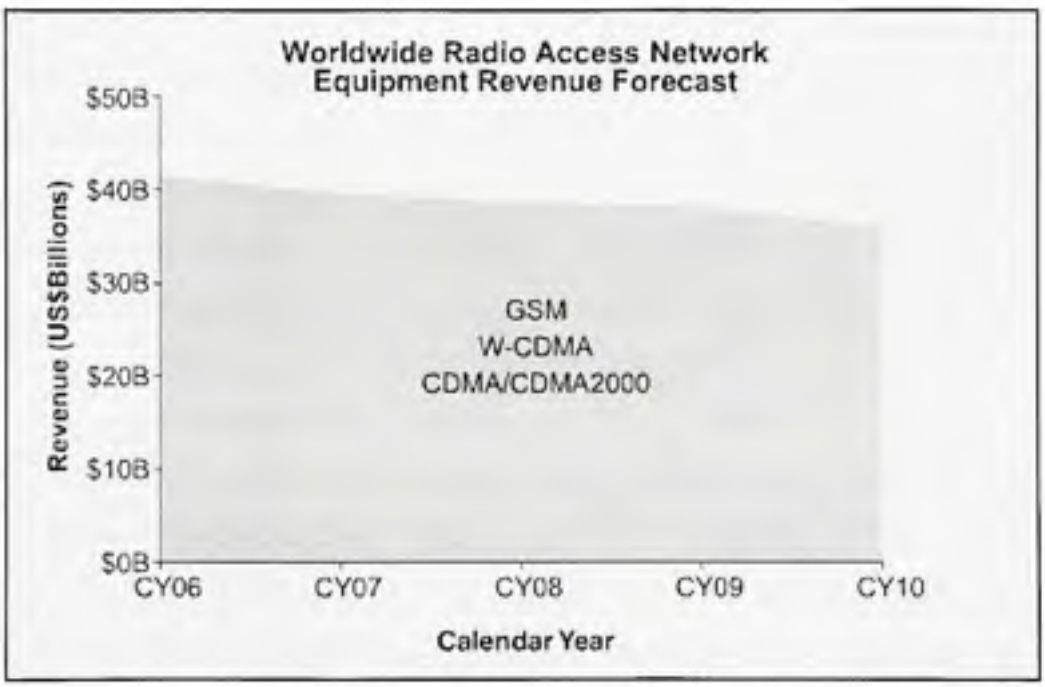

Figure 1.3 ***État du marché des réseaux cellulaires.***

(Tiré de infonetics research, 2008a)

Source : Cette figure a été tirée d'infonetics research 2008 « W-CDMA and CDMA2000 infrastructure markets up, GSM down » et correspond à la « Figure 1 Worldwide Radio Access Network Equipment Revenue Forecast » présentée en page pr/2007/ms07.ran.4q06.nr.asp sur le site web de l'organisme. (La référence complète du document est présentée dans la liste de références).

1.4 Le *Bluetooth*

Initié au départ par *Ericsson, IBM, Intel, Nokia, Toshiba* et plusieurs autres sociétés, en 2002, le BSIG (*Bluetooth Special Interest Group*) a publié les premières spécifications de la technologie *Bluetooth* IEEE 802.15.1. Cette technologie opère dans la bande de fréquences de 2.4 GHz divisée en 79 canaux de 1 MHz (mêmes canaux de IEEE 802.11b), elle avait pour objectif d'éviter le câblage entre les différents dispositifs numériques tels qu'un ordinateur et ses périphériques (clavier, souris, *Walkman*, PDA, imprimante, appareil photo numérique, etc.) dans un rayon de portée assez limitée (PAN) pour un faible coût de mise en place et une faible consommation électrique.

Le groupe de travail IEEE 802.15.2 s'occupait des problèmes de coexistence et d'interférences avec tout autre réseau opérant dans la bande de 2.4 GHz. La portée de transmission varie de quelques mètres à une dizaine de mètres en fonction de la puissance de transmission du terminal. Des mécanismes d'économie d'énergie sont utilisés pour économiser la consommation des batteries et permettre une autonomie prolongée.

Un schéma de communication *Bluetooth* consiste à former plusieurs *piconets* (microcellules). Un *piconet* peut prendre en charge jusqu'à neuf terminaux actifs (255 terminaux en mode veille), dont un est un maître élu qui orchestre les huit autres esclaves situés à un seul saut du maître. Toutes les communications se font toujours dans un mode maître-esclave, le maître d'un *piconet X* peut alors devenir l'esclave du maître d'un autre *piconet Y*. D'autre part, un esclave peut être l'esclave de plusieurs maîtres, il peut se détacher d'un maître pour se raccrocher à un autre *piconet* puis revenir à son premier maître après avoir terminé sa communication avec le second telle qu'illustrée sur la figure 1.4.

L'interconnexion de ces *piconets* permet de construire un réseau multisaut donnant lieu à un *scatternet* (cellules dispersées). Le débit maximal dans un *piconet* dépend du nombre de terminaux actifs, ce débit peut atteindre 1 Mbit/s en utilisant la technique de modulation de

fréquences FSK (*Frequency Shift Keying*). Pour des communications bidirectionnelles entre deux terminaux, ce débit est d'au plus de 434 Kbit/s.

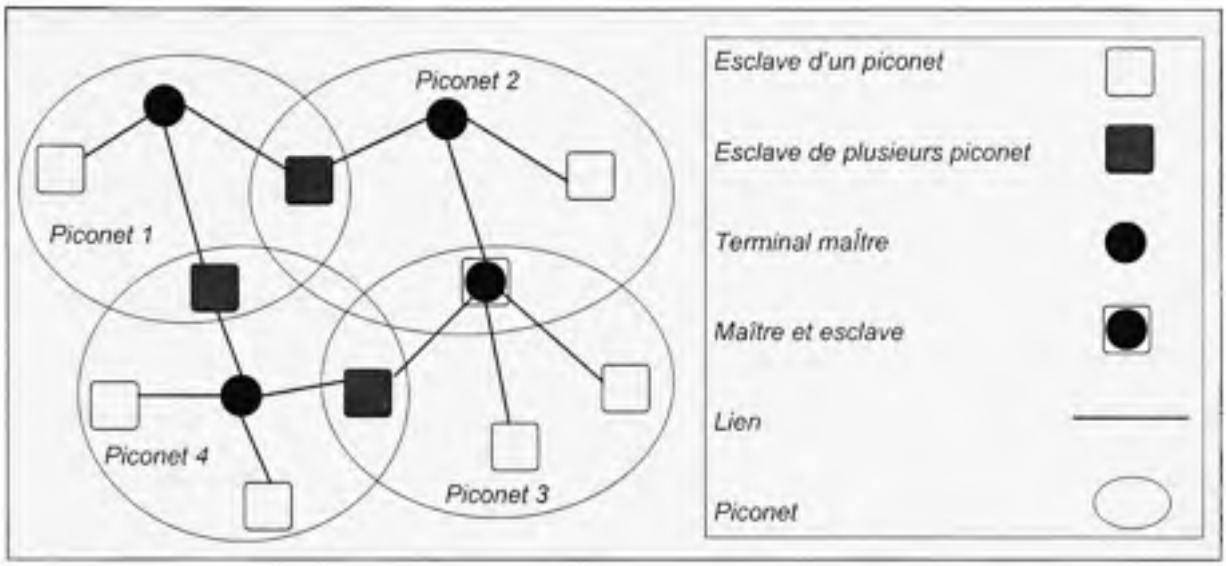

Figure 1.4 ***Schéma de communication scatternet d'une architecture Bluetooth.***

Chaque équipement *Bluetooth* détient une adresse *Bluetooth* unique et une horloge interne. Pour garantir une allocation équitable des canaux, la technique d'étalement de spectre par saut de fréquence FHSS (*Frequency Hopping Spread Spectrum*) est utilisée. La durée de chaque créneau de temps (*slot*) étant de 625 microsecondes, ce qui permet 1600 sauts de fréquence par seconde.

Le maître prend en charge la répartition des fréquences et des temps de maintien de façon que tous les terminaux du même *piconet* adhérent à la même suite de saut de fréquence, et lorsqu'un nouveau terminal veut se connecter, il doit essayer de reconnaître l'ensemble des sauts de fréquence pour pouvoir les respecter. *Bluetooth* offre également divers mécanismes d'authentification et de cryptage au niveau MAC pour renforcer la sécurité de cette architecture face aux intrus environnants.

1.5 La norme IEEE 802.11

Bluetooth, pour sa faible puissance, sa faible portée et son faible débit, peut être vu comme une première initiative de « remplacement de câble », alors qu'à l'opposé, la norme IEEE 802.11 présente des caractéristiques dignes d'une vraie « technologie d'accès ». Dans cette section, nous décrivons l'architecture et les principes de fonctionnalité de cette technologie dans un contexte d'utilisation générale (avec et sans point d'accès).

En 1997, les premières spécifications de la norme IEEE 802.11 ont été publiées. Considéré comme un rival d'Ethernet, IEEE 802.11 est un standard international décrivant les caractéristiques de la couche physique et d'accès au canal d'un réseau local sans fil (WLAN). Il permet de créer des LAN sans fil à haut débit selon deux configurations : avec et sans station de base (point d'accès). Le rayon de couverture varie de dizaines à une centaine de mètres selon la topographie du réseau. Ce qui s'avère important dans des zones à forte concentration d'utilisateurs (*hot spots*) telles que les gares, les aéroports, les hôtels et les trains, où l'installation du câble est très contraignante.

L'élément essentiel de cette architecture est la cellule, aussi appelée ensemble de services de base BSS (*Basic Service Set*). Un BSS est composé de plusieurs terminaux et d'une station de base centrale, appelée point d'accès AP (*Access Point*). Ce dernier se compose habituellement d'un émetteur/récepteur radio, d'une carte réseau filaire (par exemple IEEE 802.3) et d'un logiciel de pontage conforme au standard 802.1d.

La figure 1.5 montre que les terminaux IEEE 802.11 peuvent aussi être regroupés en réseau *Ad hoc*. Ce regroupement également appelé ensemble de services de base indépendant IBSS (*Independent Basic Service Set*) est formé d'un réseau sans point d'accès et sans contrôle centralisé; il permet de créer rapidement un réseau sans fil là où il n'existe pas d'infrastructure filaire, ou encore lorsqu'une telle infrastructure n'est pas nécessaire pour les services attendus (chambre d'hôtel, centre de conférence ou aéroport), ou lorsque l'accès au

réseau filaire est interdit. Au sein de l'IETF, les activités des réseaux *Ad hoc* sont placées sous la responsabilité du groupe de travail MANET.

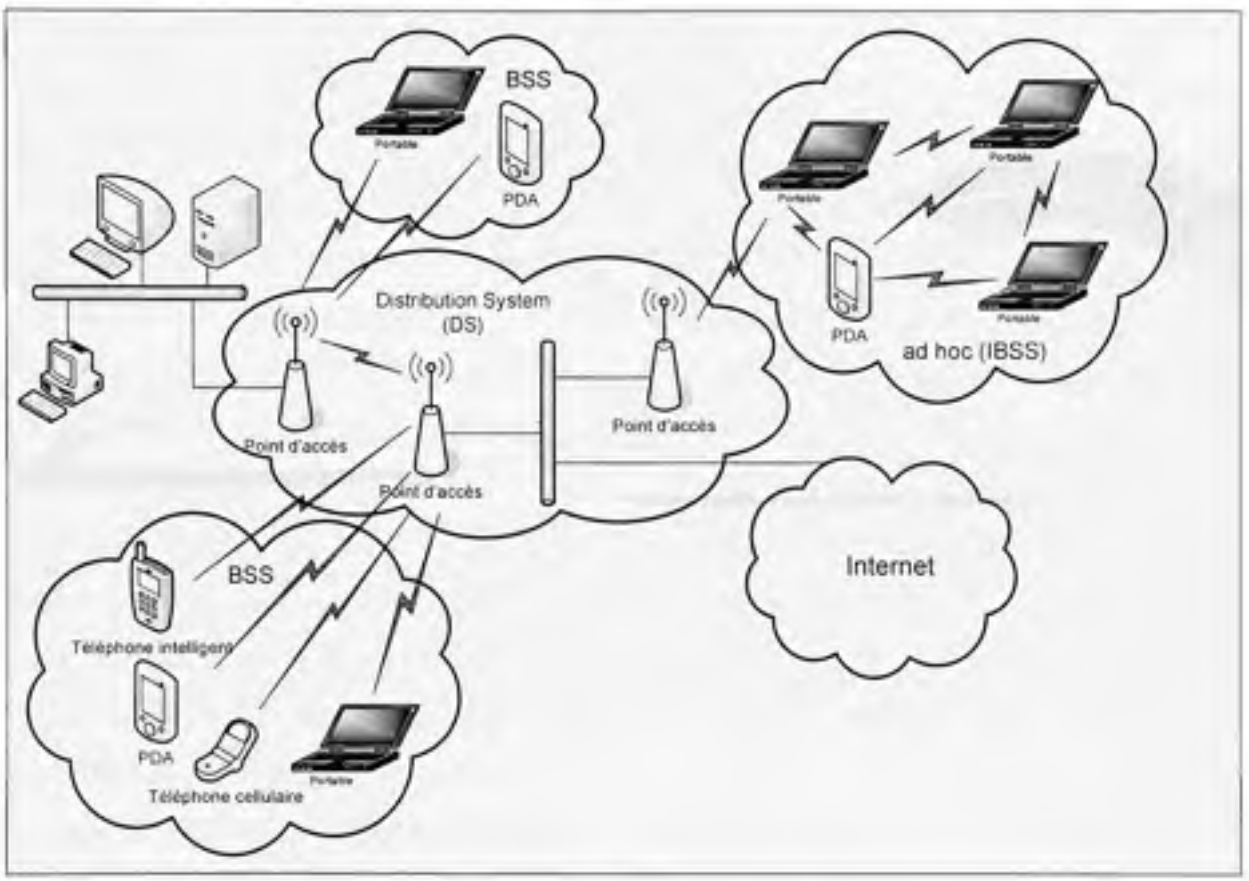

Figure 1.5 ***Architecture d'un réseau 802.11 en modes infrastructure et Ad hoc.***

Toutefois, il est possible de relier plusieurs BSS et/ou IBSS par un système de distribution DS (*Distribution System*) afin de constituer un ensemble de services étendus ESS (*Extended Service Set*). Le système de distribution peut être aussi bien un réseau filaire qu'un réseau sans fil. Un ESS est repéré par un identificateur servant de nom pour le réseau. Ce nom, souvent abrégé en SSID (*Service Set Identification*), représente le nom du réseau où la connaissance du SSID est nécessaire pour qu'un terminal se connecte au réseau étendu. Un terminal est capable de changer de point d'accès selon la qualité de réception des signaux provenant des différents points d'accès. Cette caractéristique d'itinérance permettant aux

terminaux de passer de façon transparente d'un point d'accès à un autre est bien décrite dans le groupe de travail IEEE 802.11r.

En présence d'un point d'accès, un terminal balaye les canaux afin de trouver le meilleur signal provenant d'un AP. Le balayage se fait soit en mode passif où le terminal attend simplement de recevoir une trame balise (*Beacon Frame*) du point d'accès contenant des informations importantes telles que le SSID, les taux de transfert supportés, etc., soit en mode actif où le terminal essaye de trouver un point d'accès en transmettant une trame de requête d'enquête (*Probe Request*) et attend la réponse des points d'accès disponibles. Un terminal se trouvant à la portée de plusieurs points d'accès peut choisir le point d'accès offrant le meilleur compromis de débit et de charge.

Après la détection d'un AP, un terminal 802.11 doit s'associer à ce dernier avant d'envoyer des trames de données. Il initialise l'association en envoyant une trame de demande d'association contenant des éléments tels que le SSID et le débit supporté. Le point d'accès répond en envoyant une réponse d'association contenant entre autres un identificateur d'association. C'est seulement une fois que le processus d'association a été accompli que la station et le point d'accès peuvent échanger les trames de données.

La sécurité est le premier souci de ceux qui déploient les réseaux locaux sans fil. Le comité IEEE 802.11 a apporté des solutions en élaborant des processus optionnels à implémenter comme WEP (*Wired Equivalent Privacy)* basé sur les algorithmes de cryptage RC4 (*Ron's Code 4*) et RSA (*Rivest, Shamir, Adleman*). Son inconvénient est qu'il est possible de casser les clés de chiffrement sans trop de difficulté. Un groupe spécifique IEEE 802.11i propose une solution prometteuse, normalisée en 2004, pour combler les lacunes de WEP. Des protocoles comme WPA (*Wifi Protected Access*), WPA2, EAP (*Extensible Authentification Protocol*) permettront de renforcer la sécurité dans ces réseaux.

D'autre part, l'énergie de la batterie des terminaux est une ressource importante. C'est pour cette raison que le standard IEEE 802.11 définit tout un mécanisme d'économie d'énergie

facultative pour mettre en veille les terminaux pendant de longues périodes sans perdre d'informations. Ce mode est signalé au point d'accès de façon à lui permettre de maintenir les paquets adressés aux terminaux travaillant en mode d'économie d'énergie. Le point d'accès transmet périodiquement des informations spécifiant les terminaux qui ont des paquets stockés par lui. Dans le cas échéant, ces terminaux peuvent se réveiller pour récupérer ces paquets.

Pour contrôler l'accès au medium radio, un réseau IEEE 802.11 avec point d'accès utilise à la fois deux méthodes d'accès : DCF (*Distributed Coordination Function*) et PCF (*Point Coordination Function*). Seule la méthode DCF peut être utilisée dans un réseau en mode *Ad hoc*. Cette méthode se base sur la technique d'écoute du canal avec évitement de collision CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*) dont certains de ces aspects avancés seront détaillés dans la section 2.3.2.3 du chapitre 2.

Quant à elle, la méthode PCF est seulement employée en présence d'un point d'accès, PCF permet de fournir des services temps réel, comme la transmission de la voix ou de la vidéo. Dans ce cas, le temps est toujours divisé en périodes répétées, appelées les supertrames. Une supertrame est composée d'une période sans contention CFP (*Contention Free Period*) et d'une période avec contention CP (*Contention Period*). Pendant CFP, le PCF est utilisé pour accéder au canal, alors que le DCF est utilisé pendant CP. Dans un CFP, il n'y a pas de compétition entre les terminaux pour accéder au canal, car c'est le AP qui contrôle l'accès au canal, il invite les terminaux à transmettre à tour de rôle. L'AP continue à inviter d'autres stations jusqu'à ce que le CFP expire.

Le groupe de travail IEEE 802.11e a défini des améliorations sur la couche MAC 802.11. Il propose les nouvelles fonctions EDCF (*Enhanced DCF*) et HCF (*Hybrid Coordination Function*). Avec IEEE 802.11e, les deux phases de l'opération sont toujours présentes dans les supertrames, un CFP et un CP, qui alternent avec le temps sans interruption. EDCF est utilisé seulement dans le CP, alors que HCF est utilisé dans les deux phases CP et CFP. EDCF permet de faire un contrôle local de priorité au niveau de chaque terminal en

modifiant le mécanisme d'accès de base DCF. Un simple terminal peut mettre en application plusieurs files d'attente de transmission décrites dans quatre catégories de trafic TC (*Traffic Category*). Ces files sont réalisées en tant que terminaux virtuels à l'intérieur du même terminal, avec les paramètres de la qualité de service qui déterminent leurs priorités.

HCF est aussi vu comme une variante de PCF; il est utilisé par un point d'accès pendant la période d'accès contrôlé CAP (*Controlled Access Period*), qui peut commencer n'importe quand dans la supertrame. Ceci permettra de prioriser l'accès d'un flux urgent au medium à n'importe quel moment dans le BSS.

Du côté routage, un terminal faisant partie d'un BSS et équipé d'une carte sans fil conformément à la norme 802.11. Il communique avec son point d'accès au moyen du protocole MAC IEEE 802.11 en utilisant une des méthodes décrites ci-haut (DCF, PCF, EDCF, HCF) pour contrôler l'accès au medium sans fil. Les mêmes protocoles de routage classiques tels que RIP (*Routing Interior Procotol*), OSPF (*Open Shortest Path First*), etc. peuvent être implémentés sur les points d'accès pour acheminer les paquets entre les différents BSS.

Toutefois, dans un mode *Ad hoc*, où il n'existe plus d'entités centralisées (points d'accès), il faut de nouveaux protocoles de routage pour acheminer les paquets entre des terminaux qui sont eux-mêmes mobiles. Parmi ces protocoles, nous citons les familles des protocoles proactifs comme DSDV (*Destination-Sequenced Distance Vector protocol*) et OLSR (*Optimized Link State Routing protocol*), réactifs comme DSR (*Dynamic Source Routing*) et AODV (*Ad hoc On-demand Distance Vector*) et hybrides comme ZRP (*Zone Routing Protocol*) et LANMAR (*Landmark Ad hoc Routing*). La section 2.3.3 du chapitre 2 est consacrée pour élaborer ces protocoles dans un contexte de réseaux *Ad hoc*.

Du côté marché, les statistiques illustrées sur la figure 1.6 montrent des chiffres très avantageux concernant l'émergence des technologies WLAN 802.11. Actuellement, nombreuses sont les compagnies qui ont adopté les WLAN. Les revenues de ventes des

équipements WLAN continueraient à grimper rapidement. Les statistiques montrent qu'elles atteindraient 5 milliards $ US d'ici l'an 2010.

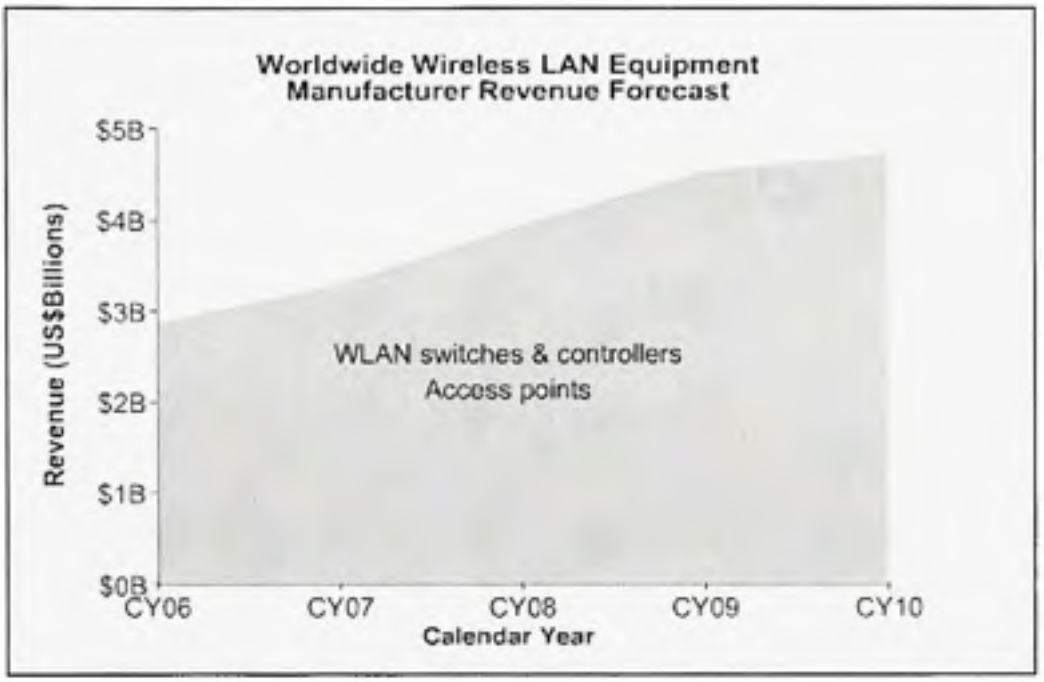

Figure 1.6 ***Croissance du marché de la technologie WLAN.***

(Tiré de infonetics research, 2008b)

Source : Cette figure a été tirée d'infonetics research 2008 « Wireless LAN equipment market up 5% in 1Q07 » et correspond à la « Figure 1 Worldwide Wireless LAN Manufacturer Revenue Forecast » présentée en page pr/2007/ms07.wl.1q07.nr.asp sur le site web de l'organisme. (La référence complète du document est présentée dans la liste de références).

1.6 Conclusion

Dans ce chapitre, nous avons survolé rapidement l'état d'avancement de différentes technologies sans fil actuellement déployées. Nous avons également évoqué les réseaux de la quatrième génération qui seront disponibles au cours des prochaines années. Une présentation succincte des réseaux locaux sans fil nous a permis d'introduire le thème de recherche que nous visons dans cette dissertation.

Dans le chapitre suivant, nous ferons une étude complète et détaillée de la technologie visée dans nos travaux de recherche.

CHAPITRE 2

LES RÉSEAUX SANS FIL *AD HOC*

2.1 Introduction

L'accès à Internet atteint de plus en plus les terminaux sans fil, enrichissant ainsi notre vie quotidienne de plusieurs applications. Une large bande de standards accompagne cette progression pour supporter les technologies d'accès, nous donnant la liberté de nous déplacer, tout en restant connectés, et ouvrant la voie à plus d'applications telles que les courriers électroniques, les navigateurs, l'audio et la vidéo aux terminaux sans fil. Toutefois, ce développement pose de nouveaux défis en termes de gestion de mobilité, d'interopérabilités entre systèmes hétérogènes, de qualité de service, ainsi que d'allocation des bandes de fréquences adéquates par des organismes dédiés.

Parmi ces standards émergents, nous nous intéressons aux réseaux sans fil *Ad hoc* largement élaborés dans des travaux de grand intérêt comme ceux de Perkins (2001), Toh (2001) et Murthy et Manog (2004). Grâce à leur flexibilité d'utilisation, ces réseaux ont fortement attiré l'attention des chercheurs et des industriels, qui nous préparent une ère prometteuse avec des champs d'applications illimités.

Ce type de réseaux peut être déployé dans tout environnement difficile où le déploiement d'une infrastructure réseau filaire est très contraignant, soit parce que difficile à mettre en place, soit parce que la durée d'installation du réseau ne justifie pas de câblage à demeure. Il s'agit ici d'un réseau de paquets radio à multisauts où les données se propagent de mobile à mobile sans nécessiter des points d'accès fixes.

Un tel réseau dépourvu d'équipement de cœur trouve de nombreux champs d'applications : lors d'une compagne militaire, d'une situation de catastrophe (tremblement de terre,

désastre naturel, etc), d'une urgence, d'une salle de conférence, ou encore dans le cadre des réseaux de senseurs/capteurs (*sensor networks*).

2.2 Les caractéristiques des réseaux *Ad hoc*

Un réseau *Ad hoc* est caractérisé par des propriétés particulières. Chaque propriété est considérée dans la littérature comme étant une problématique en soi. Dans cette section, nous allons décortiquer ces propriétés dans l'objectif de définir un langage commun permettant de nous familiariser avec les particularités de ces réseaux.

- **absence d'infrastructure et nature de communication multisaut :** un réseau *Ad hoc* se distingue des autres réseaux mobiles par l'absence d'infrastructure dédiée et de tout genre d'administration centralisée. Les fonctionnalités réseau (routage, localisation, etc.) seront à la charge des terminaux mobiles, introduisant la notion de nœud-routeur pour tout terminal. Chaque terminal *Ad hoc* doit être capable de maintenir une vue partielle du réseau qui lui permettra de transmettre ses informations, en sachant que le destinataire potentiel peut se trouver à plusieurs sauts, et donc qu'il doit obligatoirement passer par d'autres nœuds tels que lui. Cette participation collaborative entre les terminaux illustrée sur la figure 2.1 conduit à établir et à maintenir une connectivité continue du réseau;
- **topologie dynamique et décentralisée :** les nœuds *Ad hoc* ont la liberté de se déplacer aléatoirement dans une direction quelconque provoquant des problèmes de routage et de coupures des liens entre les paires de nœuds. Ces liens inter-nœuds étant de nature unidirectionnelle ou bidirectionnelle sont caractérisés par un degré d'instabilité très haut causant des variations intempestives de la qualité des connexions. Ceci entraîne des changements imprévisibles dans la topologie d'une façon rapide et brusque, nécessitant ainsi une adaptation dynamique et décentralisée des capacités du réseau afin de pouvoir garantir une connectivité permanente;
- **bande passante limitée et débit variable des liens :** l'utilisation des méthodes de partage du canal radio (accès multiple) influence directement la bande passante réservée à un terminal *Ad hoc*. De plus, le terminal n'atteindra jamais le débit maximal alloué

dans la largeur de bande radio. Ceci est dû aux différents phénomènes de perturbations caractérisant un signal électromagnétique transmis sur un canal radio. Entre autres, nous citons les phénomènes d'atténuation (*shadowing*) causée par les obstacles, de distorsion causée par le déphasage du signal, et d'évanouissements causés par la propagation multi-trajets (*multipath fading*);

- **contraintes d'énergie et de capacité de calcul (CPU et quantité mémoire) :** les terminaux étant alimentés par des sources d'énergie limitées et rarement disponibles, le facteur de consommation d'énergie devient primordial et doit être pris en considération dans tout contrôle fait par le système afin de ne pas épuiser trop rapidement les batteries des terminaux;
- **sécurité physique limitée :** la nature ouverte du canal radio facilite l'écoute des communications dans un réseau *Ad hoc*. Un intrus malveillant pourrait tout simplement intercepter le trafic et le détourner pour faire tout type d'attaques que nous pouvons imaginer. Nombreux sont les axes de recherches qui tentent de renforcer l'aspect de sécurité dans ces réseaux vulnérables. En effet, la sécurité est nécessaire à la démocratisation de tels réseaux.

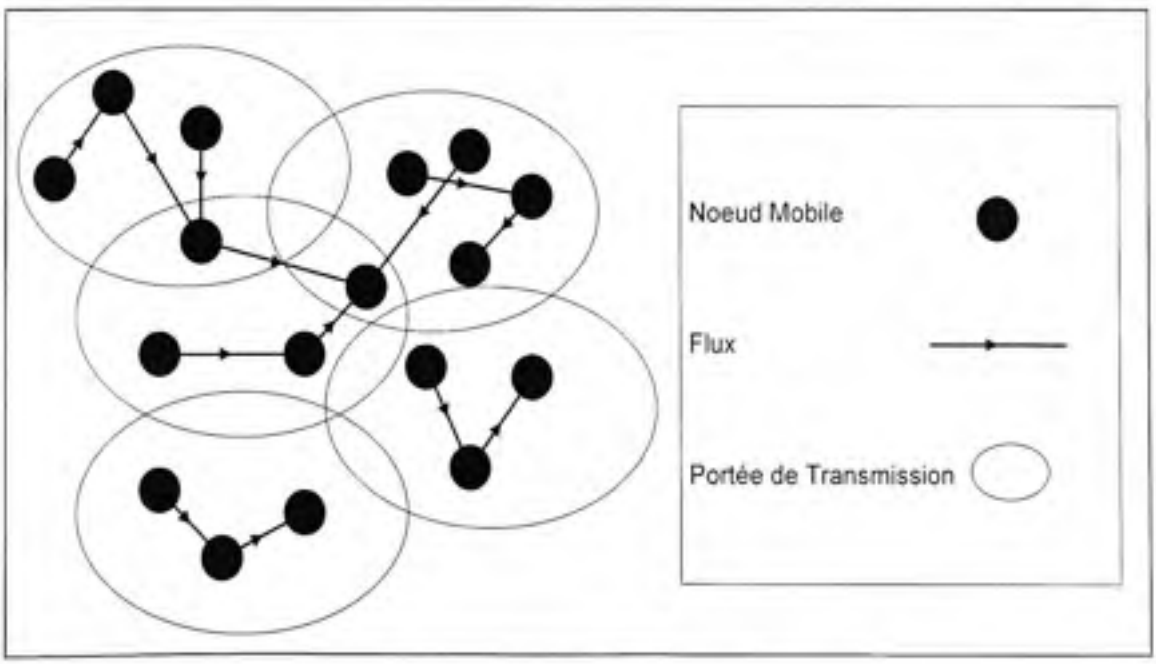

Figure 2.1 ***Exemple d'un réseau Ad hoc.***

2.3 Les couches des réseaux *Ad hoc*

Après avoir défini les propriétés de ce type des réseaux, nous procédons dans cette section à la présentation de différentes spécificités et particularités des couches du modèle de communication dans les réseaux sans fil *Ad hoc*.

2.3.1 La couche physique

2.3.1.1 Introduction

Il n'est pas question ici d'effectuer une étude approfondie de la couche physique. Cependant, il est intéressant de présenter succinctement les principales techniques adoptées par la norme IEEE 802.11 pour augmenter les débits tout en limitant les problèmes dus aux interférences. Toutefois, nous préférons commencer par la description des caractéristiques du canal radio qui dépendent en grande partie des conditions de propagation et de la distance qu'un signal doit parcourir. Ces conditions sont notamment touchées par ce qui suit :

- phénomène d'affaiblissement de trajet (*path loss*) et l'affaiblissement par diffraction sur les obstacles (*shadow fading*), qui ont pour effet d'atténuer le signal en fonction de la distance et des obstacles rencontrés en chemin;
- phénomène d'évanouissements par propagation multi-trajets (*mutlipath fading*) causés par la réflexion du signal au contact d'objets interférents;
- phénomène d'interférences liées à l'existence d'autres canaux radio ou d'autres signaux électromagnétiques.

De plus, la puissance du signal reçu, le SNR (*Signal to Noise Ratio*) et le BER (*Bit Error Rate*) nous donnent de bonnes indications sur la qualité du lien radio. Le choix des techniques de modulation et de codage canal impactent considérablement sur le débit atteignable pour un niveau de bruit donné.

2.3.1.2 Techniques de modulation et bande de fréquences utilisées

D'une façon générale, la couche physique définit la modulation des ondes électromagnétiques. La plupart des pays ont alloué certaines bandes de fréquences, appelées ISM (*Industriel, Science, Medical*), pour une exploitation non soumise à licence (Exemple : porte de garage et jouets télécommandés, etc.), permettant à tout le monde d'émettre à volonté, mais en contrôlant la puissance de transmission de façon que les émetteurs aient une portée limitée pour qu'il n'y ait pas d'interférences entre eux.

Aux États-Unis, le FCC (*Federal Communications Commission*) exige l'utilisation des techniques d'étalement de spectre dans ces bandes de fréquences. Les bandes utilisées varient d'un pays à autre, celles les plus prisées sont les bandes de 2.4 GHz et 5 GHz.

Le premier standard IEEE 802.11 fournit des débits allant de 1 à 2 Mbit/s dans la bande de fréquence ISM de 2.4 GHz en utilisant une technique DSSS. Il utilise la modulation de phase PSK (*Phase Shift Keying*) à 1 Mbaud, transmettant 1 bit par baud pour un débit de 1 Mbit/s et 2 bits par baud pour un débit de 2 Mbit/s.

La version IEEE 802.11b qui utilise également DSSS dans la bande ISM, fournit des débits allant jusqu'à 11 Mbit/s dans un milieu ouvert. La version IEEE 802.11a offre des débits allant jusqu'à 54 Mbit/s; elle utilise le multiplexage en fréquences orthogonales OFDM (*Orthogonal Frequency Division Multiplex*) opérant dans la bande de fréquences libres UNII (*Unlicensed National Information Infrastructure*) de 5 GHz. IEEE 802.11a est incompatible avec IEEE 802.11b.

La norme IEEE 802.11g est la plus répandue actuellement. Elle pioche à la fois dans les extensions 802.11a et 802.11b. Cette approche est très naturelle lorsqu'on pense à prolonger les produits IEEE 802.11b en assurant une compatibilité descendante des produits IEEE 802.11g vers des produits actuels IEEE 802.11b. IEEE 802.11g utilise OFDM ce qui permet

de récupérer les investissements consentis sur la norme IEEE 802.11a et un débit allant jusqu'à 54 Mbit/s.

La norme IEEE 802.11n, récemment standardisée, offre de très hauts débits (100 à 320 Mbit/s) sur les bandes de 2.4 et 5 GHz en utilisant des techniques MIMO (*Multiple Input Multiple Output*) qui se basent sur un système à multiples antennes pour l'émission et la réception simultanée des signaux.

Le standard IEEE 802.11 a également prévu l'utilisation de la lumière infrarouge, qui consiste à utiliser une onde lumineuse pour la transmission de données. Les transmissions sont unidirectionnelles, elles se font soit en vue directe soit par réflexion. Il est possible grâce à cette technologie d'obtenir des débits modérés allant de 1 à 2 Mbit/s en utilisant une technique de modulation appelée PPM (*Pulse Position Modulation*). Cette technique n'est pas très populaire en raison de la faible bande passante qu'elle offre et de la portée très limitée des ondes infrarouges.

2.3.2 La couche liaison de données

2.3.2.1 Introduction

Une des problématiques des réseaux *Ad hoc* est le partage efficace du canal radio entre les différents participants actifs. Aujourd'hui, IEEE 802.11 est vu en tant qu'un réseau Ethernet sans fil en vertu de son service 'best effort' basé sur le protocole de contrôle d'accès MAC similaire à celui d'Ethernet, mais il n'offre aucune notion de QoS.

La couche liaison de données est composée de deux sous-couches : Le contrôle de la liaison logique LLC (*Logical Link Control*) et le contrôle d'accès au médium MAC (*Media Access Control*). Le protocole IEEE 802.11 couvre les couches MAC et physique. Il utilise la sous couche LLC définie dans IEEE 802.2 et l'adressage sur 48 bits, tout comme les autres LAN 802, simplifiant ainsi le pontage entre les réseaux sans fil et les réseaux filaires.

2.3.2.2 Particularités de la couche MAC dans les réseaux *Ad hoc*

Dans un réseau *Ad hoc*, le signal s'atténue radicalement avec la distance parcourue, ce qui empêche l'utilisation d'un mécanisme de détection des collisions durant l'émission à l'instar du mécanisme CSMA/CD (*Carrier Sense Multiple Access with Collision Detection*) utilisé dans les réseaux Ethernet classiques. Par conséquent, d'autres mécanismes basés sur l'évitement de collision semblent plus acceptables dans un contexte *Ad hoc*.

La solution qui a été adoptée pour ce type de réseau est le CSMA/CA qui se base sur le protocole « arrêt et attente » (*stop and wait*) où l'émetteur ne doit pas transmettre la trame courante tant qu'il n'a pas reçu l'acquittement correspondant à la réception de la trame précédente. Cette méthode sera abordée en détail dans la section 2.3.2.3 et s'applique à tout type de couche physique sous jacente. Cependant, les paramètres de la couche MAC diffèrent d'une couche physique à autre.

Dans un réseau sans fil comme IEEE 802.11, le canal physique radio (la bande de fréquences) peut être divisé en plusieurs canaux logiques. Plusieurs mécanismes, datés dès la première génération des réseaux cellulaires peuvent être utilisés. Ces mécanismes ont été développés au fur et à mesure de l'évolution technologique et suite aux besoins des opérateurs en termes de bande passante, d'immunité aux interférences et de robustesse.

La figure 2.2 illustre le cas du multiplexage fréquentiel FDMA (*Frequency Division Multiple Access*) qui consiste à diviser toute la largeur de bande de fréquences en plusieurs canaux (fréquences) ayant des largeurs de bande égales, séparés par des canaux de garde très étroits afin d'éviter toute interférence inter-canaux. Chaque utilisateur utilise un canal spécifique pendant toute la durée de sa transmission.

La figure 2.3 illustre le cas du multiplexage temporel TDMA (*Time Division Multiple Access*) qui suit le même mécanisme utilisé dans le multiplexage fréquentiel FDMA; mais au lieu de découper la bande en fréquences distinctes, elle est plutôt divisée en tranches de

temps (*slots*), lesquelles sont affectées régulièrement à chaque canal logique. Chaque utilisateur utilise la totalité de la bande passante pendant une tranche de temps.

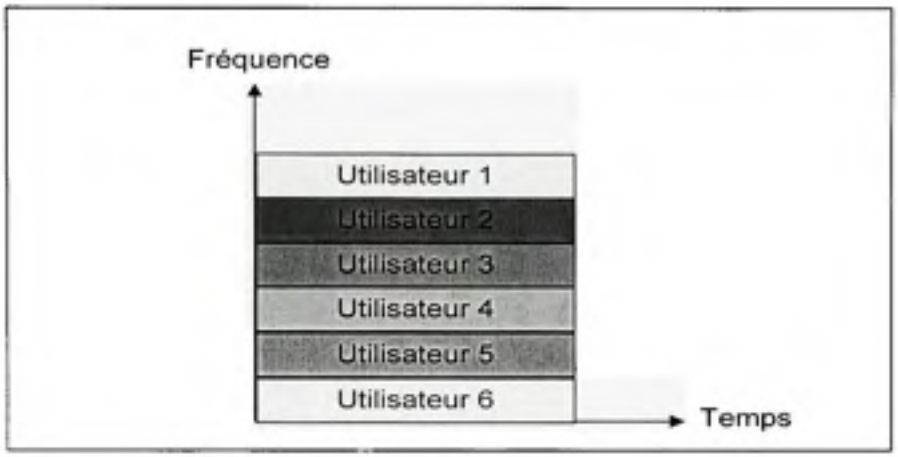

Figure 2.2 ***Technique de multiplexage FDMA.***

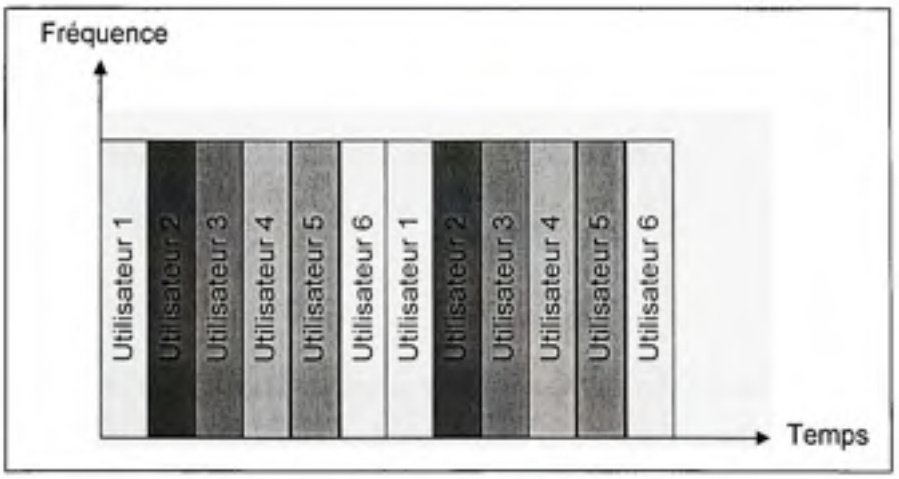

Figure 2.3 ***Technique de multiplexage TDMA.***

Que ce soit FDMA ou TDMA, la bande passante allouée à chaque canal logique est réduite. Elle est égale à la bande passante totale divisée par le nombre de canaux logiques. La figure 2.4 illustre un multiplexage mixte FDMA/TDMA qui alloue une fréquence distincte à chaque utilisateur (FDMA) parmi toute la bande de fréquence disponible, mais pendant une tranche de temps (TDMA).

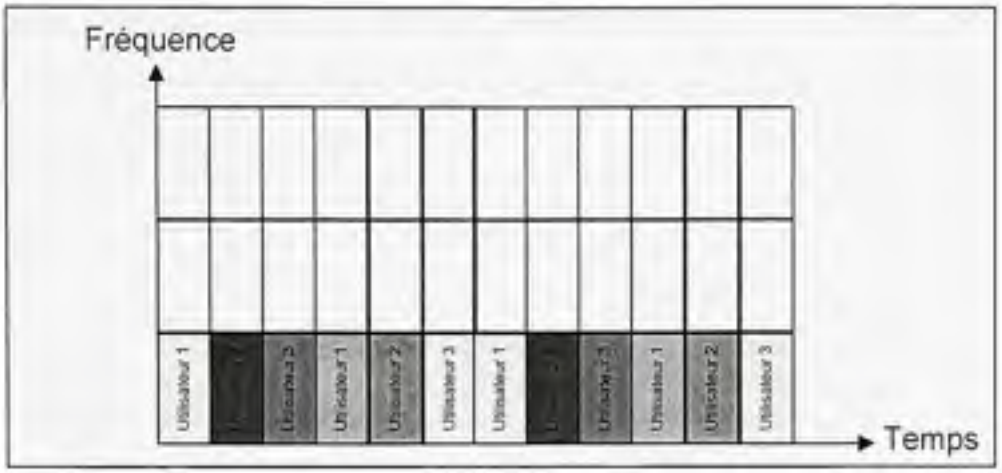

Figure 2.4 ***Technique de multiplexage FDMA/TDMA.***

Cependant, dans les réseaux sans fil modernes, il existe des mécanismes qui permettent de réduire les interférences et les perturbations qui peuvent arriver sur un signal transmis. L'idée est d'étaler le signal sur tout le spectre fréquentiel disponible de façon à ce qu'il apparaisse comme un bruit informe. La figure 2.5 illustre le protocole d'accès multiple par répartition de code CDMA et la technique d'étalement de spectre à séquence directe DSSS qui autorisent chaque utilisateur à émettre sur la totalité du spectre au lieu de répartir la plage de fréquences autorisées en quelques canaux étroits (un canal étroit par utilisateur) comme c'était le cas dans TDMA et FDMA.

Par conséquent, plusieurs transmissions simultanées sont divisées au moyen de techniques de codage. Le principe consiste à transmettre pour chaque bit d'information une séquence binaire de *chips* souvent générée de façon pseudo-aléatoire PN (*Pseudo-random Noise*). Chaque bit valant 1 est remplacé par la séquence de bits et chaque bit valant 0 est remplacé par le complément de cette séquence. Ainsi, le codage requiert que la bande passante disponible soit multipliée par le nombre de bits dans le code utilisé.

Chaque utilisateur doit être identifié de façon unique par un code, il sera capable d'extraire le signal souhaité tout en rejetant le reste comme étant du bruit parasitaire. Cette technique est utilisée en raison de sa grande efficacité spectrale et de son immunité contre les bruits.

Les transmissions d'un utilisateur quelconque ne sont pas affectées par celles d'autres utilisateurs, pourvu que tous les codes alloués aux différents utilisateurs dans le même rayon de portée, soient quasi orthogonaux.

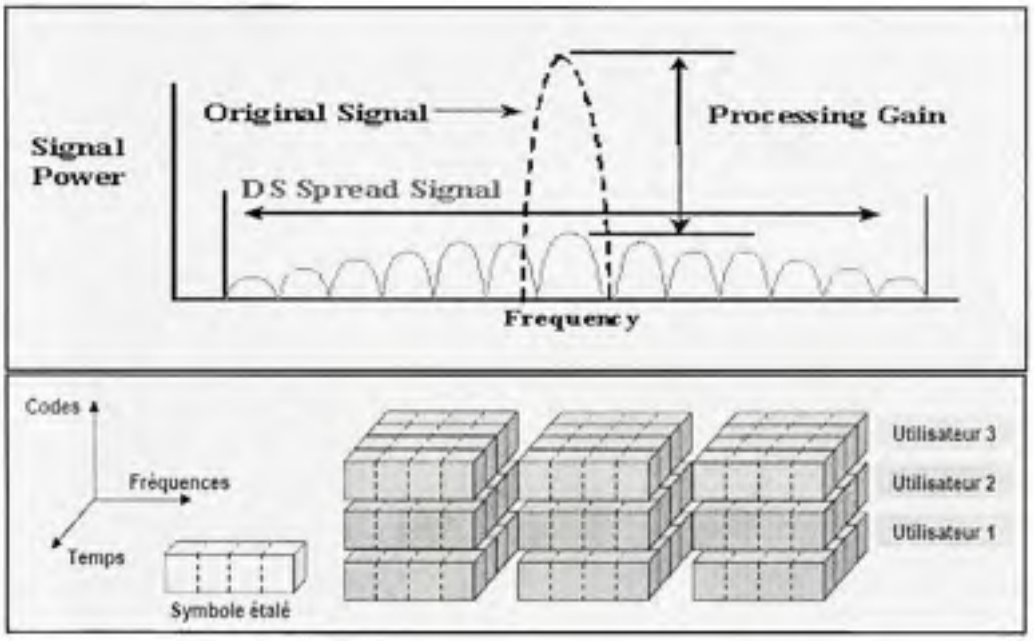

Figure 2.5 ***Technique de multiplexage CDMA/DSSS.***

(Tiré de Lavoie, 2007)

Source : Cette figure a été tirée de Lavoie 2007 « Notes du cours LOG-610 du programme de baccalauréat en génie logiciel et des TI » et correspond à la « Figure 1 Étalement spectral - DSSS » présentée en acétate 41 dans le document original. (La référence complète du document est présentée dans la liste de références).

La figure 2.6 illustre la technique de multiplexage orthogonal en répartition de fréquence OFDM qui consiste à diviser la bande de fréquences 5 GHz en des sous canaux de fréquence plus faible pour offrir un débit de 54 Mbit/s. Un utilisateur étale sa transmission sur différentes bandes de fréquences orthogonales couvrant toute la largeur de bande disponible pendant un certain temps. Ce qui amoindrit l'effet des perturbations et le taux d'erreur sur les bits BER.

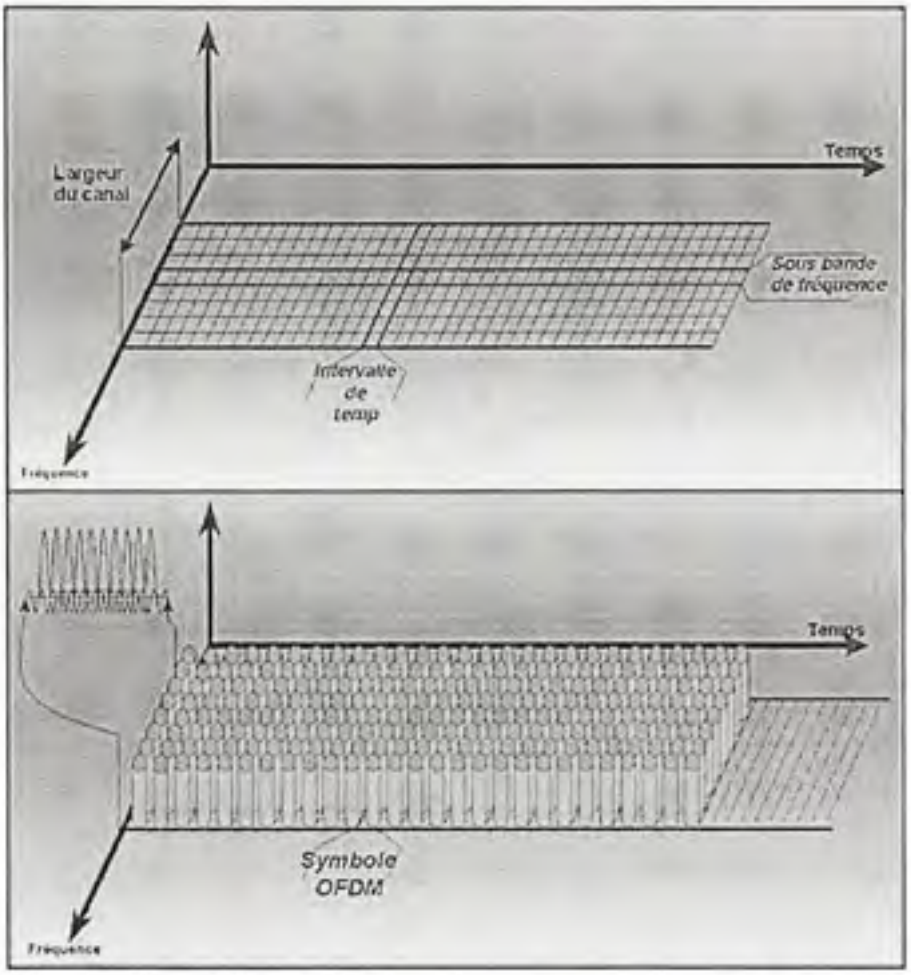

Figure 2.6 ***Technique de multiplexage OFDM.***

(Tiré de Lavoie, 2007)

Source : Cette figure a été tirée de Lavoie 2007 « Notes du cours LOG-610 du programme de baccalauréat en génie logiciel et des TI » et correspond à la « Figure 1 couche physique - OFDM » présentée en acétate 43 dans le document original. (La référence complète du document est présentée dans la liste de références).

La figure 2.7 illustre la technique d'étalement de spectre par saut de fréquence FHSS qui consiste à découper la large bande de fréquence de 2.4 à 2.4835 GHz en 79 canaux de 1 MHz; puis de transmettre en utilisant une combinaison de canaux connue de toutes les stations de la cellule BSS. Une station commence à émettre successivement sur un canal puis sur un autre pendant un laps de temps très court (d'environ 400 ms), ce qui permet une allocation équitable du spectre et améliore le niveau de sécurité puisque l'écoute clandestine est impossible sans connaitre la séquence de canaux. FHSS résiste bien aux phénomènes

d'atténuation et d'évanouissements dus à la propagation multi-trajets (*multiple fading*) sur de longues distances et offre une bonne immunité contre les interférences radio. Toutefois, cette technique souffre d'un faible débit.

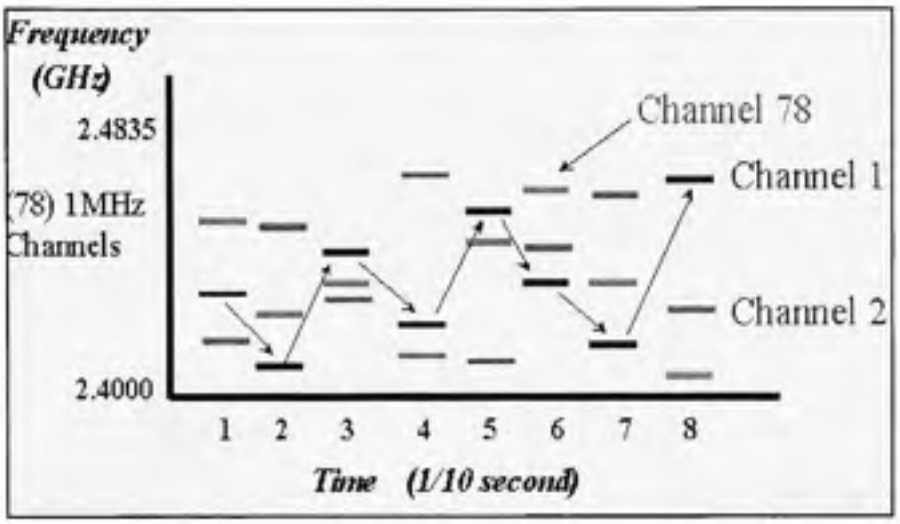

Figure 2.7 ***Technique de multiplexage FHSS.***

(Tiré de Lavoie, 2007)

Source : Cette figure a été tirée de Lavoie 2007 « Notes du cours LOG-610 du programme de baccalauréat en génie logiciel et des TI » et correspond à la « Figure 1 Étalement spectral – FHSS » présentée en acétate 40 dans le document original. (La référence complète du document est présentée dans la liste de références).

La figure 2.8 illustre le cas d'une technique récemment développée et implémentée dans la norme IEEE 802.11n, celle nommée MIMO. Cette technique se base sur un multiplexage par division spatiale SDM (*Spatial Division Multiplexing*) qui permet à une station de multiplexer spatialement de multiples flux de données indépendants et de les transmettre simultanément sur la même largeur de bande de fréquences [Ke *et al.* (2007)]. Chaque flux de données requiert une antenne *Tx/Rx* sur chaque extrémité de la communication (au niveau de la source et au niveau de la destination). Ce qui permet d'atteindre des débits théoriques énormes de l'ordre de 320 Mbit/s.

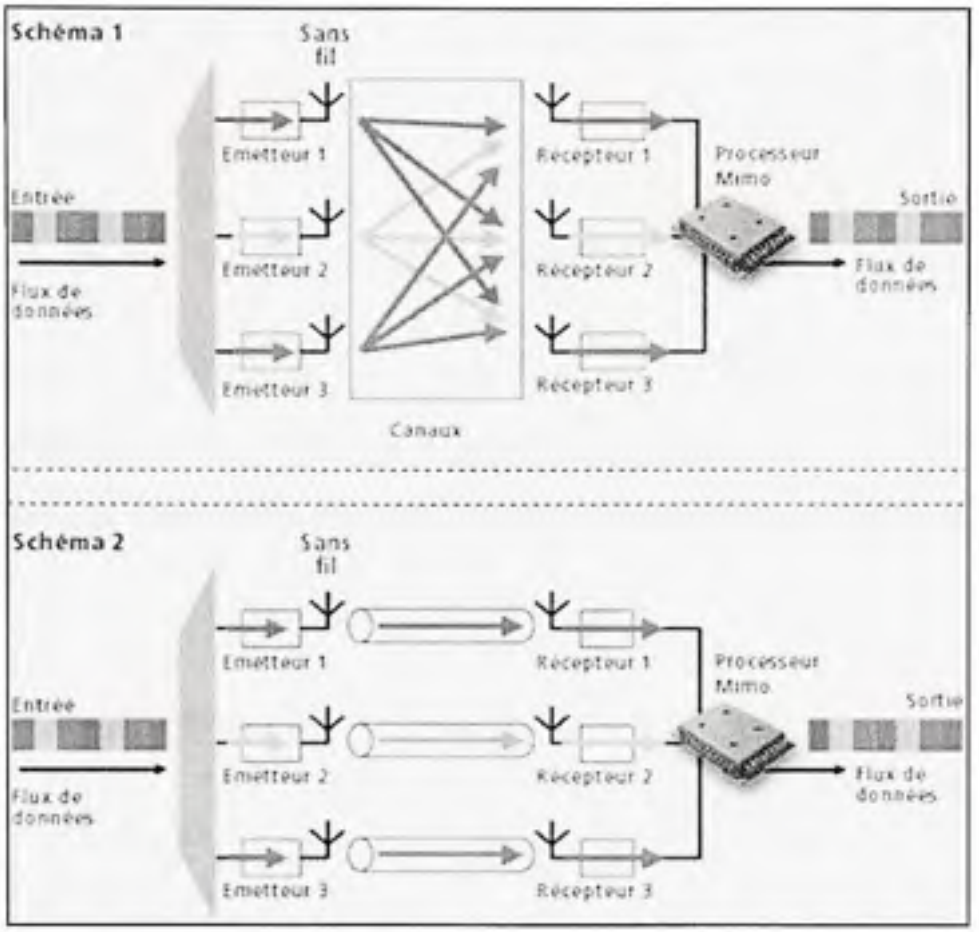

Figure 2.8 ***Technique de multiplexage MIMO.***

(Tiré de 01net, 2008)

Source : Cette figure a été tirée de 01net. 2008 « Mimo, la technique expérimentale qui accélère le sans-fil » et correspond à la « Figure 1 Le principe de la technique MIMO » présentée en page article/212443.html sur le site web de l'organisme. (La référence complète du document est présentée dans la liste de références).

2.3.2.3 Protocoles de la couche MAC pour les réseaux *Ad hoc*

Sur chacun des canaux logiques décrits auparavant, plusieurs problèmes peuvent survenir. En effet, une contention est possible entre les différentes stations pour avoir accès à ce canal logique. Ainsi, chaque station doit utiliser son protocole CSMA/CA pour contrôler l'accès à ce canal. L'autre problème très connu dans les réseaux *Ad hoc* est celui des stations cachées et des stations exposées, qui provoquent des situations de faux blocage (*false blocking*) et/ou d'impasse temporaire (*temporary deadlock*).

DCF est la méthode d'accès élémentaire aux réseaux IEEE 802.11. Les stations peuvent accéder de façon aléatoire au canal de transmission. Pour les réseaux *Ad hoc*, le standard définit deux types d'espace inter-trames :

- SIFS (*Short Inter Frame Space*) est utilisé pour séparer les transmissions appartenant à un même dialogue (fragments et acquittements). C'est le plus petit écart entre deux trames. Cette valeur est fixée par la couche physique et est calculée de telle façon que la station émettrice sera capable de commuter en mode réception pour pouvoir décoder le paquet entrant;
- DIFS (*Distributed Inter Frame Space*) est utilisé par une station voulant commencer une nouvelle transmission.

La méthode DCF se base sur un mécanisme d'évitement de collision (*Collision Avoidance*), ainsi que le principe des acquittements positifs. Ce mécanisme se résume à un algorithme simple d'envoi et d'attente (*stop-and-wait*) d'un acquittement ACK (*Acknowledgment*), où la station émettrice n'est pas autorisée à transmettre un nouveau fragment tant qu'un des deux évènements suivants n'est pas survenu [Kadoch (2004), Kurose et Ross (2003)] :

- recevoir un ACK pour le dit fragment;
- décider que le fragment a été retransmis trop souvent et abandonner la transmission de la trame.

Comme montré sur la figure 2.9, une station voulant transmettre doit écouter le canal logique avant d'émettre, selon un principe de détection de porteuse CSMA basé sur la technique MACAW (*Multiple Access Collision Avoidance protocol for Wireless LANs*) de Bharghavan et *al.* (1994). Si le canal est libre depuis un DIFS, la station est autorisée à transmettre immédiatement. La station réceptrice vérifie le CRC (*Cyclic Redundancy Check*) de la trame reçue et renvoie un acquittement après un temps SIFS. La réception de l'acquittement indiquera à l'émetteur qu'aucune collision n'a eu lieu. Si l'émetteur ne reçoit pas l'ACK après un laps de temps, il doit retransmettre le fragment jusqu'à ce qu'il obtienne un ACK ou bien il abandonne au bout d'un certain nombre de retransmissions.

Si le canal est occupé, la station devra attendre un autre DIFS après la libération du canal, puis déclencher un *backoff* aléatoire défini comme un multiple d'emplacements (slots) de temps et maintenu dans une fenêtre de contention CW (*Contention Window*). La station pourra ainsi émettre aussitôt que ce *backoff* aura expiré. Si au cours de cette attente une autre station vient d'émettre, la décrémentation du *backoff* est interrompue durant la période d'occupation du canal.

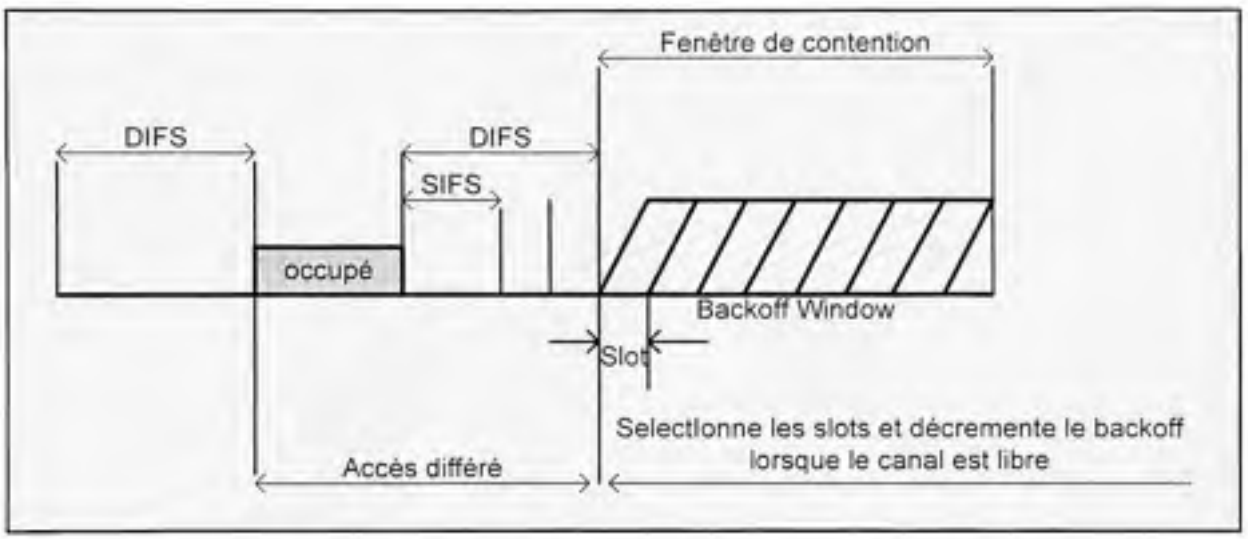

Figure 2.9 ***Méthode d'accès au canal CSMA/CA.***

Si la deuxième transmission échoue, la taille de la fenêtre de contention sera doublée. Ceci réduit la probabilité de collision au cas où il y aurait plusieurs stations essayant d'accéder au canal. Chaque station utilise un vecteur d'allocation réseau NAV (*Network Allocation Vector*) qui lui permet de savoir la durée pendant laquelle elle doit continuer à ne pas décrémenter sa fenêtre vu que le canal est occupé. La valeur du NAV est copiée à partir du champ durée de vie contenu dans la trame qui occupe le canal et déterminant la durée de transmission de cette trame.

Un problème spécifique très connu dans le monde des réseaux sans fil est celui de la station cachée et de la station exposée. Pour résoudre ces problèmes, le standard définit un mécanisme basé sur l'envoi des trames RTS/CTS (*Ready To Send/Clear To Send*) entre les

stations désirant se communiquer. Commençons tout d'abord par la présentation de quelques scénarios possibles montrant ce type de problématiques.

Comme illustré sur la figure 2.10, deux stations *A* et *B* situées chacune à l'opposé d'une troisième station *C* peuvent entendre toutes les activités de *C*, mais ne peuvent pas s'entendre l'une l'autre du fait que la distance les séparant est trop grande ou qu'un obstacle physique les empêche de communiquer entre elles.

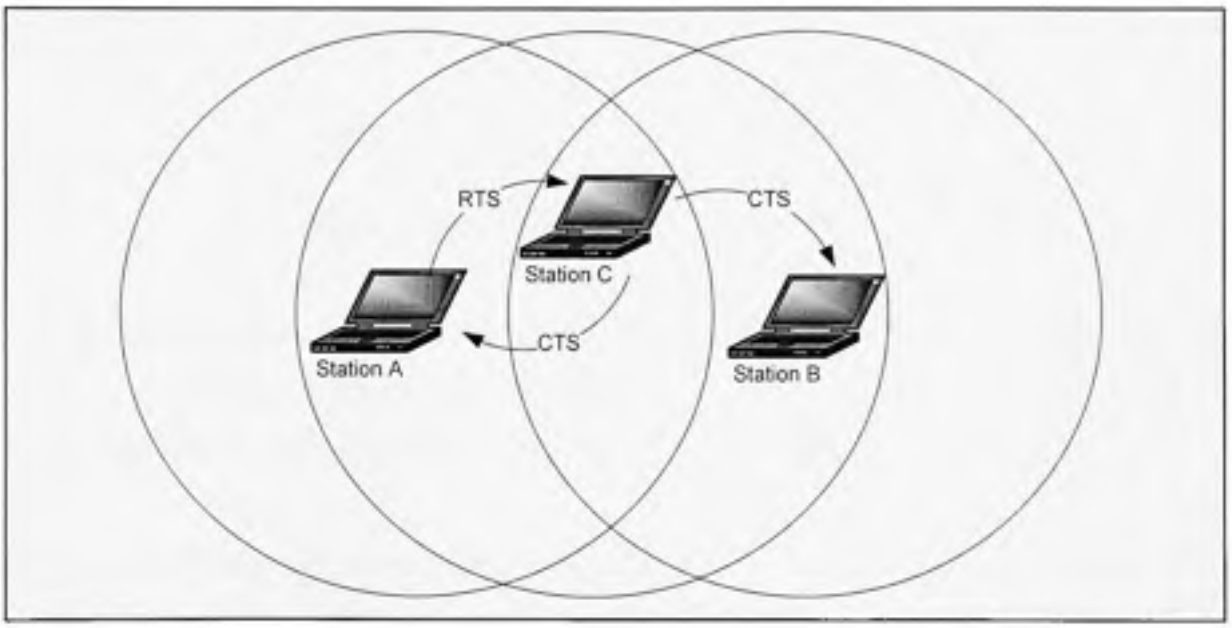

Figure 2.10 ***Phénomène de stations cachées.***

Sur la figure 2.11, le problème des stations exposées est présenté. Dans cette situation, *B* veut envoyer une trame à *A* et écoute le canal. Comme elle détecte une transmission en cours (*C* et *B* sont dans le même rayon de portée), elle en conclut qu'elle ne peut pas commencer à transmettre en direction de A, même si *C* communique avec une autre station *D*. *B* n'a aucun moyen de savoir que la transmission qu'elle veut engager avec *A* n'entraînerait pas de collision.

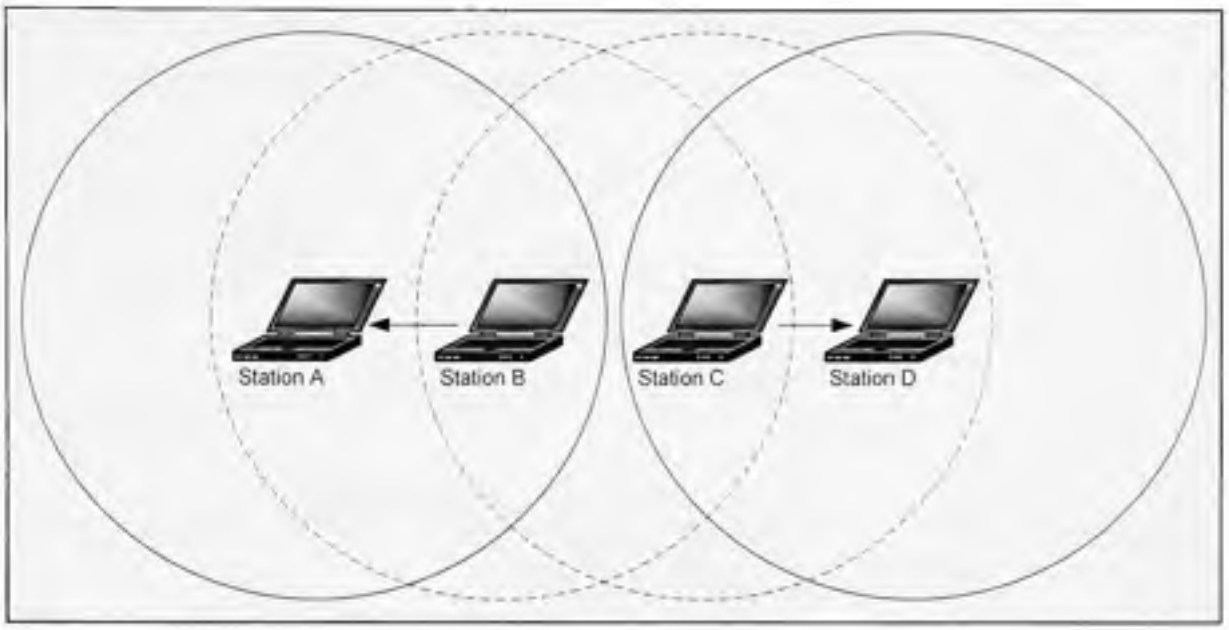

Figure 2.11 ***Phénomène de stations exposées.***

Pour résoudre ce genre de problèmes, le standard définit un mécanisme de réservation VCS (*Virtual Carrier Sense*) illustré sur la figure 2.12 et basé sur l'envoi des trames RTS/CTS entre les stations désirant communiquer (*A* et *C* dans l'exemple).

La station *A* voulant émettre, transmet tout d'abord un RTS qui contient la source, la destination, et la durée de vie de la communication. La destination *C* répond par un CTS seulement dans le cas où il n'y pas d'autres transmissions en cours dans son voisinage. La non réception d'un CTS ne permet pas à *A* de débuter sa transmission. Si *C* détecte le canal libre pendant un SIFS, il répond par un CTS qui contient la même information sur la durée de vie. Toutes les autres stations entendant le RTS et/ou le CTS lisent le champ durée de vie du RTS/CTS et mettent à jour leur NAV en extrayant l'information de durée de vie dans les différentes trames échangées.

Nous remarquons également qu'entre deux trames consécutives de la séquence RTS, CTS, données, et ACK, un SIFS sépare les différentes trames de cette séquence pour prioriser les stations *A* et *C* à gagner accès au medium. Il est à noter que RTS et CTS sont des trames de petites tailles (respectivement 20 et 14 octets) comparativement aux trames des données qui

ont une taille de l'ordre de 2300 octets. Par conséquent, la probabilité qu'une collision ait eu lieu sur les trames RTS/CTS devient très faible.

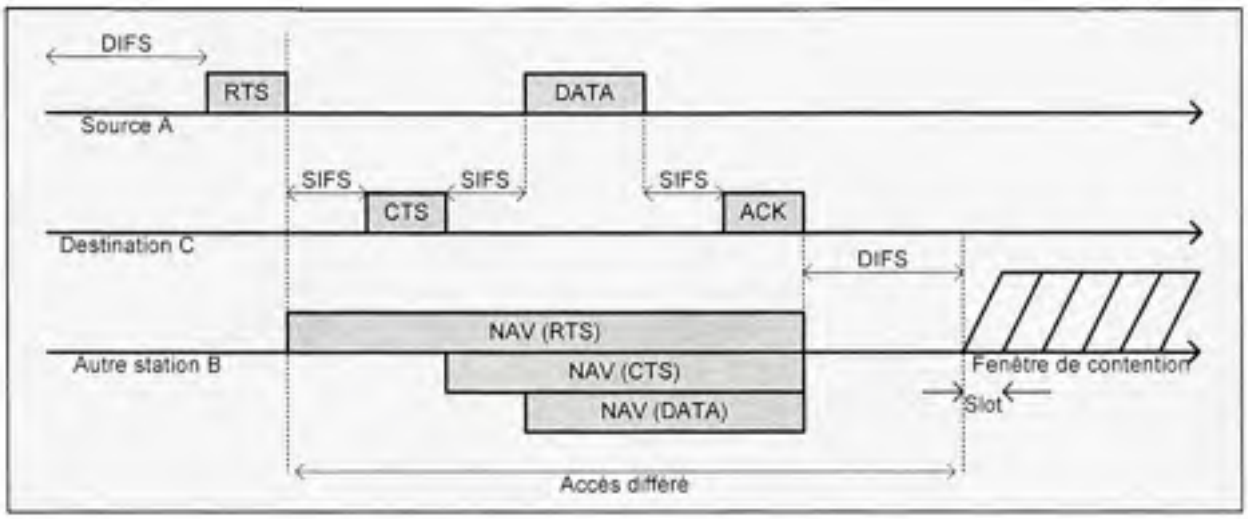

Figure 2.12 ***Mécanisme de réservation RTS/CTS.***

Ainsi, en revenant au cas présenté sur la figure 2.10, *B* est caché de *A*, mais pas de *C*. *A* transmet des données à *C*, mais *B* ne peut en aucun cas détecter l'activité de *A*. Ainsi, *C* peut décider de transmettre librement à *C* sans interférer avec la transmission de *A*. Toutefois, une collision aura lieu au niveau de *C*. RTS/CTS remédie à ce problème. Avant tout envoi, *A* et *C* s'échangent des RTS et des CTS. *B*, bien que n'écoutant pas directement *A*, est informé par l'envoi par *C* d'un CTS que le canal est occupé. Ainsi, *B* ne tentera de transmettre durant la transmission entre *A* et *C*.

Cependant, quelques situations de sous utilisation de bande passante découle de l'utilisation de RTS/CTS en présence des stations cachées et exposées. Nous essayerons de citer quelques unes dans l'exemple suivant :

Sur la figure 2.13, *D* est une station *falsely blocked* : *B* vient de recevoir un CTS et met à jour son NAV. Juste après, *D* transmet un RTS à *B*, mais ce dernier ne pourra pas lui répondre par un CTS étant donné que son NAV était positionné suite aux activités de

transmission entre *A* et *B*. Ainsi, *D* entre dans un état de blocage en attendant le CTS de la part de *B*, ce qui entraîne une sous utilisation du réseau.

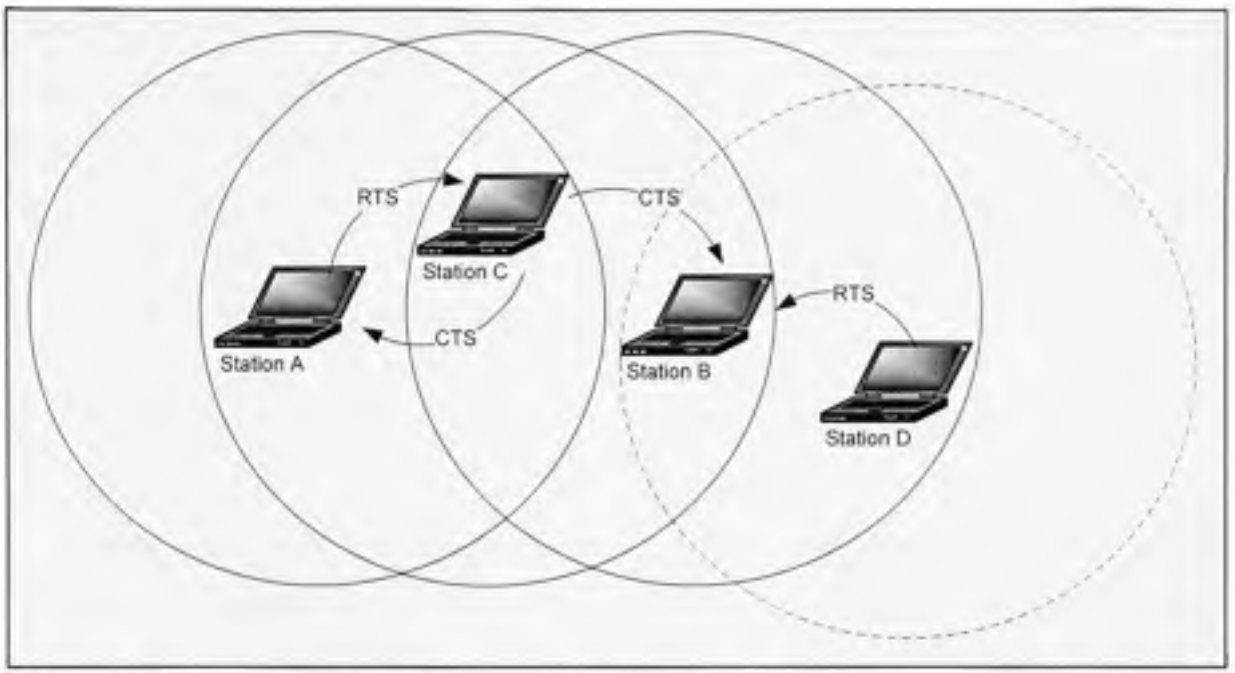

Figure 2.13 ***Situation d'une station falsely blocked / temporary deadlock.***

Éventuellement, la situation s'empire lorsque le réseau est plus dense et en cas de forte mobilité des stations. Plusieurs stations peuvent entrer dans l'état *falsely blocked*, ce qui rend des parties du réseau dans un état de *temporary deadlock*. Certes, des telles situations se débloqueront après avoir engendré plusieurs retards sur les stations.

2.3.2.4 Approches avancées pour la couche MAC dans les réseaux *Ad hoc*

Un réseau *Ad hoc* n'a plus d'antenne fixe et se reconfigure pour établir des routes à travers les nœuds mobiles présents. Dans la plupart des cas, les nœuds *Ad hoc* partagent un canal radio qui fonctionne dans la bande ISM ou dans les bandes de fréquence militaires. La solution MAC la plus répandue de nos jours est celle de la technologie IEEE 802.11 employant la méthode DCF. Une de ces problématiques est qu'elle n'offre aucune notion de qualité de service.

Avec le modèle *Best effort* ordinaire offert par DCF, les performances dépendent étroitement du nombre d'utilisateurs et des risques d'interférences par recouvrement de cellules. En effet, la bande passante est partagée et aucune allocation de ressources n'est effectuée par utilisateur. Or, des services différenciés sont nécessaires pour pouvoir maintenir une certaine qualité de service aux nouvelles applications multimédia. C'est pour cette raison que, plusieurs mécanismes ont vu le jour afin de mieux repartir la bande passante et ainsi pouvoir dans une certaine mesure garantir une qualité de transmission. [Jurdak *et al.* (2004), Wu (2006)] ont fait une synthèse complète de différentes solutions de la couche MAC des réseaux *Ad hoc*.

Aad et Castelluccia (2001) proposent trois techniques de différenciation de services au niveau MAC :

- différents facteurs d'incrémentation du *backoff* pour différentes priorités : chaque priorité a une fonction d'incrémentation de *backoff* différent. En allouant une petite fenêtre de contention aux stations les plus prioritaires, ceci permet de leur offrir un accès prioritaire par rapport aux stations moins prioritaires;
- différents DIFS : quand on associe différents DIFS à différents nœuds, on aboutit à des débits différenciés plus stables, pouvant s'appliquer aux flux TCP (*Transport Control Protocol*) des nœuds. Dans cette approche, chaque niveau de priorité a un DIFS différent, par exemple, $DIFS_{j+1}$ est inférieur à $DIFS_j$, où j représente la priorité;
- différenciation par limitation de la taille des paquets : chaque station détient d'une taille maximale pour les paquets à envoyer, qui dépend de son niveau de priorité. La première possibilité consiste à se débarrasser des paquets qui ont dépassé la taille maximale assignée à une station donnée, la deuxième possibilité est de fragmenter les paquets qui dépassent cette taille.

Basé sur la méthode classique DCF, Veres *et al.* (2001) proposent des algorithmes qui ne se basent pas sur la différenciation de service, mais plutôt sur l'estimation des mesures de performances, utilisées au niveau applicatif pour faire un contrôle d'admission. Les auteurs proposent deux nouveaux algorithmes pour le mode DCF : (i) MAC virtuel VMAC (*Virtual*

MAC) et (ii) source virtuelle VS (*Virtual Source*). VMAC observe passivement le canal radio et établit des estimations locales de délais, gigues, collisions et taux de perte de paquets. Ceci prend en considération les conditions locales du canal et les interférences causées par les cellules avoisinantes. En utilisant les estimations de VMAC, VS ajuste les paramètres de l'application et détermine si une nouvelle session, demandant un certain niveau de qualité de service, peut être admise.

La méthode EDCF de la norme IEEE 802.11e améliore le DCF classique en permettant jusqu'à huit files d'attente par station. Romdhani *et al.* (2003) proposent un nouveau schéma AEDCF (*Adaptive Enhanced DCF*) qui dérive de EDCF et prend en considération les conditions du réseau afin de mieux partager les transmissions sur le canal. Les auteurs proposent de réinitialiser la fenêtre de contention à des valeurs qui tiennent en compte du taux de collision et du niveau de priorité. Ainsi, à chaque instant, le trafic le plus prioritaire a la plus petite fenêtre de contention. Ce schéma propose également d'augmenter la fenêtre après chaque collision en utilisant un facteur qui dépend de l'état du réseau et de la classe du trafic.

Pour résoudre les problèmes d'EDCF dans les applications temps réel, Lin et Lee (2003) proposent un nouveau modèle de *forward backoff* et de contrôle d'admission qui garantit la qualité de service en termes de débit, de délai et de gigue pour le trafic temps réel. Le *forward backoff* est compatible avec le standard IEEE 802.11, il permet d'ajuster automatiquement la fenêtre de contention entre le trafic temps réel et le trafic *best effort*. Afin de ne pas influencer sur la qualité de service offerte dans le réseau, le contrôle d'admission sert à réduire les collisions non nécessaires quand la charge du trafic devient très lourde.

En se basant sur le modèle d'estimation du débit atteignable par flux proposé par Bianchi (2000), Pong et Moors (2003) dérivent un algorithme de contrôle d'admission avancé pour EDCF permettant d'estimer le débit dans le réseau. Un flux est défini comme un ensemble

de paquets appartenant au même AC (*Access Category*) d'une station et utilisant les mêmes paramètres ainsi que le même *backoff*.

Dans le but de fournir plus d'équitabilité, Vaidya *et al.* (2005) proposent un schéma d'accès appelé DFS (*Distributed Fair Scheduling*) basé sur l'idée de SCFD (*Self-Clocked Fair Queuing*) dans un milieu sans fil. Dans DFS, le *backoff* est toujours initialisé avant la transmission de la trame, il est proportionnel à la taille du paquet à envoyer et inversement proportionnel au poids du flux. Le poids représente une valeur associée avec la classe de service. Par exemple, le poids d'un trafic ayant une haute classe de service est plus grand que celui ayant une basse classe de service.

2.3.3 La couche réseau et le routage

2.3.3.1 Introduction

Le routage est un élément primordial et essentiel dans les réseaux *Ad hoc*. Dans les réseaux filaires, il existe une entité fixe (routeur) qui implémente toutes les fonctionnalités intelligentes de routage et prend en charge l'échange des informations topologiques avec les routeurs avoisinants afin de construire une table de routage répondant à des besoins spécifiques en termes de métriques de nombre de sauts, de coût, de bande passante, etc. Contrairement, un réseau *Ad hoc* n'a aucune entité fixe, toutes les stations jouent le rôle d'un routeur classique, elles doivent collaborer afin d'acheminer les paquets d'un bout à autre du réseau. La figure 2.14 illustre un exemple des stations *Ad hoc* qui relayent les paquets et prennent des décisions de routage.

Chaque station doit construire sa propre table de routage pour qu'elle devienne apte à diriger les paquets en transit. La problématique qui se pose ici est d'optimiser la fréquence des mises à jour des tables pour ne pas surcharger le réseau par un trafic de contrôle ne servant qu'à garantir une connectivité parfaite, mais au détriment d'une très haute consommation des ressources du réseau (bande passante, capacité mémoire et traitement,

énergie, etc.). Cependant, la situation n'est pas si aisée, la mobilité des stations n'a pas de limite. Une station peut se déplacer d'une zone à autre, voire disparaître brusquement suite à un manque d'énergie ou parce qu'elle a été éteinte. Ainsi, on se trouve devant un routeur mobile et instable qui ne garantit rien, mais doit malgré tout être utilisé pour passer les paquets.

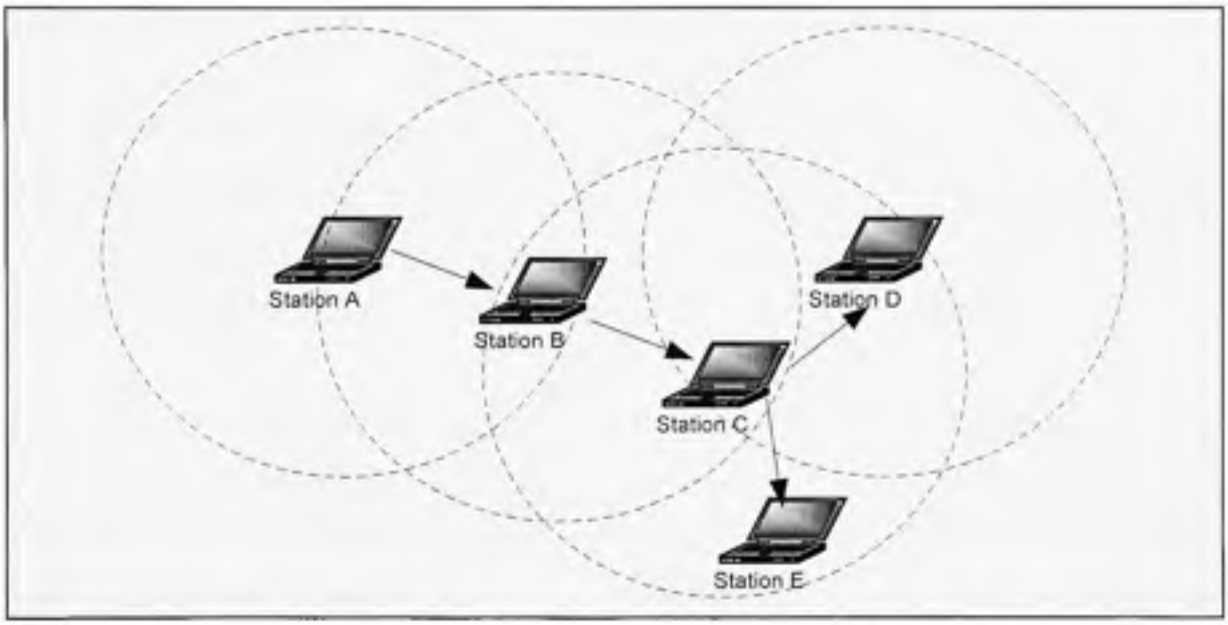

Figure 2.14 ***Principe de relai des paquets dans les réseaux Ad hoc.***

2.3.3.2 Particularités du routage dans les réseaux *Ad hoc*

Démarrant un réseau *Ad hoc*, l'autoconfiguration est la première étape à franchir. Le réseau doit donc, collaborer avec de potentiels nœuds intermédiaires, et s'auto-attribuer des adresses non redondantes. Toutes les fonctionnalités doivent, à terme, se déployer automatiquement sans paramétrage éventuel de l'utilisateur et sans l'aide d'un serveur DHCP centralisé. L'émergence d'IPv6 dans les prochaines années permettrait sans doute de simplifier la problématique étant donné que les stations pourront s'auto-attribuer des adresses IPv6 uniques construites à partir de leurs adresses physiques (MAC).

Plusieurs approches utilisant IPv4 ont été proposées pour que les réseaux *Ad hoc* survivent le temps qu'IPv6 sera déployé à l'échelle. Nous référons le lecteur aux travaux de Perkins *et al.* (2001), Mohsin et Prakash (2002), Günes et Reibel (2002), Chelius et Fleury (2002), Nesargi et Prakash (2002), Zhou *et al.* (2003), Weniger (2005), Adjih *et al.* (2005) qui ont récemment publié plusieurs approches essayant d'allouer dynamiquement les adresses IP et d'éviter la redondance de ces adresses suite aux fusions et aux partitionnements de plusieurs réseaux *Ad hoc* adjacents.

Liu et Suresh (2001) ont proposé différentes méthodes permettant de minimiser les coupures des connexions TCP suite aux changements potentiels d'adresse IP. En outre, les auteurs ont pensé à une version TCP pour les réseaux *Ad hoc* tenant compte des facteurs de délai, de mobilité des nœuds et de changement d'adresses IP au cours d'une connexion.

D'autre part, de nombreux écueils peuvent être rencontrés lors de la construction de la table de routage d'une station *Ad hoc* : Un facteur primordial est la mobilité des stations avoisinantes, ce qui entraîne des variations constantes de la connectivité et de la topologie globale du réseau. Le canal radio est une ressource rare et a des caractéristiques très particulières. Il est possible que le lien entre deux stations ne soit pas symétrique, un sens de la communication étant acceptable et pas l'autre (stations cachées et exposées). Le débit de chaque lien et son taux de perte de paquets peuvent aussi varier au cours du temps, ce qui entraîne des changements continus au niveau des routes. Par conséquent, la table de routage doit tenir compte de toutes ces situations.

À part ces particularités, la question qui se pose et continue à être débattue à l'IETF est à savoir s'il vaut la peine de maintenir assez fréquemment les tables de routage qui changent sans arrêt dans un contexte *Ad hoc* ou n'est-il pas plus judicieux de déterminer la table de routage au moment de l'arrivée des paquets ? En effet, l'échange des tables de routage, suite aux variations des routes, est une vraie problématique dans les réseaux *Ad hoc*. Le fait de garder une connectivité maximale nécessite un envoi continu des tables de routage au détriment de la bande passante du medium radio. Aussi, le fait de réagir lorsque le flux des

paquets est prêt à être émis, exige un délai d'attente de recherche de route, mais évite énormément de trafic de contrôle dans le réseau.

Une multitude de protocoles de routage existent dans la littérature. Il n'est pas question de faire une étude complète et détaillée de ces protocoles. Toutefois, nous avons essayé de les classifier en trois grandes familles comme illustré sur la figure 2.15:

- géographique : où le protocole utilise tout système de positionnement tel qu'un GPS (*Global Positioning System*);
- à plat : où le réseau est considéré comme à plat sans aucune hiérarchie;
- hiérarchique : où le protocole de routage exploite soit des structures basées sur des clusters dynamiquement construits et maintenus, soit sur une dorsale virtuelle pour acheminer le trafic de contrôle. La façon dont les clusters et les dorsales sont construits sera décrite dans le chapitre 3.

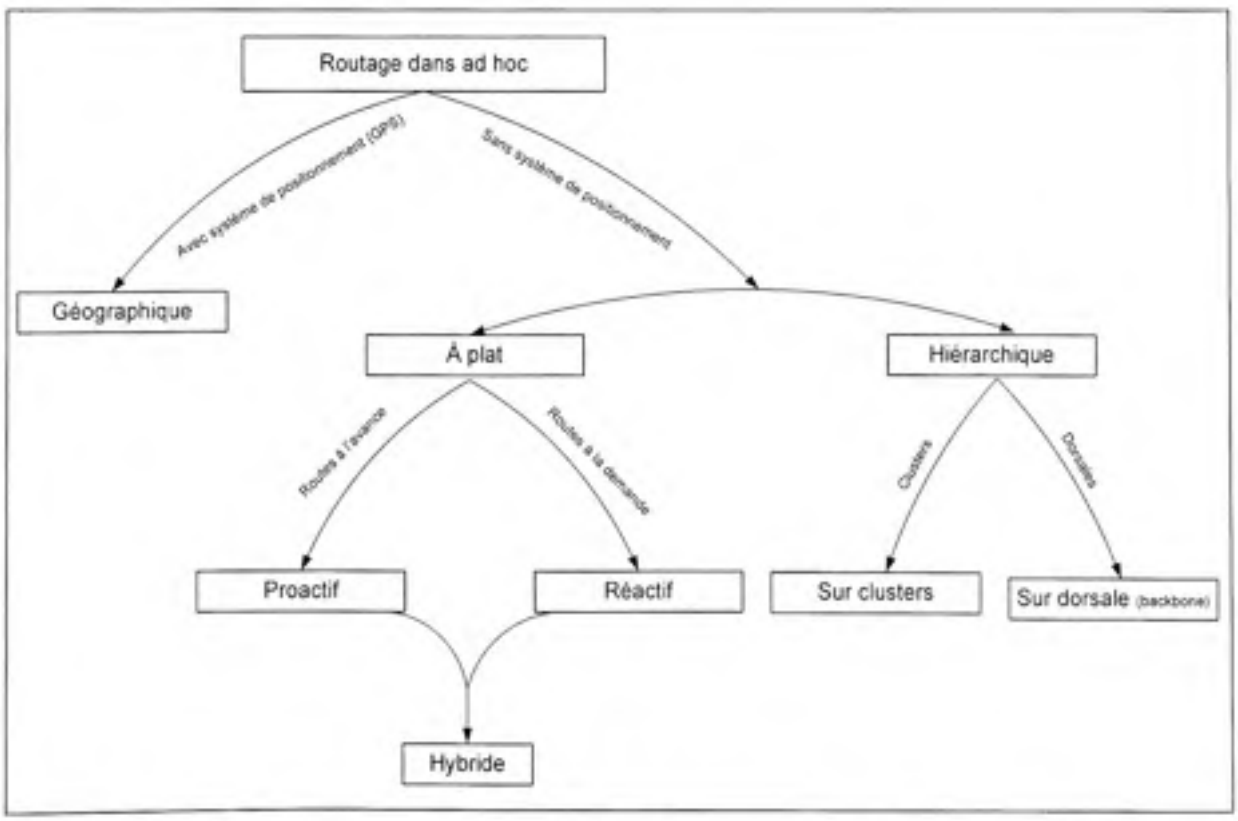

Figure 2.15 ***Taxonomie des protocoles de routage dans les réseaux Ad hoc.***

Un protocole de routage pour les réseaux *Ad hoc* devrait présenter les propriétés suivantes :

- réagir rapidement aux changements topologiques;
- minimiser le trafic de contrôle;
- minimiser le délai de bout en bout;
- réduire le taux de perte de paquets.

2.3.3.3 Routage à plat

Dans les réseaux *Ad hoc*, les nœuds sont à la fois stations ordinaires et routeurs. Aucune organisation hiérarchique préalable n'est en place. La majorité de travaux de recherche sur le routage consistent à traiter de façon égalitaire tous les nœuds, considérés ainsi à plat. Le routage doit se faire sans intervention extérieure et sans hiérarchie fixée à l'avance. Deux approches ont donc été proposées : le routage proactif et le routage réactif. Par la suite, d'autres propositions ont essayé de combiner les avantages du routage proactif et réactif dans une approche hybride.

2.3.3.3.1 Routage proactif

Les protocoles proactifs maintiennent des informations globales sur le réseau avant toute utilisation. Chaque nœud maintient une route vers chacune des destinations possibles du réseau. Pour cela, tout nœud doit annoncer sa présence par l'inondation d'un paquet de contrôle dans le réseau comme dans OSPF. Ceci entraîne une tempête d'inondation [Tseng *et al.* (2002)] causant de multiples collisions et rendant le réseau inexploitable. Par conséquent, les solutions se focalisent sur la réduction du trafic de contrôle dans un réseau qui souffre déjà du manque de bande passante.

Perkins et Bhagwat (1994) ont proposé DSDV qui adopte un algorithme de type vecteur de distance (*Bellman-Ford*). La destination utilise un numéro de séquence croissant, servant à éviter les boucles de routage. Cependant, des mises à jour sont déclenchées lors de chaque changement de route. Nous concluons que ce type de protocole est mieux adapté à des

topologies peu changeantes. Dans le cas des nœuds très mobiles, le trafic de contrôle peut causer des problèmes de performance [Toh (2001)] en termes de débit étant donné que les nœuds ne maintiennent qu'une seule route qui n'est pas à jour vers chaque destination.

Le protocole OLSR de Claussen et Jacquet (2003) est un protocole à état de liens optimisant la diffusion du trafic de contrôle. Des paquets *hello* contenant la liste des voisins sont périodiquement échangés. L'objectif est de permettre à chaque nœud de connaître la liste de ses voisins à deux sauts afin de sélectionner des relais multipoints MPR (*Multipoint Relay*). Ainsi, les MPR d'un nœud quelconque sont définis de sorte que si ces MPR relaient un paquet, tous les voisins à deux sauts du nœud le reçoivent. Ceci permet de minimiser l'effet de l'inondation d'un paquet *hello* de façon que ce dernier ne soit relayé par un nœud x que s'il provient d'un autre nœud ayant choisi le nœud x en tant qu'un MPR.

En revanche, lorsque l'inondation dans OLSR se fait à un rythme plus lent, certaines routes pourraient ne pas être présentes, engendrant ainsi des pertes de paquets. Benzaid *et al.* (2002) ont proposé une alternative à la première version d'OLSR (2001) tenant compte des nœuds fortement mobiles de façon à leur permettre exclusivement d'envoyer plus fréquemment des messages *hello*.

2.3.3.3.2 Routage réactif

Afin de réduire au maximum le trafic de contrôle, les protocoles de routage réactifs ne demandent d'informations que lorsqu'ils en ont besoin. Ils effectuent des requêtes de routes RREQ (*Route Request*), inondées dans tout le réseau lorsqu'un nœud a besoin de transmettre des informations.

Johnson *et al.* (2007) ont publié la dernière version du protocole DSR proposé pour la première fois en 1996. Dans DSR, les nœuds intermédiaires enregistrent un état temporaire leur permettant de faire suivre la réponse RREP (*Route Reply*), s'il y en a une, vers la source. Lorsque la requête de route atteint la destination, cette dernière répond et crée une

route au sein de chaque nœud intermédiaire. Il faut donc mettre en place un mécanisme complexe de cache et de maintenance de routes pour reconstruire localement ces routes en cas de bris causés par les changements de topologie. Cependant, le fait d'attendre que la route brise pour initier une redécouverte de routes, peut provoquer beaucoup de pertes de paquets utiles. De plus, la redécouverte de routes ajoute un délai important avant le premier envoi de paquet, pouvant perturber certaines applications.

Qin et Lee (2003) ont proposé IDSR (*Improved DSR*) qui n'est qu'une amélioration de la première version de DSR. Dans le but de réduire la quantité du trafic de contrôle, IDSR diffuse un message de mise à jour lorsqu'un nœud intermédiaire, reconnaissant une route vers une destination, retourne un message RREP. Le message de mise à jour permet à tous les nœuds d'obtenir une route vers la destination, cette route sera utilisée quand c'est nécessaire.

Dans Perkins *et al.* (2003), les auteurs présentent AODV, un protocole similaire à DSR. Cependant, ils proposent d'éviter le routage par la source, utilisé dans DSR. Par conséquent, la route n'est plus contenue dans l'entête des messages de découverte de route RREQ et un *overhead* important est ainsi supprimé.

En outre, AODV utilise les messages *hello* afin de maintenir la liste des voisins présents. Lorsqu'un voisin n'est plus dans le voisinage, les routes passant par lui deviennent invalides, notifiant les nœuds empruntant ces routes qu'ils doivent découvrir une nouvelle route. Ainsi, les bris de routes sont détectés plus rapidement, conduisant à moins de pertes de paquets utiles. De plus, la destination dans AODV maintient un numéro de séquence permettant d'éviter d'utiliser des informations obsolètes dans les caches de routage.

Costa *et al.* (2004) ont proposé d'optimiser la redécouverte d'une route en limitant la recherche d'une nouvelle route à des zones proches de l'ancienne route.

2.3.3.3.3 Routage hybride

Certaines approches, appelées hybrides, tentent de combiner les avantages du proactif et du réactif afin de router les paquets d'un bout à l'autre. Haas et Pearlman (2001) proposent ZRP qui maintient des informations proactives au sein d'une zone locale. Les communications locales sont donc optimisées. Un nœud n'envoie de requêtes de route RREQ que lorsqu'il ne connaît pas la destination. Au lieu d'inonder le réseau, le nœud envoie la requête aux nœuds en périphérie de zone, nommés les nœuds bordures. Ceux-ci acheminent ensuite cette requête à leurs propres nœuds bordures, jusqu'à ce que la destination soit trouvée. La réponse de route RREP emprunte la suite de nœuds bordures utilisés à l'aller par RREQ.

Cependant, il faut prendre quelques précautions dans ZRP lorsque le nombre de nœuds bordures augmente de façon drastique de sorte à ne pas détériorer les performances en relayant plusieurs fois la même requête lors du relais de requêtes. Les auteurs dans [Pearlman et Haas (1999)] avaient déjà proposé un mécanisme de sélection plus fine d'un sous-ensemble de nœuds bordures, ayant la responsabilité exclusive de relayer les messages de contrôle.

Wang et Olariu (2004) ont proposé TZRP (*Two Zone Routing Protocol*) permettant de mieux adapter ZRP en créant deux différentes zones pour optimiser l'inondation et l'*overhead*. Toutefois, les auteurs ne justifient pas leur choix et il pourrait être intéressant d'examiner d'autres structures plus persistantes.

Gerla *et al.* (2000) ont présenté le protocole LANMAR pour des réseaux dans lesquels des groupes ont été préconfigurés. Un nœud *landmark* est élu par groupe. Le protocole se base sur une connaissance proactive de sa zone locale. Un algorithme à vecteur de distance permet de construire une route vers chaque *landmark*. En outre, si un nœud d'un groupe est en dehors de la zone de son *landmark*, une route spécifique est maintenue vers ce nœud grâce au même algorithme à vecteur de distance. Ce protocole semble performant pour des

scénarios spécifiques, mais le fait de prendre comme hypothèse des groupes statiques fixés au préalable rend difficile son passage à l'échelle.

2.3.3.4 Routage hiérarchique

À la différence du routage à plat, le routage hiérarchique essaye de calquer le mode de routage utilisé dans les réseaux filaires en créant de structures hiérarchisées servant au routage. La figure 2.16 illustre une comparaison entre ces deux types de routage.

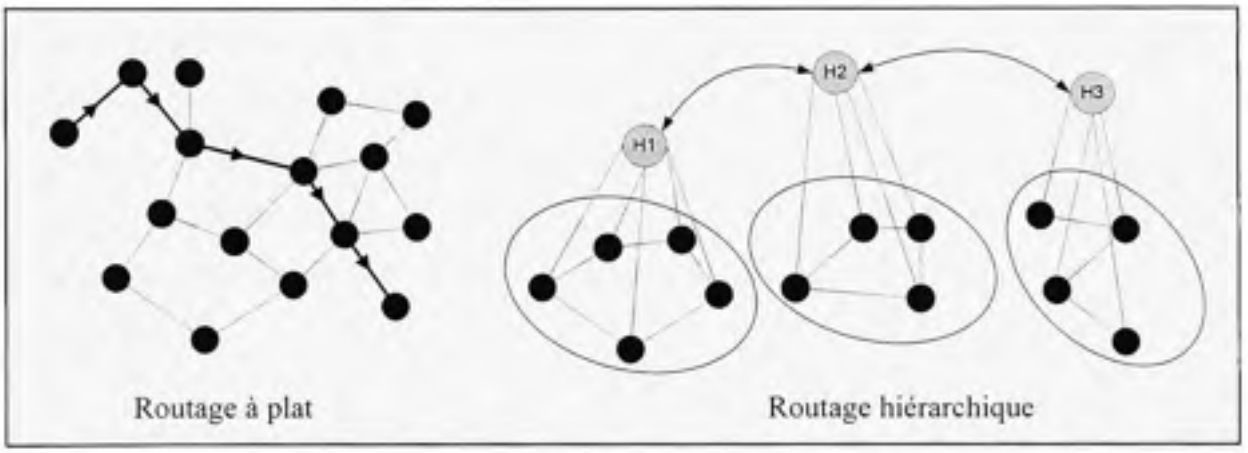

Figure 2.16 ***Routage à plat versus routage hiérarchique.***

Nous avons séparé les approches de routage hiérarchique en deux classes : routage basé sur des clusters et routage basé sur des dorsales virtuelles.

2.3.3.4.1 Routage basé sur des clusters

Les clusters sont des zones gérés par des nœuds-chefs appelés *clusterhead*. Ces *clusterhead* sont élus et se chargent de coordonner le routage dans leur cluster. La notion des clusters sera détaillée dans le chapitre 3. Nous nous limitions ici à décrire les protocoles utilisant des clusters pour des fins de routage.

Jiang *et al.* (1999) ont proposé CBRP (*Cluster Based Routing Protocol*) qui permet un routage sur clusters. Chaque nœud envoie périodiquement des messages *hello* contenant la liste de ses voisins à un saut, et celle des clusters adjacents. Un *clusterhead* peut donc choisir un nœud passerelle vers chacun des autres *clusterhead* à moins de 3 sauts de lui. L'ensemble des nœuds passerelles et des *clusterhead* forme un sous-ensemble connexe sur lequel les requêtes RREQ peuvent circuler. Toutefois, nous pensons que CBRP présente les désavantages suivants :

- engendrer un énorme trafic de contrôle, car l'ensemble des passerelles et de *clusterhead* peut être redondant;
- du fait que la route est constituée d'une liste de nœuds, et non de clusters, il suffit qu'un nœud quelconque se déplace pour que toute la route casse;
- entraîner un allongement de route à chaque reconstruction : lorsqu'un nœud relayant un paquet n'obtient aucun acquittement du prochain saut, le nœud enclenche une reconstruction locale de route en essayant d'atteindre le prochain saut via un intermédiaire.

Tan *et al.* (2005) ont proposé de modifier le protocole DSR de façon qu'il s'adapte sur une topologie de clusters. Les RREQ ne sont ainsi traités et relayés que par les *clusterhead* et les passerelles. Afin d'optimiser l'utilisation des ressources radio, les auteurs ont également inséré de nouveaux champs spéciaux dans les paquets de données pour des fins de contrôle. Une telle optimisation peut être applicable à tout protocole.

2.3.3.4.2 Routage basé sur des dorsales virtuelles (*Backbone*)

Das et Bharghawan (1997) proposent de créer une dorsale virtuelle maillée. Ils commencent par élire de façon distribuée un ensemble de nœuds dominants formant une dorsale. Ces dominants doivent découvrir les dominants avoisinants par l'inondation des messages *hello* jusqu'à 3 sauts. Le trafic de contrôle est un facteur négatif à considérer. La dorsale est ensuite utilisée afin d'optimiser les inondations de paquets de topologie : seuls les nœuds de

la dorsale sont habilités à retransmettre les paquets. Comme tout nœud possède au moins un voisin dans la dorsale, tous les nœuds reçoivent les paquets de topologie.

Dans CEDAR (*Core-Extraction Distributed Ad hoc Routing algorithm*), Sivakumar *et al.* (1999) proposent d'élire dynamiquement un cœur de réseau stable qui approximise un ensemble dominant minimal (dorsale). Cette dorsale est utilisée pour inonder les changements dans la bande passante des liens radio des nœuds du cœur et d'assurer le routage dans le réseau en impliquant le moins possible des nœuds tout en limitant le trafic de contrôle. Un mécanisme est proposé pour que seuls les liens radio stables soient propagés loin dans le réseau.

Dans le but d'éviter tout goulot d'étranglement, la dorsale dans CEDAR ne sert pas à router les paquets de données. D'autre part, et afin de résoudre les problèmes de mobilité des nœuds, des routes de secours doivent être maintenues pendant le temps de reconstruction de la route principale lorsque celle-ci est perdue. Les routes calculées par CEDAR convergent vers des routes optimales lorsque la topologie devient stable. Sinha *et al.* (2001) ont plus tard utilisé cette dorsale afin d'optimiser les inondations engendrées par les protocoles AODV et DSR.

Dans DDR (*Distributed Dynamic Routing*) [Nikaein et Bonnet (2004)], chaque nœud cherche un père qui n'est qu'un voisin de degré supérieur afin de créer une forêt d'arbres de recouvrement (*spanning tree*). Chaque arbre étant une zone, implémentant un protocole de routage proactif et utilisant l'arbre pour acheminer les informations topologiques. Le routage interzones est réactif et les requêtes sont acheminées aux zones avoisinantes via des passerelles.

Toutefois, nous pensons que l'architecture présentée dans DDR ne permet pas d'optimiser le trafic de contrôle puisque chaque nœud relaye la requête de route. De plus, les routes doivent passer par la structure en forêt, augmentant potentiellement la longueur totale de la route.

Dans Rieck *et al.* (2005), les auteurs utilisent l'algorithme de Wu et Li (2001) pour la construction de leur dorsale. Un nœud à moins de d sauts est considéré comme un voisin direct dans leur graphe. Ainsi, la connaissance du voisinage à d sauts est primordiale. Un protocole de routage proactif est utilisé au sein de la zone. Par contre, les auteurs ne précisent pas de protocole pour le routage interzone, présentant seulement la possibilité d'adapter un protocole à état de liens sur la topologie des dominants. De plus, aucune étude de performances n'est donnée pour valider leur proposition.

2.3.3.5 Routage géographique

Ce type de routage utilise les informations fournies par un système de positionnement tel que le GPS dans le but d'optimiser la construction des routes. Cependant, nous pensons que l'obligation d'embarquer systématiquement un tel système de positionnement est un point bloquant pour beaucoup d'équipements sans fil bon marché.

Ko et Vaidya (1998) ont proposé LAR (*Location-Aided Routing*) qui est similaire à DSR, mais il utilise un routage aidé par la localisation des nœuds. Ayant l'information sur l'ancienne position de la destination, chaque nœud intermédiaire choisit de relayer la requête de route RREQ selon la position de la destination, de la source, et selon sa propre position. Un tel choix permet de diriger l'inondation vers l'ancienne position de la destination. Ceci conduit à une réduction importante du trafic de contrôle (*overhead*) en limitant la recherche d'un chemin à des zones limitées, fournies par le système de positionnement.

GPSR (*Greedy Perimeter Stateless Routing*) de [Karp et Hung (2000)] est un protocole de routage glouton (*Greedy*) où un nœud relaie un paquet de données vers le prochain saut le plus proche de la destination. Si un tel nœud n'existe pas, le relai se fait le long du périmètre de la zone sans nœud. Toutefois, pour que le périmètre soit bien emprunté sans qu'il ait des boucles, il est nécessaire de construire au préalable un graphe sans boucle.

Liao *et al.* (2001) ont proposé GRID, un protocole de routage réactif utilisant le concept des clusters. Un système de positionnement GPS permet de découper tout le réseau en plusieurs clusters (*grid*) qui contiennent un nombre de nœuds dépendant de leurs positions géographiques. Des *clusterhead* sont élus (un par cluster) et serviront pour faire un routage interclusters par le biais d'un des protocoles de routage réactif. L'utilisation du GPS permet de réduire le trafic additionnel utilisée par la procédure de formation des clusters. Toutefois, l'information sur la position des destinations doit toujours être à jour afin de minimiser autant que possible le nombre de requêtes de route RREQ.

Chao *et al.* (2003) ont présenté le protocole ECGRID (*Energy Conserving GRID*) qui n'est qu'une amélioration du protocole GRID. L'objectif était d'étendre la durée de vie des *clusterhead* qui sont responsables de toute la gestion de leur zone. Dans ECGRID, le choix du *clusterhead* est plus sophistiqué. Le niveau d'énergie d'un nœud a été pris en considération et des mécanismes de réélection ont été intégrés de sorte qu'un nouveau *clusterhead* doit être élu lorsque le niveau d'énergie de l'ancien *clusterhead* atteint un certain seuil. Les nœuds ordinaires utilisent également un mécanisme nommé RAS (*Remote Activated Switch*) leur permettant de s'endormir lorsqu'ils n'ont pas de communications à faire. Cependant, le mécanisme de réélection engendre un autre *overhead* à considérer.

2.4 *Cross-Layer* : une interaction entre les couches OSI

Le *Cross-Layer* est une conception multicouche de réseaux sans fil qui apporte un gain de performance significatif par rapport à une approche conventionnelle OSI. Srivastava *et* Motani (2005) ont bien décrit les différents *designs* d'interaction entre les couches protocolaires. Ils les ont rangés dans 6 classes comme illustré sur la figure 2.17.

Le modèle traditionnel OSI propose un découpage en plusieurs couches indépendantes. Chaque couche est conçue séparément, et l'interaction entre les couches se fait via une interface bien définie. L'avantage principal de ce type d'approche est sa flexibilité architecturale. Ainsi, le fait de remplacer les fonctions d'une couche quelconque n'influence

pas les autres couches. Dans les réseaux câblés, cette solution est parfaite dû à l'indépendance des entités communicatrices. Par contre, ceci semble désavantageux pour les réseaux sans fil *Ad hoc* où chaque activité sur le canal peut être entendue par les nœuds avoisinants et perçue comme des interférences.

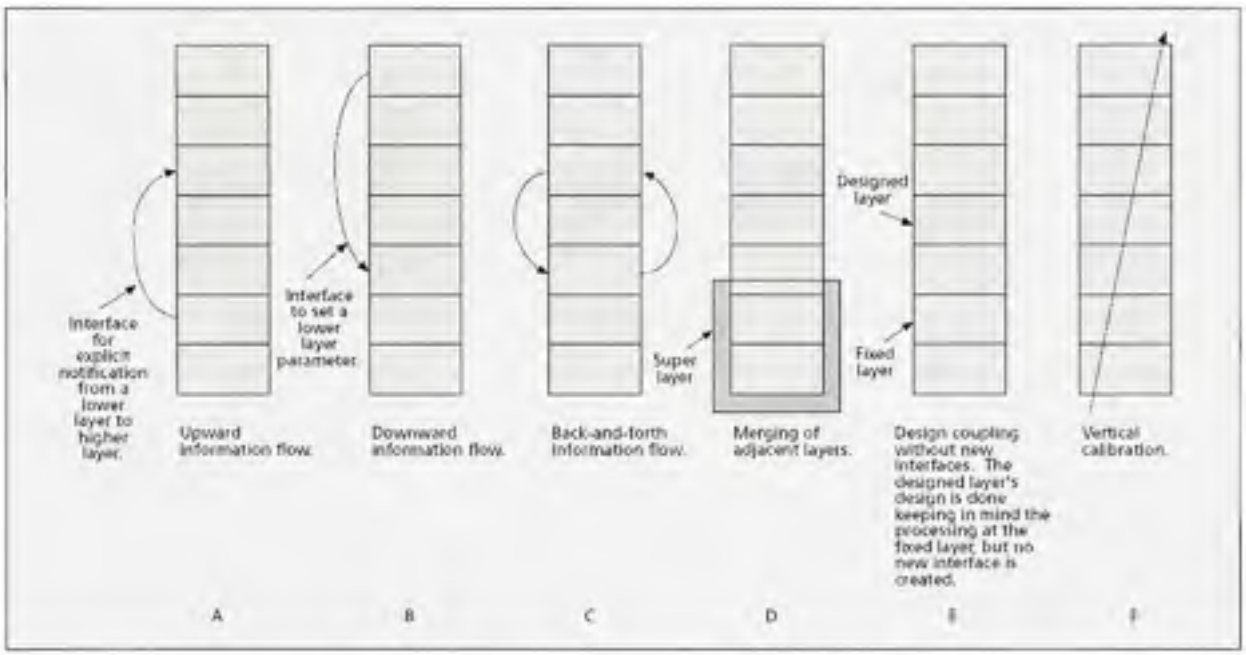

Figure 2.17 ***Différents design possibles de Cross-Layer.***

(Tiré de Srivastava et Motani, 2005)

Source : Cette figure a été tirée de Srivastava et Motani (2005) « Cross-Layer Design: A Survey and the Road Ahead » et correspond à la « Figure 1 Illustrating the different kinds of cross-layer design proposals. The rectangular boxes represent the protocol layers » présentée en page 3 dans le document original. (La référence complète du document est présentée dans la liste de références).

Il est primordial de décrire brièvement ce type d'approches de grand intérêt dans les réseaux *Ad hoc*. Nous référons le lecteur aux travaux de [Madan *et al.* (2006), Choi *et al.* (2008b)] qui ont synthétisé les problèmes et les solutions proposées. En effet, l'interdépendance des couches dans les réseaux sans fil nécessite de trouver des mécanismes d'accès au canal plus complexes. D'une part, ces mécanismes doivent contrôler les interférences éprouvées par les récepteurs. D'autre part, ils doivent exploiter la réutilisation spatiale des canaux et gérer les transmissions concurrentes par canal.

Ce contrôle au niveau MAC influence grandement la couche physique. Si les interférences sont très élevées, la couche physique devrait réduire son taux de transmission. Au contraire, si les interférences sont basses, la couche physique devrait bénéficier de ces conditions et transmettre à un taux plus élevé. Ceci semble être en contradiction avec les approches des réseaux filaires où deux transmissions quelconques concurrentes causent toujours une collision, et il n'existe aucun concept d'adaptation de taux de transmission en fonction du trafic existant.

Un autre aspect est le contrôle de puissance de transmission dans les réseaux sans fil *Ad hoc*. Plus la puissance de transmission est haute, plus la puissance du signal reçu est élevée. Cependant, si nous augmentons la puissance de transmission, nous créerons plus d'interférences au niveau d'autres nœuds avoisinants. Ainsi, le contrôle de puissance dans les réseaux *Ad hoc* est étroitement couplé tant à la couche physique qu'à la couche liaison de données.

D'autre part, le milieu sans fil permet des communications entre deux nœuds quelconques dans une région. Plus les nœuds sont éloignés l'un de l'autre, plus le débit réalisable sera inférieur. Si un lien direct de point à point existe entre la source et la destination, mais ne fournit qu'un débit médiocre, le protocole de routage peut décider de transmettre sur une autre route plus longue en traversant plusieurs nœuds intermédiaires. De cette façon, le paquet sera transmis sur plusieurs liens plus courts, provoquant ainsi une augmentation dans la consommation des ressources de plusieurs nœuds, mais pendant une période de temps plus courte, car les transmissions intermédiaires se font à un débit beaucoup plus élevé.

Il est également facile de voir qu'un changement de la politique de routage peut influencer les couches basses. À savoir, si un protocole de routage décide de router sur un chemin indirect composé de plusieurs liens au lieu de router directement d'une source à une destination, nous aurons plus d'activité sur les nœuds intermédiaires, et ceci créera plus d'interférences sur d'autres nœuds dans le voisinage. En revanche, ces liens courts sont plus résistants aux interférences qu'un simple lien direct plus long.

Des exemples ci-dessus, il est clair qu'un changement d'un protocole d'une couche affectera les autres couches. Nous pouvons ainsi conclure que l'indépendance des couches proposée dans le modèle conventionnel OSI doit être repensée pour ne pas réduire les performances des réseaux *Ad hoc*. D'où l'utilité du *Cross-Layer*.

2.5 Qualité de service dans les réseaux *Ad hoc*

Il n'est pas question ici de faire l'état de l'art sur toutes les approches de QoS. En effet, la qualité de service est un sujet très vaste, nous essayons dans cette dissertation de décortiquer brièvement des points intéressants à considérer dans ce contexte.

Une fois le réseau est établi et les adresses IP sont allouées, la prochaine étape est de trouver un chemin de la source vers la destination de façon à fournir les ressources nécessaires satisfaisant le niveau de service requis tout en prenant en considération le facteur de mobilité. Ce processus est appelé le routage avec QoS. Après avoir déterminé le chemin, un protocole de réservation des ressources peut être utilisé pour réserver les ressources nécessaires tout au long du chemin. La QoS requiert la négociation entre l'utilisateur et le réseau, le contrôle d'admission, la réservation des ressources et l'ordonnancement des paquets selon leurs priorités.

Il est indispensable de concevoir un protocole efficace lorsque le nombre de participants et leur mobilité respective augmentent. Un protocole répond généralement à un certain nombre de contraintes, rarement à toutes de façon efficace. Il faut donc quantifier le comportement de chaque protocole proposé vis-à-vis ces différents critères.

Reddy *et al.* (2006) ont fait une synthèse globale pour une large gamme de solutions de qualité de service tant au niveau de la couche liaison de données qu'au niveau de la couche réseau. Ils ont également synthétisé les solutions *Cross-Layer* permettant une meilleure interaction entre les couches du modèle OSI et offrant un niveau de QoS plus élevé. Les auteurs ont classifié ces solutions de la façon illustrée sur la figure 2.18.

Toutes les approches doivent prendre en considération des facteurs tels que la dynamique de la topologie, la qualité des liens, l'interférence multi-usagers, et une très haute tolérance aux pannes. À cette fin, il est fondamental de définir des mécanismes pour la sélection intelligente et dynamique de stratégies de routage qui permettent l'adaptation aux changements rapides et imprévus des caractéristiques des réseaux *Ad hoc* à travers le temps.

La découverte proactive des caractéristiques du réseau telles que la proximité relative des nœuds et la qualité des liens se font grâce à une approche d'intelligence de groupe. En effet, une connaissance globale de l'état du réseau en mouvement émerge à partir d'une multitude simple d'interactions et d'échanges entre les nœuds (*Cross-Layer*).

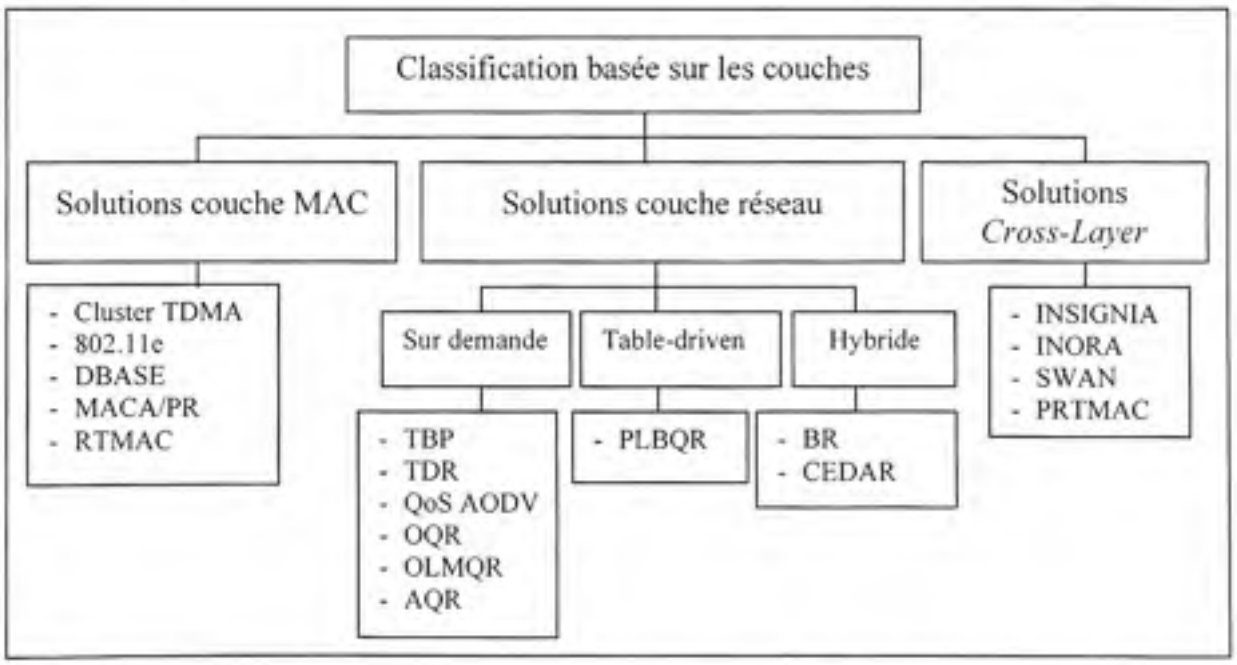

Figure 2.18 ***Classification des solutions de QoS.***

(Tiré de Reddy et al., 2006)

Source : Cette figure a été tirée de Reddy *et al.* 2006 « Quality of service provisioning in Ad hoc wireless networks: a survey of issues and solutions » et correspond à la « Fig.3 Layer-wise classification of QoS solutions » présentée en page 6 dans le document original. (La référence complète du document est présentée dans la liste de références).

Dans les réseaux filaires, les deux grands modèles de QoS sont *IntServ*[9] et *DiffServ*[10]. *IntServ* offre un service par flux où un flux est représenté par une session établie entre deux applications utilisatrices. Chaque Routeur *IntServ* maintient les informations sur les états de tous les flux comme la bande passante requise, le délai et le coût. Le protocole RSVP (*Resource Reservation Protocol*) est utilisé pour réserver les ressources tout au long du chemin. Ainsi, le volume d'informations maintenues par routeur est proportionnel au nombre de flux, ce qui entraîne le problème de mise à l'échelle (*Scalability*) dans Internet.

Ce problème peut ne pas se produire dans les réseaux *Ad hoc* où le nombre de flux est trop restreint dans un réseau de petite taille. Pourtant, il est très difficile de maintenir les états des flux sur un nœud à cause des limitations sur la capacité de traitement, le manque d'énergie, les changements fréquents (mobilité) et la variation de la capacité des liens avec le temps. Le modèle *DiffServ* a été proposé pour éviter le problème de mise à l'échelle imposé par *IntServ*. Dans *DiffServ*, les flux sont agrégés dans des classes de service différentes et chaque flux appartient à l'une des classes de service de *DiffServ*.

Ces deux modèles ne peuvent pas être appliqués comme tels dans les réseaux *Ad hoc* à cause des caractéristiques inhérentes de ces réseaux. Un modèle FQMM (*Flexible QoS Model for Mobile Ad hoc Networks*) a été proposé par [Xiao *et al.* (2000)], couplant *IntServ* et *DiffServ* dans un contexte *Ad hoc*. FQMM fournit le service par flux et évite le problème de mise à l'échelle en classifiant le trafic moins prioritaire dans des classes de service différentes. Cependant, plusieurs problèmes n'ont pas été résolus comme la classification du trafic, le choix d'un service par flux ou d'un service agrégé pour un flux donné, la quantité du trafic appartenant au service par flux ainsi que les mécanismes utilisés par les nœuds intermédiaires pour avoir une idée sur le flux et ordonnancer ou retransmettre le trafic.

---

[9] Integrated Services, RFC 1633
[10] Differentiated Services, RFC 2474 et RFC 2475

Les chercheurs ont conclu qu'une meilleure solution de QoS pour les réseaux *Ad hoc* devrait intégrer des solutions de la couche liaison de données avec celles de la couche réseau de sorte que chaque couche fournisse des *feedback* à l'autre pour faire telle ou telle action suivant les conditions du réseau. Dans ce qui suit, nous allons mettre l'emphase sur ce type de solutions tout en essayant d'extraire les lacunes que nous *jugeons* pertinentes.

INSIGNIA (*In-Band Signaling Support for QoS*) de [Lee *et al.* (1999)] fournit une approche pour l'intégration de la qualité de service en combinant la signalisation dans la bande (*in band*), le contrôle d'admission et l'ordonnancement des paquets (*packet scheduling*). Un schéma de réservation de type *soft state* permet une libération rapide des ressources durant la reconfiguration du chemin. Lorsque les ressources ne sont plus disponibles, l'application peut changer son niveau de service à un niveau plus bas, voire un niveau *best effort*. L'utilisation de la signalisation dans la bande permet de ne pas allouer les ressources qu'après le début de transmission des données. Cependant, INSIGNIA semble inapproprié pour les applications temps réel qui demandent de fortes contraintes de QoS.

INORA (*Insignia Temporally Ordered Routing Algorithm*) de [Dharmaraju *et al.* (2002)] est plus avantageux comparativement à INSIGNIA dans le sens où il peut séparer le flux sur plusieurs routes pour assurer un service garanti. Dans INORA, aucune ressource n'est réservée qu'après le début de transmission des données et les paquets de données doivent être transmis en mode *best effort* dans le cas où le contrôle d'admission échoue sur les nœuds intermédiaires. Par conséquent, ce modèle peut ne pas être applicable pour des flux qui ont de fortes contraintes de QoS.

SWAN (*Stateless Wireless Ad hoc Networks*) de [Ahn *et al.* (2002)] supporte les applications temps réel en utilisant un protocole MAC de type *best effort* et sans faire aucune réservation de ressources. SWAN utilise les *feedback* pour régler le trafic temps réel en cas de congestion. Nous pensons que ce modèle est inefficace dans des réseaux où la majorité du trafic est de type temps réel. Cependant, il permet un passage à l'échelle, car les nœuds intermédiaires n'ont pas à maintenir d'états sur les flux. Un autre inconvénient est

qu'un flux temps réel accepté peut faire face à des violations dans sa bande passante, voire une interruption totale de service. Ainsi, le mécanisme de contrôle local du débit de trafic *best effort* semble insuffisant pour fournir la qualité de service aux applications temps réel.

PRTMAC (*Proactive Real Time MAC*) de [Vishnumurthy *et al.* (2004)] offre un support pour le trafic temps réel et pour la différenciation de service dans des réseaux où la mobilité est trop élevée et les ressources (énergie et batteries) sont toujours disponibles et ne constituent pas une contrainte majeure comme dans le cas des réseaux militaires. Cependant, pour des nœuds où les contraintes énergétiques constituent une ressource rare, le fait d'avoir un autre canal de contrôle comme celui utilisé dans PRTMAC peut ne pas être une solution idéale.

2.6 Conclusion et perspectives

Après avoir vu toutes les contraintes des réseaux *Ad hoc*, il est clair que l'application des solutions des réseaux classiques doit être repensée. Plusieurs domaines semblent incontournables et divers groupes de recherche tentent de trouver des solutions adéquates. Il est à noter qu'au niveau des contraintes matérielles, des nœuds à forte autonomie énergétique et ergonomique doivent être développés pour stimuler l'évolution de ces réseaux.

La conception d'un protocole de routage performant en termes de délai, de perte de paquets et de stabilité des routes devient une nécessité. Rajaraman (2002) présente les caractéristiques d'un bon protocole de routage. Il doit être extensible (*scalable*) et présenter un trafic de contrôle négligeable afin de ne pas influencer le trafic de données.

De nombreux protocoles de routage ont été synthétisés dans [Adibi et Erfani (2006)]. Très souvent, ces protocoles utilisent massivement la diffusion d'information dans tout le réseau. Nous pensons qu'un bon protocole de routage devrait limiter les diffusions, les collisions et la consommation de ressources radio. En effet, dans une inondation aveugle, un terminal qui

reçoit un paquet à diffuser le relaye s'il ne l'a pas déjà reçu. Ainsi, l'intégralité des nœuds devrait le recevoir. Cependant, Tseng *et al.* (2002) ont démontré qu'un tel mécanisme occasionne des problèmes de stabilité (à cause des collisions) et de redondance (beaucoup de transmissions sont inutiles). Ce problème est connu sous le nom de tempête d'inondation (*Broadcast Storm Problem*).

IP représente *de facto* un standard, l'intégration des réseaux *Ad hoc* avec l'Internet doit se faire d'une façon robuste face aux fautes et aux incohérences de sorte à gérer efficacement la micro et macro-mobilité, l'attribution de préfixes routables, etc. Ceci permet de passer ces réseaux à l'échelle et de sortir du cadre de routage interne d'un réseau *Ad hoc*. Cette intégration exige une sorte d'autoconfiguration pour s'adapter à l'environnement ainsi que des solutions robustes de configuration d'adresses, d'apprentissage des paramètres du réseau tels que son préfixe et son serveur DNS (*Domain Name Server*).

Pour la qualité de service, Reddy *et al.* (2006) ont présenté une revue de tous les protocoles tant au niveau MAC qu'au niveau réseau. Il est primordial de pouvoir établir des priorités entre les flux, limiter les pertes de paquets vitaux pour la gestion du réseau, ou du moins en restreindre l'impact. En fait, Chaudet *et al.* (2005) ont montré que la couche MAC du standard IEEE 802.11 présente des dysfonctionnements importants dans un réseau multisauts comme *Ad hoc*. Une nouvelle couche MAC et de nouvelles techniques de modulations doivent donc être conçues afin de garantir des débits plus acceptables sur des rayons de portée plus étendus et de permettre le transfert du multimédia, une des applications les plus à la mode de nos jours.

Santivanez *et al.* (2002) ont montré qu'aucun des protocoles de routage ne permet pas le déploiement à grande échelle du réseau *Ad hoc*. Une sorte d'interaction entre les différentes couches (*Cross-Layer*) semble une solution prometteuse. Les modèles classiques des réseaux filaires supposent une indépendance entre les couches protocolaires afin d'optimiser leur flexibilité : routage et fonctions MAC sont par exemple indépendants. Cependant, Raisinghani et Iyer (2006) ont montré qu'une telle indépendance est communément

considérée comme trop coûteuse pour les réseaux sans fil. Conséquemment, une coopération *Cross-Layer* achevant un compromis entre performance et flexibilité doit être mise en œuvre.

Du côté sécurité, des mécanismes d'authentification, de confidentialité et d'intégrité doivent être mis en place au sein d'une communauté d'utilisateurs. Yang *et al.* (2004) ont synthétisé les défis et les solutions. Cependant, tout mécanisme de sécurité doit être assez flexible pour rendre les communautés semi-perméables, en fonction du niveau de confiance requis pour les échanges.

Dans le chapitre suivant, nous évoquerons notre vision consistant à hiérarchiser un réseau *Ad hoc* dans l'objectif de calquer certaines approches utilisées dans les réseaux filaires classiques, simplifiant ainsi le déploiement à grande échelle de toute solution proposée. Nous pensons que cette hiérarchisation permet de simplifier le développement de protocoles pour les réseaux *Ad hoc*, et surtout d'optimiser leurs performances.

CHAPITRE 3

PROPOSITION DE CLUSTÉRISATION DES RÉSEAUX *AD HOC*

3.1 Introduction

Nombreuses sont les approches qui ont été proposées pour faire un contrôle efficace de la topologie dans les réseaux *Ad hoc*. En effet, ce contrôle requiert une structuration pertinente des nœuds dans le réseau afin de réduire leur complexité et d'intégrer quelques fonctionnalités et solutions des réseaux filaires.

Des auteurs ont donc proposé la création dynamique des dorsales dans un réseau *Ad hoc*, servant de collecteur et de diffuseur de trafic. Ces dorsales ont montré de bons avantages en formant des structures assez robustes. Ainsi, les nœuds d'un réseau *Ad hoc* peuvent représenter les feuilles d'une dorsale qui elle-même peut être organisée hiérarchiquement. Une dorsale doit être stable dans le sens où les nœuds la constituant doivent rester inchangés pendant un laps de temps suffisant. Le fait de créer une structure redondante dans la dorsale permettrait ainsi d'optimiser sa robustesse ainsi que sa durée de vie.

D'autres auteurs ont proposé la formation des clusters qui eux, introduisent une hiérarchie dans la topologie du réseau, découpant le réseau en plusieurs zones de service. Ce découpage d'un réseau étendu permet de simplifier les problématiques d'adressage, de routage, et d'agrégation du trafic. Les clusters ayant des nœuds-chefs qui orchestrent localement la zone, forment une topologie virtuelle utile servant à agréger les flux et à cacher localement tous les changements topologiques. Les clusters peuvent servir pour implémenter des solutions de routage comme mentionné dans la section 2.3.3.4 du chapitre 2, de qualité de service ainsi que des solutions pour la couche MAC [Hou *et al.* (2001)].

Dans ce chapitre, nous allons tout d'abord aborder des thèmes et des sujets de grand intérêt dans le cadre de la structuration des réseaux *Ad hoc*. Ces derniers sont actuellement au cœur

de travaux de recherche très approfondis et font l'objet de nombreuses publications. Notant que nous présenterons brièvement les approches basées sur les dorsales et nous détaillons celles basées sur les clusters étant donné que notre proposition se base sur des structures en clusters. Par la suite, nous évoquerons notre vision de clustérisation du réseau, nous détaillons le mécanisme de formation des clusters et nous proposons un modèle analytique pour modéliser les différents paramètres qui nous serviront à implémenter un contrôle d'admission sur les *clusterhead* en vue de balancer la charge sur plusieurs clusters.

3.2 État de l'art

3.2.1 Dorsales virtuelles (*backbones*)

Les dorsales ont largement été utilisées dans les réseaux filaires. Cependant, les caractéristiques des réseaux *Ad hoc*, créent des complications quant à leur intégration sur des topologies mobiles et très changeantes. Une dorsale consiste en un regroupement de nœuds dominants qui auront la responsabilité d'acheminer les paquets et constituant un CDS (*Connected Dominating Set*). Tout nœud ordinaire (dominé) doit donc être connecté à la dorsale pour lui envoyer le trafic à diffuser. Toutefois, un nombre optimal de dominants formant la dorsale doit être construit afin de réduire le nombre de nœuds diffusant les paquets et de minimiser la charge du réseau.

La détermination de l'ensemble de domination de taille minimale MCDS (*Minimum Connected Dominating Set*) est la clé de construction d'une telle dorsale CDS. Cependant, ceci n'est pas une chose aisée, le fait de déterminer tel ensemble est un problème *NP-Complet* [Das et Bharghawan (1997)]. La figure 3.1 illustre un exemple de construction de l'ensemble CDS, les nœuds foncés représentent les nœuds dominants de la dorsale.

Chen et Liestman (2002) ont proposé un algorithme distribué WCDS (*Weakly Connected Dominating Set*) en colorant à chaque étape un nœud situé à au plus deux sauts d'un dominant. Ensuite, plusieurs pièces seront créées et fusionnées ensemble. Cependant, cet

algorithme nous semble non robuste à cause de sa grande quantité de trafic de contrôle et de son temps de convergence très long.

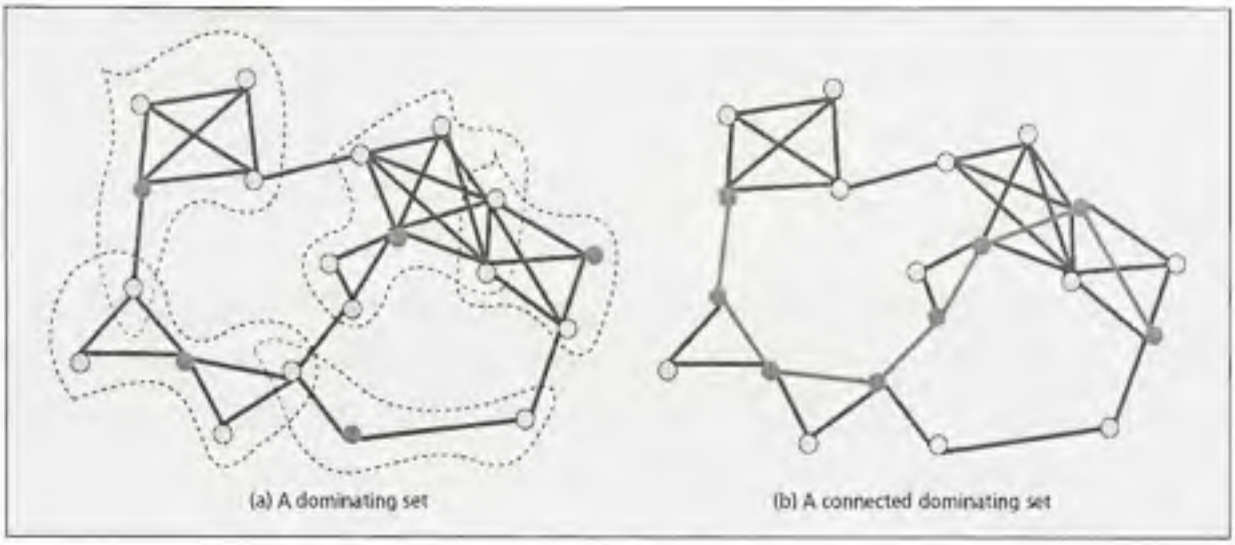

Figure 3.1 ***Exemple de construction d'une dorsale virtuelle.***

(Tiré de Yu et al., 2005)

Source : Cette figure a été tirée de Yu et *al.* 2005 « A SURVEY OF CLUSTERING SCHEMES FOR MOBILE AD HOC NETWORKS » et correspond à la « Figure 2 Dominating set illustration » présentée en page 4 dans le document original. (La référence complète du document est présentée dans la liste de références).

Une multitude d'approches de détermination de MCDS existe dans la littérature. Ces approches peuvent être centralisées ou distribuées. Il est à noter qu'il existe d'autres approches comme celles de [Rubin *et al.* (2003), Meraihi *et al.* (2004)] qui essayent de calquer la hiérarchie des réseaux filaires classiques en sélectionnant certains nœuds BCN (*Backbone Capable Node*) pouvant avoir un large rayon de portée radio, ces nœuds agiront comme une dorsale virtuelle (*virtual backbone*) dans le réseau. Les autres nœuds non BCN agiront en tant que clients de cette dorsale. Nous pensons que le fait de classifier les nœuds d'un environnement hétérogène semble une solution assez puissante pour faciliter le déploiement des réseaux *Ad hoc*.

3.2.1.1 Algorithmes centralisés

Wu et Li (2001) ont présenté une solution simple de construction centralisée d'un CDS où chaque nœud commence par diffuser des *hello*. Un nœud est élu dominant s'il possède au moins deux de ses voisins n'étant pas connectés par un lien radio. Le nœud est dominé dans le cas contraire. L'inconvénient de ce mécanisme est qu'il génère un CDS de cardinalité élevée dû au nombre élevé de dominants élus.

Pour éviter cet inconvénient et réduire la taille de la dorsale, Wu (2003) propose de modifier le statut d'un dominant u à un dominé si un des voisins de u ayant un *id* plus grand est aussi dominant et couvre tous les dominés de u, ou bien si deux des voisins de u ayant des *id* plus grands sont aussi dominants et couvrent tous les dominés de u. Les auteurs dans [Carle et Simplot-Ryl (2004)] considèrent qu'un nœud est couvert si un sous ensemble de ses voisins forme un ensemble connexe et dominant de son voisinage.

Dans le but d'optimiser la diffusion des informations, d'autres approches dont celle de [Stojmenovic *et al.* (2001)] proposent la création d'une dorsale CDS orientée source. Dans ces approches, c'est l'ensemble des nœuds-relais qui formera le CDS : à la réception d'un paquet à relayer, un nœud u initialise un temporisateur durant lequel il continue à surveiller les retransmissions de ses voisins. Lorsqu'un voisin v relaie le paquet, u considère tous les voisins de v comme couverts. S'il existe d'autres voisins non couverts après l'expiration de ce temporisateur, u relaie le paquet. Sinon, u peut supprimer le paquet, car une transmission serait inutile.

Toutefois, dans le cas d'une dorsale CDS orientée source, si le paquet provient d'un nœud différent, nous pensons que les nœuds-relais ont une forte probabilité de changer. Ceci provoquerait beaucoup de changements dans la dorsale, chose qui n'est pas souhaitée dans les réseaux *Ad hoc*.

3.2.1.2 Algorithmes distribués

Les algorithmes distribués requièrent une coordination entre les nœuds du réseau afin d'élire un ensemble dominant CDS, pour l'interconnecter tout en limitant le nombre de dominés à colorer en dominants. Les algorithmes de construction sont très souvent similaires et l'élection se base sur un poids attribué aux nœuds. Ce poids peut prendre la valeur du degré comme dans [Ryu *et al.* (1999), Liang et Haas (2000), Butenko *et al.* (2003)], de l'identifiant du nœud ou d'un poids non explicité comme dans [Chen et Liestman (2002)].

Un nœud est caractérisé par l'un des quatre états suivants : dominant (appartenant au CDS), dominé (voisin d'un dominant), actif (en phase d'élection) et isolé (en phase d'attente). Le nœud actif qui a le plus fort « poids » parmi ses voisins actifs devient dominant, ses voisins deviennent ses dominés. Par la suite, les nœuds dominés obligeront d'autres nœuds à devenir actifs.

Ryu *et al.* (1999) supposent que les nœuds doivent envoyer périodiquement des messages *hello*, inondés par la dorsale. Si un nœud n'entend pas un voisin dominant relayer ces messages issus d'un de ses voisins, alors il devient dominant. L'inconvénient de cette approche est l'inondation périodique des messages *hello*. Dans [Bao et Aceves (2003)], chaque dominant doit, en utilisant les informations de ses dominés, essayer de s'interconnecter à tous les dominants qui sont à moins de trois sauts dans son voisinage. Kozat *et al.* (2001) proposent une interconnexion commandée par le dominant si deux dominants sont éloignés par au plus deux sauts, et une interconnexion commandée par un dominé si deux dominants sont éloignés de trois sauts. La redondance introduite par cette dernière interconnexion semble importante. [Wang et Li (2006)] étendent ces approches en prenant en considération des chemins d'interconnexion pondérés.

Liang et Haas (2000) proposent de construire des bases de données distribuées. Un ensemble nommé *r-dominant* est préalablement construit de façon centralisée. Chaque base de données envoie des messages de contrôle à $(2r+1)$ sauts pour découvrir les autres

dominants et établir des liens virtuels vers chacun d'eux. Basagni *et al.* (2001) utilise une méthode similaire, mais la distance entre un nœud et la dorsale doit être fixée à un seul saut.

D'autres approches supposent que dans le cas où ce sont seulement les nœuds isolés avoisinants d'un dominé qui deviennent actifs, il suffit que deux dominants éloignés à moins de 2 sauts l'un de l'autre se connectent pour établir l'ensemble CDS. Dans ce contexte, [Butenko *et al.* (2003)] proposent des approches basées sur un message d'exploration itérative initié par un nœud-chef dans le réseau. Ce message permet de créer un arbre d'interconnexion entre les dominants du réseau. Min *et al.* (2004) proposent une approche d'exploration similaire, mais en séparant la construction de l'ensemble CDS de son interconnexion. Dans [Cheng et Du (2002)], un dominant récemment élu colorie un de ses voisins dominés en dominant.

Il existe également des approches qui supposent qu'un dominant peut se situer à exactement trois sauts de tout autre dominant. [Alzoubi *et al.* (2002), Wan *et al.* (2004)] utilisent des messages d'invitation, initiés par le nœud-chef, à propager dans le réseau. Un dominant s'interconnecte au CDS via le premier message d'invitation reçu. [Lin *et al.* (2003)] proposent un algorithme quasi centralisé pour maintenir une interconnexion via la construction d'un arbre de *Steiner*. Cependant, cet algorithme nécessite un trafic de contrôle très lourd.

Dans sa thèse de doctorat, [Mnif (2006)] décompose le problème en deux étapes. Il commence par déterminer un ensemble de domination connexe basé sur une formulation en programmation linéaire. Par la suite, il trouve l'arbre de recouvrement de cet ensemble afin de déduire l'ensemble MCDS. L'auteur propose également une procédure de maintenance distribuée et assez élaborée avant de valider ses propositions par des simulations montrant que le réseau peut passer à l'échelle même en cas de forte mobilité. En outre, il démontre une amélioration des performances de routage de l'ordre de 20 % lorsqu'il applique les protocoles de routage en présence de sa dorsale.

3.2.2 Clusters

La clustérisation d'un réseau est son découpage en zones de diamètre défini. Formellement, chaque cluster comprend un chef (*clusterhead*). Ce découpage permet donc, de créer des zones de services, avec des points d'accès virtuels (chefs), pouvant gérer l'adressage, la localisation, l'économie d'énergie, la réservation de ressources, l'attribution de fréquences et la distribution du trafic de contrôle.

Par convention, un *k-clusters* oblige tout nœud à être à moins de *k* sauts d'un *clusterhead* (voir figure 3.2). La clustérisation cherche à optimiser plusieurs paramètres pour atteindre de meilleures performances. Parmi les paramètres que les auteurs essayent d'optimiser, nous retrouvons : le nombre de clusters élus, le trafic de contrôle généré lors de la formation des clusters, le mécanisme de réélection, la stabilité des clusters formés, etc. Pour plus de détails, nous référons le lecteur aux travaux de [Yu et Chong (2005), Wei et Chan (2006)] qui synthétisent les différents algorithmes proposés dans la littérature.

À l'époque, [Ephremides *et al.* (1987)] étaient les premiers à introduire l'idée de la clustérisation dans les réseaux mobiles. Dans la terminologie de théorie des graphes, un ensemble minimal de *clusterhead* dans un réseau *Ad hoc* est précisément un ensemble de domination de taille minimale. Alors qu'il existe de nombreux algorithmes de *1-clusters*, ceux adaptés aux *k-clusters* sont relativement rares et moins déployés. La plupart de ces algorithmes exigent une exécution sur la topologie tout entière même dans le cas de petits changements. Ceci nécessite beaucoup de bande passante et de ressources qui sont considérées rares dans les réseaux radio et pourrait tarder la transmission des données.

Basé sur des simulations, [Vuong et Huynh (2000)] ont montré que le problème d'adaptation d'un ensemble de domination minimal *d-hop* dans une topologie *Ad hoc* simple est un problème *NP-Complet*. Les auteurs ont ensuite proposé un algorithme pour restreindre les calculs justement autour du lieu des changements, ce qui permet de minimiser le temps de calcul et de reconstruction de l'arbre tout entier.

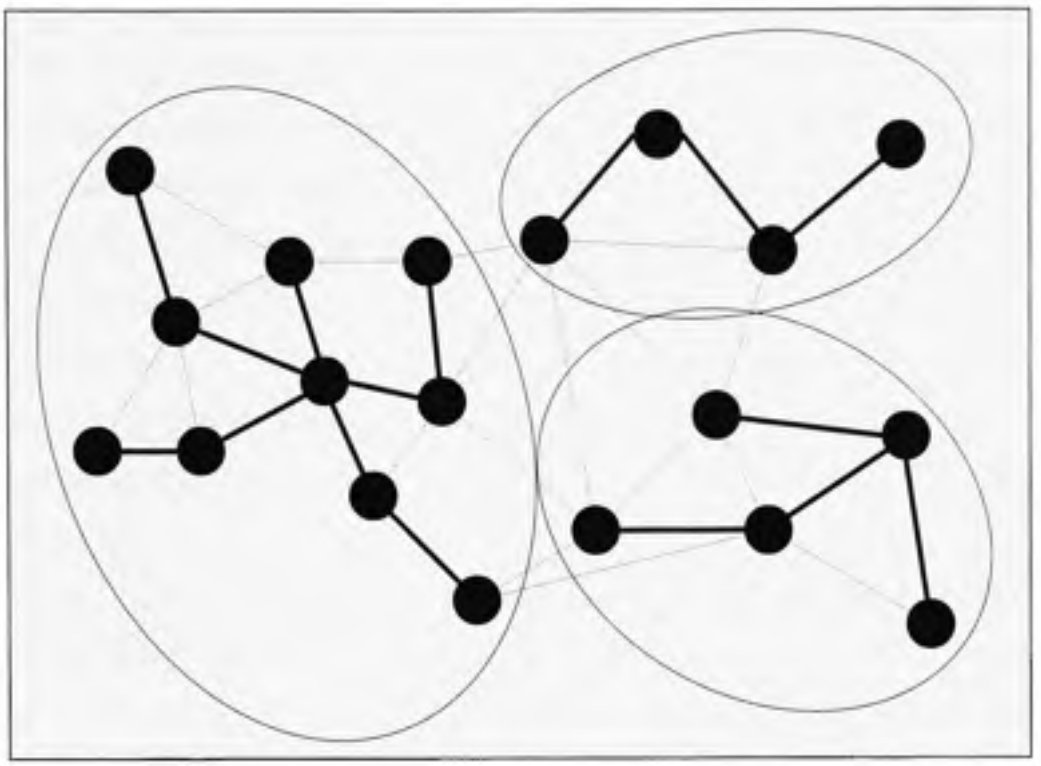

Figure 3.2 ***Exemple de 2-clusters; les nœuds non foncés représentent les clusterhead.***

Dans ce qui suit, nous détaillons notre classification des algorithmes proposés dans la littérature tout en décortiquant le niveau de complexité, les avantages et les inconvénients de chacun d'eux.

3.2.2.1 Algorithmes basés sur une maintenance modérée

Parekh (1994) a proposé l'algorithme MCC (*Maximum Connectivity Clustering*) basé sur le degré de connectivité d'un nœud. Le degré d'un nœud représente le nombre de voisins de ce nœud. Le nœud est élu *clusterhead* (CH) s'il détient le plus fort degré. [Gerla et Tsai (1995)] utilisent une approche similaire nommée *Highest Degree*. L'inconvénient de ces approches est l'instabilité du degré d'un nœud, ce qui provoque beaucoup de changements sur les CH. De plus, le CH gérant un grand nombre de nœuds va décharger rapidement sa batterie et perdre ses capacités. Le niveau de performances des clusters décroit radicalement quand le nombre de nœuds dans le cluster commence à croître.

[Lin et Gerla (1997)] ont proposé l'algorithme *Lowest-id* où la sélection du *clusterhead* est basée sur un *id* unique. Les performances en termes de débit sont meilleures que dans le cas du MCC et *Highest Degree*. Toutefois, les nœuds ayant le plus petit *id* souffrent d'une surcharge volumineuse réduisant ainsi la durée de vie du système. *Lowest-id* forme des clusters ayant un diamètre maximal de 2 sauts et fonctionne de la façon suivante :

- chaque nœud a un *id* différent et envoie périodiquement en *broadcast* la liste de tous ses voisins (incluant soi-même);
- un nœud qui reçoit seulement des *ids* plus grands que le sien devient CH;
- le nœud ayant le plus petit *id* devient son CH, à moins que ce nœud renonce à son rôle en tant que CH;
- un nœud qui détecte deux ou plusieurs CH est un nœud passerelle. Dans les autres cas, il est nœud ordinaire.

[Chiang *et al.* (1997)] ont proposé LCC (*Least Clusterhead Change*), une variante de l'algorithme *Lowest-id*. LCC est composé de deux étapes : l'étape de clustérisation et l'étape de maintenance. Dans LCC, si un membre ($id = i$) d'un cluster C se déplace vers un autre cluster qui a un *id* plus grand, alors il ne faut pas ré-exécuter la clustérisation à moins que i soit le *clusterhead* de C. Ceci aide à réduire le nombre de changements des CH causés par la ré-clustérisation fréquente. Toutefois, plusieurs réélections auront lieu à cause de la propagation de ces changements dans tout le réseau, engendrant ainsi un overhead important et inévitable.

[Lian *et al.* (2007)] ont modélisé un cadre pour étudier les performances de différentes approches de clustérisation dans les réseaux *ad hoc*. En se basant sur une étude analytique, les auteurs ont clairement démontré que *Lowest-id* est plus performant que MCC en termes de bande passante utilisée pour l'échange des messages de contrôle et de routage.

[Amis *et al.* (1999)] ont décrit un algorithme complexe nommé *Max-Min* pour former des *k-clusters* en 2 phases. La première phase exige que chaque nœud, durant l'essai i, relaye le plus grand identifiant à k sauts, découvert durant l'essai $(i-1)$ et ce jusqu'à ce que i soit

égal à *k*. La deuxième phase qui déroule également sur *k* essais a pour objectif de prévenir les nœuds qui ont été élus *clusterhead*. Dans ce cas, les nœuds relayent les plus petits identifiants découverts durant l'essai précédent. Finalement, un nœud se déclare *clusterhead* s'il entend son identifiant propagée durant la deuxième phase. Toutefois, les auteurs ne proposent aucune procédure de maintenance et l'exécution périodique de l'algorithme nous semble également très gourmande en termes de capacités de traitement et de trafic de contrôle engendré.

[Kwon et Gerla (2002)] ajoutent un champ dans un paquet et construisent des clusters par l'inondation de ce dernier. Le premier nœud qui relaye ce paquet devient *clusterhead*. Les nœuds voisins deviennent membres. Tout autre nœud qui entend plusieurs *clusterhead* est un nœud passerelle, ce dernier relaye un paquet d'inondation selon le nombre de *clusterhead* et de passerelles qu'il entend. Toutefois, plus d'*overhead* est engendré en ajoutant un autre champ dans les paquets à inonder. Cette approche étant orientée source, elle ne présente aucune stabilité dans le temps et les clusters formés peuvent changer constamment.

[Fernandess et Malkhi (2002)] commencent par la construction d'un arbre de recouvrement. Toute branche de cet arbre possédant une hauteur de *k* sauts sera étayée pour former un cluster. Toutefois, les auteurs ne proposent aucun mécanisme de maintenance. La construction et la maintenance de cet arbre engendrent un trafic de contrôle important et un temps de convergence supplémentaire.

Dans [Yu et Chong (2003)], les auteurs proposent 3hBAC (*3-hop Between Adjacent Clusterheads*) pour construire des *1-clusters* où les CH voisins sont à trois sauts l'un de l'autre. Dans un premier temps, le nœud ayant le plus fort degré est élu CH et forme le premier cluster. Ensuite, la formation des autres clusters se fait en paralléle dans tout le réseau. Les auteurs ont introduit la notion des nœuds *clusterguest* qui sont des nœuds ne pouvant être connectés à aucun CH, mais pouvant accéder à quelques clusters avec l'aide des nœuds membres. L'inconvénient de cette approche est qu'une hypothèse de stationnarité

doit être considérée au démarrage de la clustérisation afin de décider du premier cluster dans le réseau.

3.2.2.2 Algorithmes basés sur la mobilité

Les algorithmes *Lowest-id* et LCC ne prennent pas en considération le facteur mobilité où il est possible d'élire un nœud en tant que CH même si ce nœud se déplace rapidement vers la frontière du cluster, ou même si ce nœud risque de perdre sous peu son énergie. Dans ce cas, le nœud ayant le plus petit *id* doit renoncer à son rôle et le mécanisme de ré-clustérisation se déclenche très fréquemment, une chose qui n'est pas souhaitable dans les réseaux *Ad hoc*.

[Johansson *et al.* (1999)] ont proposé une métrique de mobilité géométrique pour capturer et quantifier les mouvements des nœuds. La métrique de mobilité entre n'importe quelle paire des nœuds est définie en mesurant leurs vitesses moyennes dans un intervalle de temps. Dans le but de globaliser la métrique de mobilité sur toute la topologie, cette métrique est calculée entre toutes les paires des nœuds existants dans le réseau. Toutefois, ce schéma suppose l'existence d'un GPS pour calculer les vitesses des nœuds, une chose qui n'est pas toujours offerte sur tous les appareils *Ad hoc*. Un autre inconvénient est que la métrique de mobilité offerte par le GPS est globale, ceci ne reflète pas le facteur de mouvement local des nœuds dans le voisinage d'un nœud particulier (mobilité relative), ce facteur étant la cause fondamentale des changements des *clusterhead*.

Le modèle RPGM (*Reference Point Group Mobility Model*) proposé dans [Hong *et al.* (1999)] peut être utilisé pour la prédiction de la mobilité d'un groupe. Dans ce modèle, chaque groupe a un nœud central et le mouvement de ce nœud définit le mouvement global de tout le groupe. Dans ce contexte, il serait intéressant de parler du protocole de routage ABR (*Associativity Based Routing*) qui utilise la mobilité pour faire le routage. En fait, ABR ne définit pas la métrique de mobilité, mais il utilise une métrique de stabilité de route. Une fois la route est perdue, ABR utilise un schéma de routage où la route est sélectionnée parmi les nœuds qui ont des états d'associativité ayant une période de stabilité.

[McDonald et Znati (1999)] ont proposé un cadre pour l'organisation dynamique des nœuds mobiles dans des clusters, mais ils se sont concentrés sur les caractéristiques mathématiques de la probabilité de disponibilité d'un chemin. Cependant, cette probabilité ne permet pas de capturer la mobilité relative d'un nœud quelconque par rapport à ses voisins. Donc cette approche ne fournit pas une métrique adéquate dans le cas de clustérisation à moins de 2 sauts.

Les auteurs ont plus tard proposé DDCA (*Distributed Dynamic Clustering Algorithm*) [McDonald et Znati (2001)] pour essayer de construire des *k-clusters* en se basant sur un critère (α, t) qui indique que chaque nœud appartenant à un cluster aura un chemin vers tout autre nœud dans une période de temps t avec une probabilité $\geq \alpha$ quelque soit la distance entre eux. Les valeurs de α et t sont calculées selon le modèle mathématique proposé dans [McDonald et Znati (1999)]. DDCA est un algorithme de clustérisation distribué et non périodique où un nœud peut joindre un cluster s'il existe un chemin entre lui et le CH satisfaisant le critère (α, t). Dans le cas où il n'existe aucun chemin, le nœud doit former un nouveau cluster auquel d'autres nœuds peuvent adhérer. DDCA propose également un mécanisme d'adaptation de la taille des clusters en se basant sur le paramètre α de façon à former de petits clusters pour des réseaux peu mobiles et de larges clusters pour des réseaux très mobiles.

[An et Papavassiliou (2001)] supposent l'existence d'un système de positionnement (GPS) afin de créer des clusters homogènes en termes de mobilité. Ainsi, les nœuds qui ont une faible mobilité relative peuvent appartenir au même cluster. Inversement, deux nœuds ayant une trajectoire différente risquent de former des clusters disjoints. Toutefois, il n'est pas toujours possible d'intégrer des GPS sur la majorité des terminaux *Ad hoc* actuels.

[Basu *et al.* (2001)] ont proposé MOBIC (*Mobility Based Metric for Clustering in Mobile Ad hoc Networks*) pour former des *1-clusters*. MOBIC calcule la métrique de mobilité en capturant la mobilité dans le voisinage de chaque nœud qui consiste à mesurer le niveau d'énergie reçue sur un nœud de la part de tous les voisins à un saut de lui. En réalité, un

calcul exact de la distance entre un transmetteur et un récepteur peut ne pas être obtenu à partir de la mesure de l'énergie des signaux, mais plutôt à partir du niveau d'énergie reçue entre deux transmissions successives (deux *hello* consécutifs) avec le nœud voisin. De cette façon, il serait possible d'avoir une idée précise sur la mobilité relative entre deux nœuds. Les simulations présentées par les auteurs montrent que MOBIC s'avère beaucoup plus stable pendant la formation des clusters que les algorithmes précédents et que le taux de changement des clusters diminue de 33 %.

Dans MOBIC, la métrique de mobilité relative d'un nœud Y par rapport à un nœud X est définie de la façon suivante :

$$M_Y(X) = 10log_{10}\frac{Received\ Power\ {}^{new}_{X\to Y}}{Received\ Power\ {}^{old}_{X\to Y}} \tag{3.1}$$

Si $Received\ Power\ {}^{new}_{X\to Y} < Received\ Power\ {}^{old}_{X\to Y}$, alors $M_Y(X) < 0$ et la métrique de mobilité entre deux nœuds indique que ces deux nœuds s'éloignent l'un de l'autre. Dans le cas contraire, les deux nœuds se rapprochent l'un de l'autre.

Les auteurs ont généralisé cette règle sur un ensemble de voisins. Pour un nœud ayant m voisins, le nœud recevra m valeurs $M_Y(X)$. Dans ce cas, il faut calculer une valeur de mobilité locale agrégée en utilisant la variance de tout l'ensemble. Ainsi :

$$\boxed{M_Y = VAR\{M_Y(X_j)\}_{j=1}^{m} = E[(M_Y)^2]} \tag{3.2}$$

Une grande valeur de M_Y indique que Y est très mobile par rapport à ses voisins X_i. Par contre, une petite valeur indique que Y est presque fixe par rapport à ses voisins, ce qui implique une élection potentielle de ce nœud en tant que *clusterhead*.

3.2.2.3 Algorithmes basés sur le contrôle d'énergie

Pour éviter qu'un *clusterhead* épuise rapidement ses capacités, une autre métrique a été introduite. Cette métrique prend en considération le temps durant lequel un nœud a joué le rôle de *clusterhead* ainsi que le niveau de batterie des nœuds. Ceci permet en quelque sorte d'équilibrer la charge lors du manque d'énergie sur un nœud quelconque. À titre d'exemple, nous citons les approches de [Amis et Prakash (2000), Ryu *et al.* (2001), Safwat *et al.* (2001), Wu *et al.* (2002), Sheltami et Mouftah (2003), Alipour *et al.* (2007), Choi *et al.* (2008a)].

En effet, Safwat *et al.* (2001) ont proposé une approche nommée PA-VBS (*Power-Aware Virtual Base Stations*) qui utilise la batterie des nœuds pour former les clusters. Suite à des simulations, les auteurs démontrent comment les nœuds peuvent optimiser l'utilisation de leurs batteries et comment se fait l'équilibrage de charge sur plusieurs chemins entre une paire de nœuds communiquants.

[Sheltami et Mouftah (2003)] ont également proposé un modèle similaire nommé WEAC (*Warning Energy Aware Clusterhead*). Les auteurs s'appuient sur des simulations pour montrer les caractéristiques et les performances de leur modèle dans le cas des larges réseaux.

WEAC améliore aussi les performances de tous les algorithmes de formation des clusters en sauvegardant l'énergie des stations mobiles au sein du cluster. La solution consiste à mesurer et comparer le BPL (*Battery Power Level*) des différentes stations. Dès qu'un *clusterhead* devient saturé (l'atteinte d'un seuil déterminé), WEAC évite la surcharge de sa batterie en évitant l'acceptation d'autres requêtes, Ce qui permet aux stations desservies par ce *clusterhead* de chercher un autre cluster avoisinant.

Toutefois, nous pensons que ce type d'algorithmes, étant basé sur l'optimisation de l'énergie des nœuds, est mieux adapté aux réseaux capteurs qui ont des contraintes

énergétiques très fortes. Par contre, dans le cas où les nœuds sont très mobiles, les algorithmes basés sur le contrôle d'énergie ne tiennent pas compte des facteurs de mobilité et de stabilité des nœuds.

3.2.2.4 Algorithmes basés sur un équilibrage de charge

Ce type de clustérisation tente d'optimiser le nombre de nœuds qu'un CH peut gérer afin de ne pas dégrader les performances. Un cluster ayant un grand nombre de nœuds peut engendrer une charge énorme, créant ainsi des goulots d'étranglement sur les CH et réduisant le débit total du système. Un petit cluster, n'ayant que quelques nœuds, engendre plusieurs clusters à former dans le réseau, allongeant ainsi les routes hiérarchiques et augmentant le délai de bout en bout du système. La ré-clustérisation doit être exécutée afin d'ajuster le nombre de nœuds par cluster une fois que sa taille dépasse un certain seuil.

Amis et Prakash (2000) ont proposé DLBC (*Degree Load Balanced Clustering*) qui exécute périodiquement la clustérisation afin de garder un nombre de nœuds optimal ED (*Elected Degree*) par cluster. Dans le cas où la différence entre ED et le nombre de nœuds actuellement desservis par le CH excède une certaine valeur *Max_Delta*, le CH doit résilier son rôle et devenir un nœud ordinaire. Ceci permettrait de faire en sorte que tous les CH élus servent le même nombre de nœuds dans le réseau. Toutefois l'élection des CH, étant basée sur le degré des nœuds, ceci provoque à notre avis une exécution fréquente de la ré-clustérisation à cause des variations dynamiques et continues du degré des nœuds. Un autre inconvénient est que le choix des paramètres ED et *Max_Delta* n'a jamais été justifié dans DLBC.

Bannerjee et Khuller (2001) proposent la formation des *k-clusters* limités par le nombre de participants dans chaque *k-cluster*. En premier temps, les auteurs présupposent l'existence d'un arbre de recouvrement optimal. Sur cet arbre, l'algorithme étaye les branches ayant entre k et $2k$ nœuds afin de construire des clusters de taille comprise entre k et $2k$. Les auteurs proposent une procédure de maintenance, permettant de fusionner les clusters trop

petits ou au contraire de partitionner ceux trop denses. Cependant, la construction d'un tel arbre est très coûteuse en termes de trafic de contrôle et de temps de convergence.

Turgut *et al.* (2003) proposent une approche basée sur la programmation linéaire et utilisant l'ensemble des *clusterhead* possibles dans le réseau afin de minimiser la différence de cardinalité entre les différents clusters. Cependant, la façon d'adapter telle optimisation à un environnement distribué n'a pas été décrite dans cette approche.

Ohta *et al.* (2003) proposent AMC (*Adaptive Multi-hop Clustering*) pour former des *k-clusters* tout en essayant d'équilibrer la charge. Les auteurs ne spécifient pas comment les clusters sont initialement construits. Pourtant, ils décrivent la procédure de maintenance où chaque nœud doit diffuser au sein de son propre cluster son *id*, son *cluster-id*, et son statut (CH, membre, passerelle). Chaque passerelle doit également communiquer avec celles des autres clusters afin de fournir des mises à jour à son CH. AMC intègre des procédures de fusion et de partition des clusters pour ajuster la taille des clusters en respectant des seuils définissant la taille minimale et maximale d'un cluster dans le réseau. Le fait de ne pas justifier le choix de ces seuils semble un inconvénient majeur à considérer.

3.2.2.5 Algorithmes basés sur le poids des nœuds

Lowest-id a été généralisé par des techniques basées sur l'attribution d'un poids unique aux nœuds au lieu de l'utilisation des *id*. Ce poids est paramétré selon les caractéristiques et les conditions des nœuds mobiles. Les algorithmes DCA (*Distributed Clustering Algorithm*) de Basagni (1999a) et DMAC (*Distributed Mobility Adaptive Clustering algorithm*) de Basagni (1999b) en sont des exemples. Le nœud ayant le plus grand poids dans son voisinage, à une distance de moins de deux sauts, sera élu CH, sinon il doit joindre un cluster existant.

DCA suppose que la topologie est fixe durant la clustérisation. Par contre, DMAC adopte une topologie changeante, répondant ainsi aux changements fréquents dans un réseau *Ad hoc*. Un des inconvénients de ces deux mécanismes est que la façon ambiguë à laquelle les

poids sont attribués aux nœuds n'a jamais été discutée et qu'il n'existe aucun moyen pour optimiser les paramètres du système tels que le débit, le contrôle de puissance, etc.

Bettstetter (2004) a étudié la densité des clusters et en particulier celle des clusters construits par DMAC. Deux approches ont été présentées. La première est basée sur des simulations et la deuxième est basée sur des calculs analytiques. Sur une surface quelconque, le processus d'arrivée et de distribution des nœuds est un processus de Poisson aléatoirement distribué avec un paramètre λ (en nombre de nœuds par unité de surface). Si μ représente le nombre prévu de voisins d'un nœud. Alors la densité du cluster définie comme étant le nombre prévu de *clusterhead* par unité de surface est exprimé de la façon suivante :

$$\rho = \frac{\lambda}{1+\frac{\mu}{2}} \tag{3.3}$$

En utilisant cette formule, il devient possible d'estimer la complexité de la hiérarchie et par suite le nombre de *1-clusters* à obtenir sur une surface déterminée. Certes, le nombre de CH élus a tendance à diminuer quand le degré augmente.

Chen *et al.* (2002) construit des *k-clusters* où tout nœud, qui n'est attaché à aucun cluster et possédant le poids le plus fort parmi tous ses voisins sans cluster situés à k sauts, devient *clusterhead.* Les auteurs modifient la procédure de maintenance utilisée dans *lowest id* pour prendre en considération la contrainte de k sauts.

Chatterjee *et al.* (2002) ont proposé WCA (*Weighted Clustering Algorithm*) basé sur la notion des poids combinés, calculés localement par les nœuds. Ce poids tient compte d'une variété de paramètres dont le degré, la somme de distance entre le CH et ses voisins, la vitesse des nœuds ainsi que le temps passé à l'état CH. Les auteurs ont prouvé que WCA démontre de meilleures performances en le comparant avec les autres algorithmes (tels que MOBIC, LCC). Toutefois, les CH commencent à souffrir rapidement une fois la taille des clusters commence à augmenter; ce qui engendre beaucoup de changements et dégrade le

niveau de stabilité du réseau. Notons que c'est cet algorithme qui sera considéré pour faire une étude comparative avec l'approche que nous proposons.

[Venkataraman *et al.* (2007)] ont proposé un algorithme de clustérisation permettant de construire et maintenir des *k-clusters* de façon à restreindre le nombre de nœuds à S nœuds par cluster. Les procédures utilisées sont distribuées; les tâches sont partagées entre les différents nœuds, ce qui permet d'allonger la durée de vie du réseau. L'algorithme utilise des poids assignés aux nœuds afin d'élire le *clusterhead*; les métriques de poids sont similaires à celles utilisées dans WCA. En se basant sur des simulations, les auteurs ont démontré que leur modèle permet de meilleures performances en termes de nombre de changements sur les CH et de durée de vie des nœuds, en le comparant avec *Lowest-id*, LCC et au modèle de [Fernandess et Malkhi (2002)]. Toutefois, les auteurs ne justifient pas leur choix des valeurs de k et S; la maintenance semble très compliquée et l'*overhead* généré est très élevé.

3.3 Une nouvelle approche de clustérisation dans les réseaux sans fil *ad hoc*

3.3.1 Introduction

Les réseaux filaires sont hiérarchisés et segmentés en des systèmes autonomes de sorte que les utilisateurs se trouvent en périphérie et un ensemble de nœuds spécialisés (routeurs) et structurés permette d'acheminer leurs paquets. Cette hiérarchie statique permet de déployer efficacement une série de protocoles indépendants les uns des autres et pouvant coopérer. Un protocole de routage externe comme BGP (*Border Gateway Protocol*) peut être utilisé entre les systèmes autonomes. À l'intérieur du même système, le gestionnaire réseau est libre de déployer un des protocoles de routage de type IGP (*Interior Gateway Protocol*) comme RIP et OSPF.

Toutefois, les protocoles classiques des réseaux filaires sont inutiles dans le contexte des réseaux *Ad hoc*. Lorsque l'on met des protocoles de niveau transport sur des protocoles de

routage *Ad hoc*, on obtient de très faibles performances à cause des déconnexions répétées dus aux erreurs de routage. Dans la littérature, on trouve plusieurs travaux qui se sont penchés sur l'analyse de l'efficacité des protocoles de routage et la comparaison de leurs performances dans différents environnements mobiles. Ceux-ci concluent que ces solutions ne sont pas assez robustes pour permettre le déploiement à grande échelle du réseau *Ad hoc*.

C'est pourquoi ces dernières années beaucoup d'efforts ont été concentrés sur l'amélioration des couches basses pour respecter les conditions des réseaux *Ad hoc* et sur l'imitation de la notion de hiérarchisation déployée dans les réseaux filaires. Un des importants axes de recherche visant à pallier ces dilemmes est celui de la clustérisation dans les réseaux *Ad hoc*. De nombreuses approches ont déjà été proposées (voir section 3.2.2). Cependant, elles souffrent de la plupart de carences de stabilité, d'efficacité, de robustesse aux changements topologiques, de persistance dans le temps ainsi que de trafic de contrôle et de manque de qualité de service.

Dans les sections suivantes, nous allons présenter notre vision pour un modèle de hiérarchisation d'un réseau *Ad hoc* en plusieurs clusters gérés par des CH (*clusterhead*). Ce modèle est efficace et permet non seulement de former des structures hiérarchiques, mais également d'optimiser l'utilisation des ressources, d'équilibrer la charge, d'éviter les congestions et de prendre en compte la nature des applications des utilisateurs. Un modèle analytique est également proposé servant comme outil d'optimisation des performances des clusters dans le but de fournir un certain niveau de qualité de service. Les clusters permettent de jouer le rôle des systèmes autonomes, cachant la topologie du cluster même, et facilitant ainsi le déploiement des solutions à large échelle.

Vu la nature dynamique des réseaux *ad hoc*, il est essentiel de proposer des mécanismes de maintenance des clusters produits par notre modèle. Dans le chapitre 4, nous allons décrire tous les mécanismes de gestion de nouvelles arrivées, des *handoff*, de réélection des *clusterhead* et de maintenance de nœuds déjà existants dans le réseau. Notons que la maintenance se base tant sur une combinaison de paramètres de stabilité que sur un

mécanisme de contrôle d'admission qui intègre le modèle analytique présenté dans la section 3.3.5; ce dernier permet notamment d'éviter les congestions au niveau des CH. Finalement, nous allons, dans le chapitre 5, implémenter notre modèle et nos mécanismes de maintenance dans un environnement réaliste et exposer les résultats des simulations afin de valider toutes nos propositions.

3.3.2 Motivations, perspectives et mise en contexte

Les réseaux *Ad hoc* sont caractérisés par des contraintes fortes, décrites dans la section 2.2 du chapitre 2. Étant donné que le routage dans ces réseaux dépend de l'état global du réseau, le défi est donc, de prendre en considération la mobilité, la disponibilité et la stabilité des ressources dans tout l'environnement. Les protocoles de routage classiques s'appuient souvent sur une vue à plat du réseau *Ad hoc* : considérer indistinctement l'ensemble des nœuds de sorte que ces derniers contribuent également et solidairement au routage et à la gestion du réseau. Dans ce cas, le routage utilisé se base sur un mécanisme d'inondation des paquets de contrôle afin de mettre à jour les tables de routage et la carte topologique du réseau.

Cette inondation crée des problèmes d'utilisation inefficace des ressources en termes de traitement, de capacité, d'énergie et de bande passante. [Tseng *et al.* (2002)] ont déjà traité ce problème sous le nom de tempête d'inondation. Ils ont démontré que l'inondation ne fait qu'aggraver la situation et engendrer d'autres problèmes : de redondance de messages, de contention entre les nœuds et de collisions continues entre les messages transmis.

[Santivanez *et al.* (2002)] ont également démontré qu'aucun protocole de routage ne permet le déploiement à grande échelle du réseau *Ad hoc*. La raison principale est le manque d'une structure hiérarchique : un protocole proactif doit obligatoirement annoncer sa présence dans tout le réseau, et de même, un protocole réactif doit chercher la destination dans tout le réseau si elle n'est pas directement voisine. C'est pourquoi ces dernières années beaucoup d'efforts ont été concentrés sur l'amélioration des couches basses pour respecter les

conditions des réseaux *Ad hoc*. Une sorte d'interaction entre les différentes couches (*Cross-Layer*) nous semble une solution prometteuse.

Toutefois, il faut développer des moyens pour simplifier le déploiement des solutions. Nous pensons qu'un réseau *Ad hoc* doit être hiérarchisé avant de concevoir des solutions l'exploitant, simplifiant ainsi le développement de protocoles de routage et optimisant leurs performances. La contrainte de passage à l'échelle serait facilement atteignable. Cette hiérarchie fournira une base pour les protocoles des couches supérieures, permettant de partager avec eux des informations communes et de cacher une grande partie des changements de topologie.

Comme mentionné dans l'introduction, la hiérarchisation des réseaux est déjà appliquée dans les réseaux filaires. Dans le cadre d'un réseau *Ad hoc*, hiérarchiser veut dire créer des ilots reflétant l'hétérogénéité naturelle du réseau tout entier, et bâtir des structures plus simples à exploiter au niveau de la couche réseau : des coordinateurs chefs (*clusterhead*) seront choisis parmi les nœuds les plus faiblement mobiles, les plus stables avec une autonomie en énergie élevée et de hautes capacités de traitement, de stockage et de mémoire. Nous verrons par la suite une description détaillée de notre modèle de hiérarchisation basé sur la formation de plusieurs clusters dans le réseau et satisfaisant les contraintes auparavant décrites.

Il est à noter que lorsque le nombre de nœuds augmente dans tout le réseau, les tables de routage et les informations de topologie à maintenir par nœud augmentent et deviennent assez larges, dégradant ainsi les performances globales. La hiérarchisation peut intervenir pour éviter ce genre de difficultés. Plus spécifiquement, des mises à jour agrégées peuvent dorénavant se faire seulement entre les nœuds-chefs. À l'intérieur de chaque cluster, des mises à jour détaillées sur la topologie du même cluster peuvent continuer à être échangées localement au sein du cluster. Ainsi, ce sont seulement les chefs qui devront garder une information à jour de la topologie globale du réseau, optimisant les capacités de stockage des nœuds ordinaires.

D'autre part, la clustérisation doit s'adapter à la mobilité des nœuds et aux changements de topologie. Les clusters doivent rester stables le plus longtemps possible. Des messages de contrôle additionnels sont nécessaires afin d'établir et de maintenir les clusters (voir la liste des messages dans le chapitre 4). Ce qui engendre un trafic de contrôle et un *overhead* important qu'il faut considérer dans le réseau, provoquant ainsi une baisse importante des débits, une consommation énorme d'énergie des nœuds et une dégradation significative des performances globales du réseau.

Pour éviter ce genre de dilemmes, nous avons minimisé le trafic de contrôle en optimisant le nombre d'invocations des mécanismes de maintenance (chapitre 4). Nous avons également profité des avantages de la technique CDMA; nous supposons qu'il existe un code CDMA exclusif que tous les nœuds doivent utiliser lors de la transmission de leurs messages de signalisation et de contrôle. Ceci permet d'une part, de garantir une séparation efficace entre les canaux de contrôle et les canaux de données utiles tout en réduisant les interférences inter-canaux, et d'autre part, de soulager énormément le CH surtout dans des cas d'achalandage où tous les nœuds envoient une quantité excessive de trafic de contrôle que le CH doit traiter (voir section 3.3.4.2).

Nous pensons que la taille des clusters à former est un facteur primordial à considérer dans n'importe quelle approche de clustérisation, maximisant ainsi les performances des clusters tout en évitant les congestions sur les nœuds-chefs, allongeant leur durée de vie et équilibrant la charge sur tout le réseau en obligeant les nœuds de s'attacher à un autre cluster si les conditions de performances ne le permettent pas.

[Bianchi (2000), Tay et Chua (2001), Oliveira *et al.* (2006)] ont étudié le problème de capacité dans les réseaux *Ad hoc*. Ils ont démontré que le niveau de performances dépend fortement du nombre de nœuds partageant le canal radio. En effet, le débit décroît radicalement lorsque ce nombre augmente considérablement, augmentant ainsi les délais, le taux de perte, et dégradant la qualité de service des applications utilisatrices.

Nous proposons dans cette dissertation un modèle analytique employant la méthode d'accès DCF pour optimiser le nombre de nœuds par cluster tout en respectant les besoins des applications en termes de débit, de délai et de taux de perte. Le modèle se base sur l'existence d'une relation entre le nombre de nœuds partageant le cluster, le nombre de retransmissions par paquet suite aux erreurs de transmissions qui peuvent avoir lieu sur le canal radio (voir section 3.3.5) ainsi que sur la distribution du trafic des nœuds.

Dans le but d'éviter les interférences entre les clusters et de limiter le nombre de nœuds par cluster, nous avons assigné un code CDMA spécifique pour les communications intraclusters. Les CH considérés en tant que des points d'accès virtuels, auront la responsabilité de distribuer les codes CDMA aux nœuds membres. Ils doivent garantir que chaque cluster utilise son propre code intraclusters; ce dernier doit être quasi-orthogonal avec ceux des clusters avoisinants.

Nous verrons dans la section 3.3.4.2 comment se fait la distribution des codes et quels sont les types des codes que nous utiliserons. En faisant les simulations dans le chapitre 5, nous allons remarquer que les résultats obtenus permettront de réajuster quelques paramètres à utiliser dans l'algorithme de clustérisation dans le but de fournir une meilleure qualité de service dans les clusters.

L'originalité de notre travail réside dans tous les mécanismes que nous venons d'exposer; nous considérons que notre modèle représente une base extrêmement flexible et extensible (*scalable*). Le réseau sera mieux organisé, rendant les protocoles plus efficaces et plus faciles à implémenter. Dans notre contexte, un ensemble de nœuds, qui sont au départ non-organisés, pourront collaborer au sein du réseau, dans l'objectif de créer une vue logique hiérarchisée.

3.3.3 Modélisation de la topologie *Ad hoc*

Nous pouvons modéliser un réseau *Ad hoc* sous la forme d'un graphe $G(V, E)$. V représente les sommets du graphe ou tout simplement les nœuds mobiles v_i. E représente l'ensemble de liens entre les nœuds : $E \in V^2$. Ainsi, si un nœud v_i possède un lien radio vers un nœud v_j, il existera dans le graphe de modélisation un arc orienté du sommet représentant v_i vers le sommet représentant v_j. Un arc correspond donc à un saut radio. La cardinalité du réseau qui détermine le nombre de nœuds dans le réseau est représentée par $n = |V|$.

Un voisin de v_i est définit comme étant un nœud possédant un lien radio avec v_i. Si le lien radio est symétrique, c'est-à-dire, les deux extrémités peuvent envoyer un paquet sur un lien radio de telle sorte que l'autre le reçoive, le réseau *Ad hoc* peut ainsi être modélisé à l'aide d'un graphe non dirigé comme illustré sur la figure 3.3.

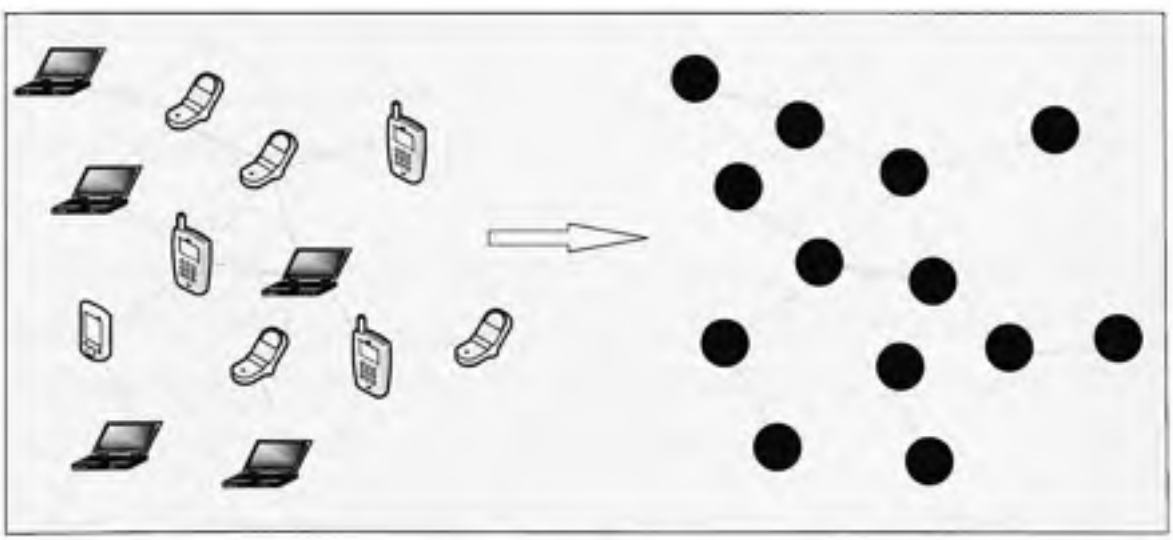

Figure 3.3 ***Modélisation d'un réseau Ad hoc par un graphe non dirigé.***

Sous des conditions où tous les nœuds v_i ont la même portée de transmission R et que le lien radio inter-nœud est idéal (sans obstacles), le réseau est souvent représenté dans la plupart des approches de la littérature par un graphe UDG (*Unit Disk Graph*) normalisé par

le rayon de recouvrement R des cercles centrés aux nœuds v_i. Dans ce cas, E peut être modélisé de la façon suivante :

$$E = \{(v_i, v_j) \in V^2 \mid d(v_i, v_j) \leq R \; et \; \cap (Cercles\ centrés\ au\ v_k)_{k=i,j} \neq \emptyset\} \quad (3.4)$$

Toutefois, nous devons garder à l'esprit que cette modélisation est simplificatrice et que la portée radio ne peut pas être modélisée par des graphes UDG. En effet, la réalité d'un environnement sans fil *Ad hoc* ne s'adapte pas aux propriétés et aux limites des graphes UDG décrites dans [Clark *et al.* (1990)]. Dans le cas où deux nœuds n'ayant pas exactement la même antenne ou la même orientation, il se peut que les liens radio soient unidirectionnels et la communication se fasse seulement dans un sens.

En réalité, la modélisation UDG est seulement utile pour donner une idée de la borne sur la cardinalité de certaines structures. Mais elle ne doit pas être considérée comme un outil reproduisant parfaitement un réseau *Ad hoc* quelconque dans un cas non idéal. Ainsi, vu que les nœuds peuvent avoir des rayons de portée différents, il est important de tenir compte de l'hétérogénéité de la portée radio des nœuds.

Dans notre approche, nous supposons que les liens sont symétriques et nous ne considérons pas la distance comme indication sur la qualité du lien. Par contre, nous nous basons sur la puissance des signaux reçus de la part des nœuds avoisinants. Ainsi, les nœuds v_i et v_j sont dans la portée de transmission l'un de l'autre si et seulement si :

$$P_{j,i}: Puissance\ reçue\ au\ v_j\ de\ la\ part\ de\ v_i \geq seuil \quad (3.5)$$

et

$$P_{i,j}: Puissance\ reçue\ au\ v_i\ de\ la\ part\ de\ v_j \geq seuil \quad (3.6)$$

Vu l'hypothèse de symétrie des liens inter-nœuds, nous pouvons ainsi modéliser l'ensemble de liens par l'équation suivante :

$$E = \{(v_i, v_j) \in V^2 \mid P_{j,i} \vee P_{i,j} \geq seuil\} \tag{3.7}$$

La clustérisation est le phénomène qui permet de partitionner ce graphe tout en respectant un certain nombre de contraintes (choix des nœuds-chefs, choix de la taille des partitions, etc.). Comme ces graphes ne respectent aucune structure régulière, partitionner un réseau *Ad hoc* d'une façon optimale devient un problème *NP-Complet* [Das et Bharghawan (1997)]. Représentons par $N_k(v_i)$, l'ensemble de nœuds voisins à k sauts de v_i et par $\Delta_k(v_i)$, la cardinalité de l'ensemble $N_k(v_i)$ est donnée par :

$$\Delta_k(v_i) = |N_k(v_i)| \tag{3.8}$$

Nous devons chercher un ensemble dominant $C \subseteq V(G)$, de cardinalité $N = |C|$ et constitué de nœuds *clusterhead* tel que :

$$\bigcup_{v_i \in C} N_k(v_i) = V \tag{3.9}$$

et

$$\forall\, v_i \in C, \exists\, v_j \in V \mid \{(v_j = v_i) \vee (v_j \in N_k(v_i))\} \tag{3.10}$$

Notre objectif étant de créer des *1-clusters*, nous pouvons ainsi écrire :

$$N_1(v_i) = \bigcup_{v_j \in C, v_j \neq v_i} \{v_j \mid P_{i,j} \geq seuil\} \tag{3.11}$$

Par simplification d'écriture dans nos *1-clusters*, nous désignons par : $N(v_i) = N_1(v_i)$ et $\Delta(v_i) = \Delta_1(v_i)$. De plus, si le réseau est homogène, son degré moyen peut être décrit par :

$$\boxed{\Delta = \frac{\sum_{v_i \in V} \Delta(v_i)}{N} = \frac{\sum_{v_i \in V} |N(v_i)|}{N}} \tag{3.12}$$

3.3.4 Description du modèle de clustérisation

3.3.4.1 Avant propos

Le routage classique utilisé dans les réseaux *Ad hoc* et basé sur les inondations, engendre des problèmes de mauvaise exploitation des ressources réseau. [Tseng *et al.* (2002)] ont étudié ce problème sous le non de tempête d'inondation (*Broadcast Storm Problem*) causant une redondance au niveau des messages transmis ainsi que de retransmissions continues causées par la contention entre les nœuds actifs.

Nous croyons que la robustesse d'un réseau *Ad hoc* est conditionnée par son hiérarchisation; il doit être hiérarchisé avant d'être utilisé. Un réseau *Ad hoc* est classiquement manipulé à plat où le traitement et la charge sont répartis de façon égalitaire dans le réseau afin de ne désavantager aucun nœud. Ainsi, la vraie problématique consiste à développer des protocoles permettant d'appliquer les fonctions des réseaux classiques (routage, découverte de services, réservation de bande passante, etc.) directement sur un grand nombre de nœuds non hiérarchisés.

Nous en déduisons un modèle basé sur la clustérisation du réseau en plusieurs clusters. L'objectif est d'étendre le routage intra et inter clusters, de faciliter la gestion, d'améliorer l'utilisation des ressources et de réduire les délais d'établissement des routes en limitant le routage aux nœuds-chefs.

Nous pensons que la clustérisation des réseaux *Ad hoc* permet d'apporter plusieurs solutions. Elle permet de tirer profit de l'hétérogénéité des nœuds, dans le sens où elle fait participer plus activement les nœuds qui sont plus stables et plus puissants. Elle permet également de réduire le nombre de transmissions des messages en optimisant les diffusions de nature *broadcast*. Ce sont seulement les nœuds-chefs qui auront la responsabilité de relayer les messages améliorant ainsi la stabilité du réseau et économisant l'énergie des nœuds ordinaires participant moins à des activités lourdes.

La clustérisation fournit une vue plus stable du réseau *Ad hoc* en cachant les changements topologiques intraclusters et réduisant ainsi l'impact de la mobilité sur la topologie entière. Ceci permet d'exploiter plus facilement le réseau ainsi que de faciliter le développement et le déploiement des protocoles à large échelle.

Plusieurs travaux de recherche dont ceux de [Vuong et Huynh (2000)] ont montré que le problème d'adaptation d'un ensemble de domination minimal *d-hop* dans une topologie *Ad hoc* simple est un problème *NP-Complet*. C'est la raison pour laquelle nous nous concentrons sur la formation et la maintenance des *1-clusters*.

3.3.4.2 Assignation des codes CDMA

Avant de discuter le mécanisme de formation des clusters, il est primordial de définir les hypothèses considérées à ce sujet afin que le modèle soit réalisable dans des conditions qui se rapprochent de la réalité. La couche MAC IEEE 802.11 constitue actuellement le meilleur candidat pour servir de couche liaison de données pour les réseaux *Ad hoc*. Nous avons utilisé l'environnement radio 802.11b en mode DCF sans RTS/CTS, avec une technique de modulation de type DSSS.

Cependant, la distribution des codes CDMA est un vrai dilemme dans des réseaux sans infrastructure comme *Ad hoc*. Nous faisons en sorte que ce sont seulement les *clusterhead* qui auront la responsabilité de gérer cette distribution. Plusieurs codes CDMA (quasi-orthogonaux entre eux) ont été adoptés dans notre approche et tiennent compte des modalités suivantes :

- utiliser un code commun et unique « *sig_code* » qui servira pour la signalisation : ce code sera utilisé lors de tout dialogue *clusterhead*-membre ou *clusterhead-clusterhead* (un nœud désirant joindre ou quitter son cluster, CH accepte un nouveau nœud, CH rejette un nœud, CH envoie une mise à jour à un CH avoisinant, etc.). Ceci s'avère très important dans le sens où le CH peut mieux performer et n'aura pas à continuellement traiter les nouvelles demandes surtout lorsqu'il y a achalandage;

- permettre à un CH élu d'assigner un code CDMA exclusif nommé « *intracluster_code* » pour sa zone. Ce qui permet non seulement de limiter le nombre de nœuds par cluster, mais également d'augmenter le débit intraclusters et diminuer les erreurs de transmission dus aux interférences. Tous les nœuds membres d'un cluster doivent utiliser ce code pour communiquer avec leur CH. Chaque CH doit garantir que ce code est exclusif pour sa zone et ne se répète pas dans les clusters adjacents. Tous les « *intracluster_code* » utilisés dans les différents clusters doivent être quasi orthogonaux entre eux; de cette façon, nous évitons les interférences avec les nœuds des clusters adjacents et nous les limiterons seulement au sein des clusters;
- utiliser un code commun et unique « *intercluster_code* » qui servira pour la transmission des données utiles entre les clusters eux-mêmes. Ce code est connu par tout nœud dans le réseau de sorte qu'il soit capable de l'utiliser une fois élu en tant que CH. Vu la condition de quasi-orthogonalité entre tous les codes adoptés, l'utilisation d'un code exclusif pour les communications entre les clusters permettra de minimiser énormément les interférences entre toutes les communications intra et inter cluster.

Nous supposons que le nombre de codes « *intracluster_code* » est assez grand. Les CH doivent faire en sorte que chaque code soit exclusif pour un cluster. Ainsi, les CH doivent maintenir une liste des « *intracluster_code* » utilisés par les clusters avoisinants. Il est à noter que le nombre de codes « *intracluster_code* » est proportionnel au nombre de bits des séquences pseudo-aléatoires utilisés pour étaler le signal transmis sur toute la bande passante (voir figure 2.5 du chapitre 2). Vu la nature asynchrone des communications dans un réseau *Ad hoc*, il devient inutile d'utiliser des codes « *intracluster_code* » orthogonaux, car il est quasi impossible de garantir une synchronisation entre les nœuds communicants.

C'est la raison pour laquelle nous avons considéré des codes quasi-orthogonaux comme ceux de *Gold* et *Kasami*. Ces codes sont des séquences d'étalement de spectre pseudo-aléatoires ayant des caractéristiques de corrélation particulières et permettent à un récepteur de pouvoir récupérer facilement les données utiles. Pour une séquence de taille 15 bits, nous pouvons avoir 17 codes *Gold* et 4 codes *Kasami*; et pour une séquence de 63 bits, nous

pouvons avoir 65 codes *Gold* et 8 codes *Kasami*. Par conséquent, ceci impose une limitation sur le nombre de clusters utilisant différents codes « *intracluster_code* ». Toutefois, un signal s'atténuant avec la distance, les codes peuvent ainsi être réutilisés à travers le réseau.

3.3.4.3 Format de l'état des nœuds

Pour la formation des clusters, nous supposons que les adresses sont statiques et non redondantes pour éviter tout conflit. Toutefois, nous assignons aux nœuds des identifiants de façon que chaque nœud $v_i \in V$ soit identifié par l'état suivant :

$$v_i: (id_{v_i}, id_{CH}, w_i, intracluster\_code, intercluster\_code, sig\_code, compteur)$$

Le pair (id_{v_i}, id_{CH}) permet d'identifier d'une façon unique l'identité d'un nœud et son appartenance à un cluster dans le réseau *Ad hoc*. Chaque nœud doit maintenir l'information concernant son CH et doit acheminer son trafic par le biais de ce dernier.

w_i représente le poids du nœud v_i. Ce poids est une combinaison de plusieurs métriques. Nous verrons par la suite comment calculer la valeur de ce poids ainsi que les métriques utilisées dans ce calcul.

Le « *compteur* » permet de compter le nombre de nœuds actuel dans le cluster et garantit que la taille du cluster ne dépasse pas une valeur optimale $|C|$. Nous verrons par la suite comment optimiser la taille des clusters afin de répondre aux besoins spécifiques des nœuds et des flux de trafic.

3.3.4.4 Choix des métriques à considérer dans la clustérisation

Comme mentionné auparavant, chaque cluster a une taille définie par le nombre de nœuds actifs qu'il peut supporter. Le choix de ce nombre permet de définir la cardinalié $N = |C|$ de l'ensemble dominant C. Nous proposons un modèle analytique détaillé dans la section 3.3.5,

permettant de définir $|C|$ de façon à garantir un niveau de qualité de service qui répond aux besoins spécifiques des clusters. Un CH doit être capable de gérer sa zone de sorte qu'il accepte ou rejette les nouvelles arrivées en se basant sur ces capacités et sans avoir à perturber les nœuds déjà admis. Le « *compteur* » permet à tout nœud membre d'avoir une idée sur la taille du cluster. Ceci l'aidera au besoin d'estimer le débit réalisable, le délai intraclusters et le taux de perte pour des prochains flux potentiels.

À la différence, les approches de clustérisation existantes ne prennent pas en considération le fait d'optimiser l'utilisation des ressources pour offrir un niveau de QoS dans le réseau. Nous voulons que chaque nœud v_i calcule un poids w_i qui détermine son niveau de performance dans le réseau. Ce poids n'est qu'une combinaison de plusieurs métriques. Nous avons choisi les métriques que nous *jugeons* les plus importantes à la détermination du niveau de stabilité et de performances d'un nœud v_i dans le réseau. Pour un nœud $v_i \in V$, ces métriques sont les suivantes :

- degré δ_i du nœud v_i: qui est défini comme étant le nombre de nœuds voisins de v_i. Ceci permet de déterminer le niveau de connectivité du nœud et sa position par rapport aux nœuds avoisinants. Ainsi :

$$\boxed{\delta_i = \Delta(v_i)} \tag{3.13}$$

- puissance de transmission P_i du nœud v_i : ceci est une meilleure indication que la métrique de somme des distances utilisée dans la majorité des approches existantes pour élire en tant que *clusterhead* le nœud couvrant le rayon de portée le plus large;

- vitesse moyenne S_i du nœud v_i calculée de la façon suivante :

$$\boxed{S_i = \frac{1}{T}\sum_{t=1}^{T} \sqrt[2]{(X_t - X_{t-1})^2 + (Y_t - Y_{t-1})^2}} \tag{3.14}$$

où (X_t, Y_t) représente les coordonnées du nœud v_i à l'instant t.

- niveau d'énergie restante E_i du nœud v_i : ceci est une meilleure indication que la métrique de temps cumulatif, pendant lequel un nœud a joué le rôle du *clusterhead*, utilisée dans la majorité des approches existantes. E_i permet à un *clusterhead* de se préparer pour résilier son rôle dès que sa valeur atteint un seuil déterminé.

3.3.4.5 Élection des nœuds *clusterhead*

Après avoir déterminé les métriques, chaque nœud v_i doit calculer son poids combiné w_i qui n'est qu'une bonne indication sur ses chances de jouer le rôle du CH. Le nœud ayant le meilleur poids dans son environnement à un saut sera élu CH. Le calcul de ce poids doit se faire localement et périodiquement de la façon suivante :

$$\boxed{w_i = a \times \delta_i + b \times E_i + c \times P_i + d \times S_i \quad où \quad a + b + c + d = 1} \qquad (3.15)$$

Les paramètres a, b, c et d sont des facteurs qui dépendent des besoins du système. La flexibilité de changer ces paramètres nous aidera à appliquer notre approche dans multiples scénarios. Vu que les réseaux *Ad hoc* sont très souvent formés à partir d'un groupe de nœuds pour un but spécifique (salle de conférence, compagne militaire, etc.), nous supposons que dans chaque cluster, tous les nœuds sont identiques et ont les mêmes besoins. Par exemple, dans une salle de conférence où l'environnement est peu mobile, les nœuds sont presque fixes, nous pouvons ainsi privilégier le facteur b pour choisir le nœud ayant la batterie la plus puissante comme CH. Dans un environnement militaire, la mobilité des nœuds est primordiale, nous pouvons privilégier le facteur d pour réfleter le choix du CH.

D'autre part, nous pensons que la vitesse S_i d'un nœud v_i ne reflète pas la mobilité relative de ce nœud. Il se peut que tous les nœuds du cluster se déplacent à haute vitesse et dans la même direction, dans ce cas le cluster devrait être considéré stable. Cependant, nous continuons à utiliser cette métrique pour faire l'élection initiale des CH, car en démarrant le réseau nous ne possédons que cette métrique (pas de point de référence).

En revanche, nous utiliserons la mobilité relative des nœuds durant la procédure de maintenance. C'est l'historique de puissance des signaux échangés entre deux nœuds avoisinants qui définit la mobilité relative d'un nœud par rapport à un autre. Ce mécanisme sera détaillé dans la section 4.3.7 du chapitre 4.

3.3.5 Modèle analytique pour le contrôle d'admission

Sous des conditions d'utilisation surchargée du réseau, l'équilibrage de charge et le contrôle d'admission deviennent une nécessité afin de fournir la qualité de service aux nœuds existants dans le réseau et aux nouvelles arrivées désirant un service de qualité. Dans cette section, nous introduisons un modèle analytique permettant d'analyser les paramètres de QoS tels que le débit, le délai et le taux de perte. L'objectif est de faire un compromis de sorte que les CH puissent admettre les nœuds tout en respectant un niveau désiré de QoS.

3.3.5.1 Modélisation des paramètres d'accès au canal radio

Nous supposons que la méthode d'accès DCF est utilisée pour gérer l'accès au canal radio entre les différents participants. Nous avons pris comme hypothèse que le CH aura la responsabilité de générer l'*intracluster_code* à utiliser pour moduler les signaux au sein du cluster. Du fait que tous les nœuds du même cluster utilisent ce même code pour leurs communications intraclusters, des interférences et des erreurs de transmission auront lieu sur le canal radio si les stations décident de transmettre au même moment [Jiang *et al.* (2007)].

Le protocole DCF de la couche MAC ne recevant pas d'acquittement pour un paquet à cause de ces erreurs et ces interférences; il considère que le paquet a été perdu à cause d'une collision et doit être retransmis après un *backoff* (nous supposons qu'il n'y a pas de mécanismes de correction d'erreurs sur les bits).

Il est à noter qu'en utilisant le même code *intracluster_code*, les collisions ne sont pas pertinentes qu'au début de transmission et non pas au cours d'une transmission. Ceci est d'autant plus important pour permettre la transmission simultanée de plusieurs stations, réduisant énormément la probabilité de collision sur le canal.

Les clusters avoisinants utilisant d'autres codes quasi-orthogonaux *intracluster_code*; ils n'auront aucun impact sur les communications locales les uns des autres. De même, les messages de contrôle et les communications interclusters se faisant sur d'autres codes CDMA; ceci permet de réduire les interférences sur les communications intraclusters.

Dans ce contexte, nous analysons l'impact des nœuds actifs et des retransmissions locales sur les performances de nos structures. Les résultats permettront aux CH de prendre des décisions sur l'admission et/ou le rejet de nouvelles arrivées en vue de maintenir un service local de qualité acceptable et déterminé par les types d'applications en cours.

Nous nous sommes intéressés à l'étude des performances du système dans le pire des cas où le CH sert tous les nœuds qui sont dans sa portée de transmission de sorte que nous puissions dériver le nombre maximal de nœuds membres qu'un CH peut servir (c'est-à-dire la cardinalité N des clusters). Cette analyse nous aidera dans le mécanisme de formation des clusters et garantit que les capacités de ces derniers ne baissent pas en termes de débit, de délai et de taux de perte.

Reprenons la méthode DCF utilisée dans la norme IEEE 802.11 que nous avons décrite dans le chapitre 2. Nous avons commencé par modéliser le schéma *backoff* employé dans DCF. Nous savons que pour chaque paquet à transmettre, un nœud initialise une période *backoff* qui est un nombre entier aléatoire uniformément distribué sur l'intervalle $(0, W-1)$. W est connu sous le nom de la fenêtre de contention (*Contention Window*), il dépend du nombre de transmissions erronées qui ont eu lieu sur un paquet quelconque. Cette fenêtre est déterminée comme étant un certain nombre d'emplacements (slots).

À la première tentative de transmission, la valeur de W est égale à CW_{min} (fenêtre de contention minimale). Désignons par p la probabilité que le paquet transmis fasse face à une retransmission dans le canal (dû à plusieurs transmissions simultanées ayant le même *intracluster_code* sur le même canal). Dans ce cas-ci, la valeur de W est doublée suite à chaque transmission non réussie, jusqu'à une valeur maximale de $CW_{max} = 2^m CW_{min}$ où m représente le nombre de tentatives de transmissions non réussies pour ce paquet. m est souvent connu par le nom de *backoff stage* et est défini dans la norme IEEE 802.11.

Après avoir atteint la valeur CW_{max}, W garde cette valeur jusqu'à ce qu'il soit réinitialisé à sa valeur initiale CW_{min}. Ainsi, nous pouvons dériver la probabilité qu'un nœud choisisse la fenêtre W de la façon suivante :

$$P\{Window = W\} = \begin{cases} p^{m-1}(1-p) \quad pour \quad W = 2^{m-1}CW_{min} \\ \\ p^m \quad pour \quad W = CW_{max} \end{cases} \tag{3.16}$$

Tay et Chua (2001) ont dérivé la probabilité de retransmission p dans le cas d'un réseau saturé où chaque nœud a toujours des paquets prêts à être envoyés dans ses files d'attentes. Nous savons que lorsque le canal est occupé, chaque paquet sera immédiatement précédé par un *backoff*.

Le paramètre m représente le compteur du nombre de retransmissions allouées pour un paquet quelconque. Le nœud ayant atteint m retransmissions sans avoir réussi à transmettre son paquet avec succès devrait considérer le canal radio comme inexploitable. Ce phénomène de transmission du paquet d'un nœud peut être représenté par une distribution

géométrique[11]. Ainsi, la probabilité que le nœud arrive à transmettre avec succès son paquet après 2 tentatives est $p(1-p)$, dans ce cas la taille moyenne du *backoff* en termes de nombre de slots à attendre est de l'ordre de $\frac{2W}{2}$. En généralisant ce concept dans le cas d'un réseau où tous les nœuds ont des paquets en attente de transmission dans leur file d'attente, le *backoff* moyen peut être modélisé de la manière représentée dans l'équation (3.17).

$$(1-p)\frac{W}{2} + p(1-p)\frac{2W}{2} + \cdots + p^m(1-p)\frac{2^m W}{2} + p^{m+1}\frac{2^m W}{2}$$

$$= \boxed{\frac{1-p-p(2p)^m}{1-2p}\frac{W}{2}} \qquad (3.17)$$

3.3.5.2 Modélisation des paramètres de performances intraclusters

Nous pensons que les nœuds d'un cluster n'ont pas toujours de paquets à transmettre. Nous proposons un modèle qui estime la probabilité de transmission non réussie sur le canal radio (suite à des interférences, des bits erronés, etc.). Nous supposons que le taux d'arrivées des paquets dans la file d'un nœud suit une distribution de Poisson sans mémoire de paramètre λ, ces paquets seront emmagasinés dans un tampon de taille infinie au niveau du nœud. Nous savons que le taux de service μ dépend des conditions de la couche physique, ce taux sera modélisé et décrit dans la section 3.3.5.2.2.

Par conséquent, chaque nœud du cluster peut être modélisé par un système *M/M/1*. Le serveur représente le canal radio qui est partagé entre les différents nœuds et dont la

---

[11] On considère une épreuve de Bernoulli dont la probabilité de succès est p et celle d'échec est $q = 1 - p$. On renouvelle cette épreuve de manière indépendante jusqu'au premier succès. La probabilité de faire k essais avant d'obtenir le premier succès est $p^{k-1}q$.

méthode DCF y gère l'accès. Pour un nœud *M/M/1* dans un état stationnaire ($\lambda/\mu < 1$), la probabilité que ce dernier n'ait aucun paquet à transmettre est donnée par :

$$\pi_0(noeud) = 1 - \frac{\lambda}{\mu} \tag{3.18}$$

La probabilité que tous les membres N du cluster n'aient pas de paquets à transmettre détermine la probabilité que le cluster soit dans un état de repos. Cette probabilité est déterminée par :

$$\pi_0(cluster) = \left(1 - \frac{\lambda}{\mu}\right)^N \tag{3.19}$$

Au niveau du nœud, si le cluster est dans un état de repos, la transmission d'un paquet ne sera jamais différée. Dans ce cas, nous pouvons conclure que :

$$P\{W \neq 0\} = 1 - \left(1 - \frac{\lambda}{\mu}\right)^N \tag{3.20}$$

Par conséquent, en adaptant l'approche de Tay et Chua (2001) sur l'ensemble des nœuds du cluster, la taille moyenne du *backoff* d'un nœud en termes de nombre de slots à attendre sera donnée par :

$$\boxed{\overline{W} = \left[1 - \left(1 - \frac{\lambda}{\mu}\right)^N\right]\left[\frac{1-p-p(2p)^m}{1-2p}\frac{W}{2}\right]} \tag{3.21}$$

3.3.5.2.1 Analyse du débit intraclusters atteignable

Nous désignons par T le débit maximal atteignable dans un cluster donné. Ce débit est défini comme étant le temps nécessaire pour transmettre un paquet de taille L durant un cycle d'échange (comme défini dans la norme 802.11).

Il est à noter qu'un cycle d'échange pour un paquet donné est composé d'un seul cycle d'échange réussi, de plusieurs cycles d'échange erronés (avec des retransmissions) ainsi que de plusieurs cycles de repos sur le canal. Il faut également tenir compte des temps de transmission de différents entêtes. Ainsi, nous pouvons définir T de la manière suivante :

$$T = \frac{P(transmission\ réussie\ du\ paquet\ sur\ le\ canal) \times Taille\ (paquet)}{durée\ cycle_{succès} + durée\ cycle_{échec} + durée\ cycle_{repos}} \quad (3.22)$$

$$\text{où} \begin{cases} durée\ cycle_{succès} = \alpha \times P(transmission\ réussie\ sur\ le\ canal) \\ durée\ cycle_{échec} = \beta \times P(transmission\ non\ réussie\ sur\ le\ canal) \\ durée\ cycle_{repos} = \gamma \times P(canal\ soit\ en\ repos) \end{cases} \quad (3.23)$$

α représente la durée d'une transmission réussie sur le canal, c'est-à-dire la durée pendant laquelle le canal a été saisi par une transmission réussie. β représente la durée d'une transmission non réussie sur le canal. Finalement, γ représente la durée d'un slot de temps défini dans la norme 802.11.

α et β dépendent du mécanisme de contrôle d'accès au canal. Il faut prendre en considération l'utilisation ou la non utilisation du mécanisme RTS/CTS. Il est à noter que nous avons supposé que les paquets sont transmis sans être fragmentés; en outre, les problèmes de stations cachées/exposées ne sont pas pratiquement applicables dans notre réseau, car en utilisant plusieurs codes CDMA les stations peuvent transmettre et recevoir simultanément.

Ainsi, pour le mécanisme de base DCF sans RTS/CTS, les valeurs de α et β sont calculées de la façon suivante :

$$\alpha = DIFS + \frac{PHY_{header} + MAC_{header} + Paquet}{\vartheta} + SIFS + \frac{ACK}{\vartheta} + 2\varepsilon \quad (3.24)$$

$$\beta = DIFS + \frac{PHY_{header} + MAC_{header} + Paquet}{\vartheta} + \varepsilon \quad (3.25)$$

ϑ représente la bande passante du canal et ε réprésente le délai moyen de propagation sur le canal. Ces paramètres sont bien définis dans le standard IEEE 802.11.

Si pour quelques raisons la méthode DCF est utilisée en présence de RTS/CTS, dans ce cas les valeurs de α et β doivent être calculées de la manière suivante :

$$\alpha = DIFS + \frac{RTS + CTS}{\vartheta} + DIFS + \frac{PHY_{header} + MAC_{header} + Paquet}{\vartheta} + 3SIFS + \frac{ACK}{\vartheta} + 4\varepsilon \quad (3.26)$$

$$\beta = DIFS + \frac{RTS}{\vartheta} + \varepsilon \quad (3.27)$$

Il s'agit maintenant de calculer ces différentes probabilités pour déduire le temps nécessaire à la transmission d'un paquet quelconque. Nous supposons qu'une retransmission pourrait avoir lieu sur un nœud si les nœuds restants choisissaient le même slot de temps de leur $\overline{W}$. Par conséquent, la probabilité p qu'une retransmission ait lieu sur le paquet d'un nœud du cluster est égale à la probabilité qu'au moins un des nœuds restants ayant du trafic dans sa file choisisse le même slot dans $\overline{W}$ (sur le même code *intracluster_code*). Ainsi, nous pouvons représenter cette probabilité par :

$$p = 1 - \left[\left(1 - \pi_0(noeud)\right)\left(1 - \frac{1}{\overline{W}}\right)\right]^{N-1} \quad (3.28)$$

En résolvant l'équation (3.28), nous pouvons déduire la relation entre p, N et W de la façon suivante :

$$\boxed{p = 1 - \left(\frac{\lambda}{\mu}\right)^{N-1}\left[1 - \frac{2(1-2p)}{W\left[1-\left(1-\frac{\lambda}{\mu}\right)^{N}\right][1-p-p(2p)^{m}]}\right]^{N-1}} \quad (3.29)$$

D'autre part, désignons par q, la probabilité qu'un nœud, ayant du trafic dans sa file, transmette dans un slot aléatoire dans $\overline{W}$. Ainsi, la probabilité qu'une retransmission d'un paquet ait lieu sur le canal dans un slot déterminé est égale à la probabilité qu'au moins un des $(N-1)$ nœuds restants, parmi ceux ayant du trafic à envoyer, transmette dans le même slot. D'une façon plus formelle, ceci s'écrit de la façon suivante :

$$p = 1 - (1-q)^{N-1} \quad (3.30)$$

Par conséquent, nous pouvons maintenant formuler les équations suivantes :

$$P(transmission\ réussie) = Nq(1-q)^{N-1} \quad (3.31)$$

$$P(transmission\ non\ réussie) = 1 - (1-q)^{N} - Nq(1-q)^{N-1} \quad (3.32)$$

$$P(canal\ en\ repos) = (1-q)^{N} \quad (3.33)$$

Par résolution de l'équation (3.22), nous pouvons établir la relation entre N et T :

$$\boxed{T = \frac{Nq(1-q)^{N-1} \times L(DATA)}{\alpha Nq(1-q)^{N-1} + \beta[1-(1-q)^{N} - Nq(1-q)^{N-1}] + \gamma(1-q)^{N}}} \quad (3.34)$$

3.3.5.2.2 Analyse du délai intraclusters

Le délai intraclusters $\overline{D}$ est défini comme étant la durée d'attente moyenne pour qu'un paquet soit transmis avec succès sur le canal. Nous ne prenons pas en considération les

durées de séjour des paquets dans les files d'attente des nœuds. $\overline{D}$ inclut la durée d'attente pour faire écouler tous les slots aléatoires du *backoff* $\overline{W}$, la durée α durant lequel le canal a été saisi par une transmission réussie, ainsi que la durée t nécessaire pour transmettre le paquet sur canal.

Il est à noter que t dépend du taux de service μ du canal radio. De cette façon, nous pouvons représenter $\overline{D}$ de la façon suivante :

$$\overline{D} = \left(\gamma\overline{W} + \alpha\right) + \frac{1}{\mu} \tag{3.35}$$

Pour calculer le taux de service μ, ce taux est exprimé par la durée d'attente moyenne pour transmettre avec succès un paquet. Ceci dépend de l'état du canal (les transmissions erronées, les slots de repos, etc.) et du nombre moyen de retransmissions $\overline{N}_{cp}$ que le paquet subira avant qu'il soit transmis correctement. Par conséquent, μ peut être exprimé de la façon suivante :

$$\mu = \frac{1}{(\gamma\overline{W}+\beta+\Delta)\overline{N}_{cp}} \tag{3.36}$$

Δ représente le temps qu'un nœud doit ré-attendre avant de recommencer à écouter le canal. Ceci respecte bien la méthode d'accès DCF définie dans la norme IEEE 802.11.

$$\Delta = SIFS + ACK\_Timeout \tag{3.37}$$

Pour calculer $\overline{N}_{cp}$, nous prenons comme hypothèse que chaque transmission d'un nœud est indépendante des transmissions des autres nœuds. Nous pouvons ainsi modéliser le nombre moyen de transmissions réussies $\overline{N}_{sp}$ pour un paquet. Ceci suit une distribution géométrique ayant une espérance :

$$Esp\left(\overline{N}_{sp}\right) = \frac{1}{1-p} \tag{3.38}$$

Cette approximation nous donne une idée sur le nombre moyen de tentatives qu'un nœud doit entreprendre afin de transmettre correctement son paquet. Ce qui veut dire que ce nœud a subi un nombre moyen de $(\overline{N}_{sp} - 1)$ transmissions erronées avant de pouvoir transmettre son paquet avec succès. Par conséquent, nous pouvons exprimer $\overline{N}_{cp}$ de la façon suivante :

$$\overline{N}_{cp} = \frac{1}{1-p} - 1 \tag{3.39}$$

Finalement, l'expression du délai intraclusters est donnée par l'équation suivante :

$$\boxed{\overline{D} = \left(\gamma \overline{W} + \alpha\right) + \frac{p(\gamma \overline{W} + \beta + \Delta)}{1-p}} \tag{3.40}$$

3.3.5.3 Modélisation des paramètres de performances interclusters

Quant à lui, le CH doit relayer tout le trafic intraclusters, il est ainsi considéré comme un point d'agrégation. D'autre part, les nœuds *Ad hoc* considérés (non MIMO) disposent d'une seule antenne servant soit à l'émission, soit à la réception (canal *half duplex*). Par conséquent, le CH ne peut pas communiquer simultanément avec ses membres et avec les autres CH avoisinants. Un CH étant capable d'écouter les autres canaux modulant des signaux avec le code *intercluster_code,* son débit sera alors influencé non seulement par les transmissions de ses membres, mais également par celles des CH avoisinants [Jiang *et al.* (2007)].

Pour toutes ces raisons, il faut ainsi tenir compte du nombre moyen de clusters dans le réseau; si le réseau est homogène, le nombre moyen de clusters produits devrait être : $\frac{|V|}{|C|} = \frac{n}{N}$ (voir équation 3.12).

D'autre part, le CH n'a pas à attendre pour transmettre immédiatement un paquet sur le canal *intercluster_code* lorsqu'il est en mode émission et lorsque tous les CH avoisinants ne

transmettent pas. Ceci veut dire que le paquet sera directement transmis si le CH n'est pas en mode réception du trafic de sa zone sur l'*intracluster_code* et si les CH avoisinants n'interfèrent pas sur ses transmissions sur l'*intercluster_code*.

3.3.5.3.1 Analyse du débit interclusters atteignable

Les CH étant des points d'agrégation de trafic, ils ont très souvent des paquets à transmettre. Nous pouvons ainsi les considérer comme étant saturé la plupart de temps. Par conséquent, en suivant le même raisonnement de la section 3.3.5.1, la taille moyenne du *backoff* pour les transmissions interclusters au niveau du CH, en termes de nombre de slots à attendre, sera donnée par l'équation suivante :

$$\boxed{\overline{\acute{W}} = \frac{1-\acute{p}-p(2\acute{p})^m}{1-2\acute{p}}\frac{W}{2}} \tag{3.41}$$

Désignons par $\acute{p}$ la probabilité de retransmission au niveau du CH dans le cas de saturation. En suivant les mêmes arguments des équations (3.28) et (3.29), nous pouvons écrire :

$$\acute{p} = 1 - \left(1 - \frac{1}{\overline{\acute{W}}}\right)^{\frac{n}{N}-1} \tag{3.42}$$

ou bien

$$\boxed{\acute{p} = 1 - \left[1 - \frac{2(1-2\acute{p})}{W[1-\acute{p}-\acute{p}(2\acute{p})^m]}\right]^{\frac{n}{N}-1}} \tag{3.43}$$

De même, si nous désignons par $\acute{q}$, la probabilité qu'un CH, ayant du trafic dans sa file, transmette dans un slot aléatoire dans $\overline{\acute{W}}$. Nous pouvons écrire la relation entre $\acute{p}$ et $\acute{q}$ de la façon suivante :

$$\acute{p} = 1 - (1 - \acute{q})^{\frac{n}{N}-1} \tag{3.44}$$

En suivant les mêmes définitions données dans les équations (3.22) et (3.23) tout en tenant compte du nombre moyen de clusters dans le réseau, le débit interclusters $\acute{T}$ peut être exprimé de la manière suivante :

$$\boxed{\acute{T} = \frac{\acute{q}\frac{n}{N}(1-\acute{q})^{\frac{n}{N}-1} \times L(DATA)}{\alpha\acute{q}\frac{n}{N}(1-\acute{q})^{\frac{n}{N}-1} + \beta\left[1-(1-\acute{q})^{\frac{n}{N}} - \frac{n}{N}\acute{q}(1-\acute{q})^{\frac{n}{N}-1}\right] + \gamma(1-\acute{q})^{\frac{n}{N}}}} \quad (3.45)$$

3.3.5.3.2 Analyse du délai interclusters et du délai de bout en bout

Ici, nous pouvons suivre la même définition donnée dans la section 3.3.5.2.2, le délai $\overline{D}$ interclusters est défini comme étant la durée d'attente moyenne pour qu'un paquet soit transmis avec succès sur le canal interclusters. En suivant le même raisonnement, nous pouvons formuler ce délai de la façon suivante :

$$\boxed{\overline{D} = \left(\gamma\overline{W} + \alpha\right) + \frac{\acute{p}\left(\gamma\overline{W} + \beta + \Delta\right)}{1-\acute{p}}} \quad (3.46)$$

Le délai moyen de bout en bout d dépend du nombre moyen de sauts séparant la source de la destination. Si le réseau est homogène, ce délai peut être approximer par :

$$\boxed{d = 2\overline{D} + \left(\frac{n}{N} - 1\right)\overline{D}} \quad (3.47)$$

3.4 Conclusion

À travers ce chapitre, nous avons pu survoler les différentes approches de contrôle de topologie proposées dans la littérature basées sur la création des structures essayant de calquer les fonctionnalités des réseaux filaires.

Les dorsales ont été initialement conçues dans le but de réduire le trafic de contrôle. Seuls les nœuds appartenant à cette dorsale sont autorisés à relayer un trafic de diffusion. Ainsi, l'optimisation de la cardinalité d'une dorsale est un des critères fondamentaux de performance. En revanche, la minimisation de la cardinalité ne doit pas pour autant pénaliser la robustesse et le rendement de ces structures. Un compromis entre cardinalité et robustesse doit être pris en considération lors de l'exécution des algorithmes de construction de la dorsale de sorte que, si la dorsale perd temporairement sa connectivité, les paquets en transit devraient être délivrés avec le minimum de pertes possibles.

Les auteurs proposent souvent une reconstruction périodique de la dorsale sans étudier le moment opportun d'une telle reconstruction. De plus, le trafic de contrôle engendré et le temps de convergence qui peut rendre le réseau inexploitable durant une certaine période, nous semblent être des inconvénients à considérer lors de l'implémentation de telles approches.

Les clusters par ailleurs présentent une autre vision permettant de contrôler la topologie du réseau. Ils consistent à découper la topologie en plusieurs zones orchestrées par des chefs attitrés. Ces chefs sont autorisés à relayer les paquets de leurs membres et de garantir la gestion de la zone. Les clusters devraient prendre en compte l'hétérogénéité du réseau et rester le plus stable possible tout en offrant un niveau de qualité de service acceptable pour leurs clients. Des clusters changeant constamment impacteront sans aucun doute sur les protocoles qui les exploitent, générant notamment plus de trafic de contrôle et engendrant des répétitions périodiques des mécanismes de clustérisation. Des tels critères n'ont, selon l'état de nos connaissances, jamais été pris en considération. Les algorithmes de clustérisation décrits auparavant n'ont notamment pas réussi à garantir la robustesse, la persistance, le passage à l'échelle et la distributivité.

Dans ce chapitre, nous avons tout d'abord présenté notre vision de clustérisation d'un réseau mobile *Ad hoc*. Nous avons décrit les métriques utilisées pour faire une construction plus performante des clusters, tenant compte de toutes ces métriques combinées dans un poids

Les dorsales ont été initialement conçues dans le but de réduire le trafic de contrôle. Seuls les nœuds appartenant à cette dorsale sont autorisés à relayer un trafic de diffusion. Ainsi, l'optimisation de la cardinalité d'une dorsale est un des critères fondamentaux de performance. En revanche, la minimisation de la cardinalité ne doit pas pour autant pénaliser la robustesse et le rendement de ces structures. Un compromis entre cardinalité et robustesse doit être pris en considération lors de l'exécution des algorithmes de construction de la dorsale de sorte que, si la dorsale perd temporairement sa connectivité, les paquets en transit devraient être délivrés avec le minimum de pertes possibles.

Les auteurs proposent souvent une reconstruction périodique de la dorsale sans étudier le moment opportun d'une telle reconstruction. De plus, le trafic de contrôle engendré et le temps de convergence qui peut rendre le réseau inexploitable durant une certaine période, nous semblent être des inconvénients à considérer lors de l'implémentation de telles approches.

Les clusters par ailleurs présentent une autre vision permettant de contrôler la topologie du réseau. Ils consistent à découper la topologie en plusieurs zones orchestrées par des chefs attitrés. Ces chefs sont autorisés à relayer les paquets de leurs membres et de garantir la gestion de la zone. Les clusters devraient prendre en compte l'hétérogénéité du réseau et rester le plus stable possible tout en offrant un niveau de qualité de service acceptable pour leurs clients. Des clusters changeant constamment impacteront sans aucun doute sur les protocoles qui les exploitent, générant notamment plus de trafic de contrôle et engendrant des répétitions périodiques des mécanismes de clustérisation. Des tels critères n'ont, selon l'état de nos connaissances, jamais été pris en considération. Les algorithmes de clustérisation décrits auparavant n'ont notamment pas réussi à garantir la robustesse, la persistance, le passage à l'échelle et la distributivité.

Dans ce chapitre, nous avons tout d'abord présenté notre vision de clustérisation d'un réseau mobile *Ad hoc*. Nous avons décrit les métriques utilisées pour faire une construction plus performante des clusters, tenant compte de toutes ces métriques combinées dans un poids

que chaque nœud détient. Nous avons utilisé plusieurs types de codes CDMA pour limiter les interférences entre clusters et maximiser l'utilisation des ressources radio.

Nous avons également proposé un modèle analytique qui permet d'optimiser le nombre de membres selon le niveau de qualité de service voulu. L'algorithme de construction des clusters s'adapte facilement pour tenir compte des paramètres de QoS. Nous allons voir dans le chapitre 5 les résultats de l'analyse numérique qui montrent l'effet de la taille des clusters sur le débit, le délai et le taux de perte de paquets. En effet, les paramètres de la couche MAC ainsi que la taille des clusters impactent directement la qualité de service intraclusters. Un bon compromis entre taille du cluster et niveau de service fourni doit être finement établi afin de maximiser les performances de ces structures.

La maintenance des clusters est rendue nécessaire dans un environnement dynamique comme *Ad hoc* où les nœuds sont libres de se déplacer en quittant/joignant les clusters et les conditions du réseau changent très fréquemment. Dans le chapitre suivant, nous allons évoquer la procédure de maintenance qui sera employée dans notre modèle dans le but de garantir une meilleure connectivité et une stabilité continue.

CHAPITRE 4

LA MAINTENANCE DES CLUSTERS

4.1 Introduction

Nous avons vu dans le chapitre précédent comment le poids assigné à chaque nœud tient compte d'un ensemble de métriques (batterie, puissance de transmission, vitesse, etc.) pour construire les clusters à partir d'un groupe de nœud dispersé aléatoirement dans le réseau. Toutefois, les réseaux *Ad hoc* sont caractérisés par des contraintes très dynamiques dans le temps. Un nœud peut à tout moment disparaître suite à un changement de position ou à un manque de batterie, rendant ainsi le réseau très changeant.

Par conséquent, nos clusters exigent une procédure de maintenance flexible et peu coûteuse en termes de ressources radio et de traitement. Cette procédure doit leur garantir une connectivité maximale et une durée de vie plus longue. Nous croyons que la mobilité est la première cause qui provoque des changements au niveau des nœuds, exigeant ainsi la gestion des abonnements/désabonnements à un cluster donné. C'est la raison pour laquelle nous avons intégré la métrique de mobilité comme une des clés primordiales pour la maintenance des *1-clusters*.

Plus spécifiquement, nous nous intéressons au facteur de la mobilité relative d'un nœud par rapport à d'autres, que nous pouvons obtenir par la surveillance de l'historique des puissances des signaux reçus de la part des nœuds avoisinants.

Dans ce chapitre, nous allons présenter les différents mécanismes utilisés pour maintenir nos structures. Les clusters doivent constamment être mis à jour pour que les *clusterhead* restent au sein de leur cellule le plus longtemps possible tout en gérant les adhésions, les départs, les *handoff*, etc. Les clusters peuvent naturellement servir au routage. Des messages de contrôle échangés entre les différents nœuds-chefs permettent de garder une vue globale

de la topologie. La maintenance des routes se fait par la maintenance des informations sur les chefs. Un chef, calculant toutes les routes et stockant les informations sur le réseau, contribue sans doute à réduire la charge des autres nœuds ordinaires. Seuls les chefs s'échangeront des paquets de topologie des clusters, minimisant davantage l'*overhead* induit et économisant énormément d'énergie.

4.2 Préliminaires

4.2.1 Messages de contrôle employés dans la procédure de maintenance

Le tableau 4.1 illustre tous les messages de contrôle utilisés dans la procédure de maintenance des clusters. Il est à noter que tous ces messages sont envoyés sur le canal en utilisant le code CDMA spécifique à la signalisation « *sig_code* » dans le but de minimiser les interférences et l'impact du trafic de contrôle sur celui des données utiles.

Tableau 4.1

Format des messages employés dans la procédure de maintenance

Types et entêtes des messages	Description des messages
hello (id_{v_i}, id_{CH}, w_i, compteur)	Mettre à jour les tables des nœuds
Join (id_{v_i}, id_{CH})	Tenter de joindre un cluster
accept (id_{v_i}, id_{CH}, w_i)	CH accepte un *Join*
reject (id_{v_i}, id_{CH}, w_i)	CH rejette un *Join*
CH_request (id_{v_i})	Nœud voulant se déclarer CH
CH_response(id_{CH}, intracluster_code)	CH accepte un *CH_request*
Join_accept (id_{v_i}, id_{CH}, w_i, compteur)	Nœud accepte le message *accept*
CH_ACK (id_{v_i}, id_{CH}, w_i , intracluster_code, compteur)	CH déclare le nœud comme membre
Database_info(id_{v_i}, id_{CH}, w_i, compteur)	CH actuel envoie la *database* au nouveau CH élu
Database_ACK (id_{v_i}, id_{CH}, w_i, compteur)	Nouveau CH élu accepte la *database* reçue
CH_change (id_{v_i})	CH annonce un changement de CH
CH_info (id_{v_i}, id_{CH}, w_i, compteur)	CH accepte la présence d'un nouveau CH dans le réseau
leave (id_{v_i}, id_{CH})	Nœud quitte son cluster

4.2.2 Maintien des informations

En plus de garder un état ou un ensemble de paramètres identifiant le nœud comme décrit dans la section 3.3.4.3 du chapitre 3, un nœud v_i non CH doit également maintenir une table qui comprend la liste de tous les nœuds avoisinants dans le cluster auquel il fait partie, ainsi que leur poids correspondant. Cette table nommée « *intracluster_table* » comprend l'identifiant des nœuds du cluster ainsi que le poids de chacun d'eux. Ceci permettra au nœud de vérifier à tout moment son éligibilité pour devenir CH. Un nœud non CH doit également mettre à jour son état suite à la réception d'un message de type *CH_change* annonçant aux membres l'identité du nouveau CH élu dans le cluster.

En outre, un nœud CH doit maintenir une deuxième table nommée « *intercluster_table* » comprenant la liste des CH avoisinants ainsi que leur poids. Ceci facilitera sans doute le routage interclusters en permettant au CH de choisir le meilleur CH avoisinant en tant que prochain saut en se basant sur les poids contenus dans la table « *intercluster_table* ».

Un nœud CH doit également traiter les messages de type *leave* envoyés de la part des nœuds membres désirant quitter le cluster. Dans ce cas, le CH met à jour ses tables et son état, et envoie un *hello* pour annoncer la valeur du nouveau « compteur » indiquant la nouvelle taille du cluster.

Dans un réseau complexe, les nœuds doivent collaborer entre eux pour mettre à jour les tables. Des messages de type *hello* ayant un TTL [12] = 1 sont ainsi employés pour rafraîchir ces tables par des valeurs fraîches de poids, compteur, etc. Ces messages sont périodiquement échangés (à toutes les 5 secondes) soit entre les CH pour mettre à jour les tables « *intercluster_table* », soit entre un CH et ses membres pour mettre à jour toutes les tables « *intracluster_table* ».

---

[12] Time To Live : représente le nombre de sauts qu'un paquet peut parcourir avant qu'il expire

4.2.3 Temporisateurs

Un temporisateur est associé à chaque entrée des tables des nœuds de sorte que ce dernier expire si le nœud ne reçoit pas un message de rafraîchissement *hello* durant un certain laps de temps pour une entrée quelconque. Si le nœud est un nœud non CH, il efface l'entrée de sa table « *intracluster_table* », mais ne modifie pas la valeur du *compteur* en attendant que le CH l'efface pareillement, décrémente le *compteur* et mette à jour les nœuds par le biais d'un message *hello*. De même, un nœud CH, ne recevant pas de message *hello* d'un CH avoisinant pendant un laps de temps défini, doit enlever l'entrée correspondante de sa table « *intercluster_table* ».

4.3 Maintenance de la topologie du réseau *ad hoc*

4.3.1 Avant propos

Dans la majorité des algorithmes de clustérisation existants, les auteurs ne tiennent pas compte du nombre d'invocations de la ré-clustérisation ainsi que des phénomènes de gestion des *handoff*. Ce qui rend le réseau instable et les structures très changeantes lorsque les nœuds sont en mouvement. Les algorithmes de maintenance centralisés exigent des CH un fardeau plus pesant, épuisant rapidement leur batterie. Toutefois, les décisions prises par un CH sont optimales. D'autre part, les algorithmes de maintenance distribués semblent beaucoup mieux adaptés aux réseaux *Ad hoc* en cas de mobilité où la maintenance sera faite par une collaboration entre tous les nœuds. Le fardeau se voit ainsi distribué sur plusieurs nœuds. Toutefois, il devient très difficile de garantir une forte connectivité entre les nœuds et une haute cohérence entre les décisions de chaque nœud.

Nous pensons qu'une procédure de maintenance pour les réseaux *Ad hoc* devrait de préférence être localisée. Nous avons proposé une procédure tirant profit des avantages des algorithmes centralisés et distribués. Dans ce cas, la procédure est distribuée, mais nous n'échangeons les informations qu'avec des nœuds à une distance bornée du CH, et éventuellement avec un nombre borné de nœuds du réseau. Ainsi, l'information sur le

réseau étant partielle, moins de trafic de contrôle est requis, remplissant par conséquent la contrainte de réduire le traitement exigé par les nœuds, ce qui permet de maximiser leur durée de vie et leur rendement.

Pour faciliter la gestion et la maintenance des clusters [Vuong et Huynh (2000)], nous avons construit des clusters où tous les nœuds sont à un saut de leur CH. Notre objectif est de maintenir un nombre réduit de clusters stables tout en minimisant le nombre d'invocations des mécanismes de formation et de maintenance ainsi qu'en maximisant le niveau de QoS, la durée de vie, la stabilité et la connectivité des nœuds dans le réseau.

4.3.2 Procédure de découverte de voisinage

La découverte de voisinage se fait par l'envoi des messages *hello* ayant un TTL = 1 et portant le poids du nœud source du *hello*. Tout nœud maintient un « *intraclusters_table* » lui permettant de découvrir la liste de tous les voisins à un saut ainsi que leur poids associé. Si le nœud s'aperçoit qu'il détient le meilleur poids, il copie son *id* dans le champ id_{CH} de son état, sinon ce champ est mis à la valeur par défaut (*NULL*).

Tout nœud v_i non CH ayant (id_{CH} = NULL) peut passer par plusieurs états durant son cycle de vie dans le réseau. Ces états, représentés par un diagramme d'états illustré sur la figure 4.1, peuvent être les suivants :

- écoute du canal;
- isolé;
- *clusterhead* d'un cluster ayant plusieurs nœuds;
- *clusterhead* d'un cluster vide (*one-node cluster*);
- membre d'un cluster quelconque.

En effet, un nœud non CH doit écouter le canal radio pour détecter les activités avoisinantes. Si le nœud s'aperçoit que le milieu est désert (manque d'activités), il entre dans l'état « isolé » avant de recommencer à essayer ultérieurement.

Le nœud, percevant quelques activités dans son entourage, envoie une requête d'adhésion (*Join*) et attend les réponses de la part des clusters existants. S'il reçoit au moins une réponse positive (*accept*), le nœud s'attache à un cluster existant et entre directement dans l'état « membre ». Si le nœud ne reçoit que des réponses négatives (*reject*), le nœud doit se déclarer *clusterhead* en créant son propre cluster formé de soi-même. Un lien interclusters sera établi à cette fin entre ce nœud et un de meilleurs *clusterhead* avoisinants (voir section 4.3.3).

En étant membre, le nœud peut, soit migrer vers un autre cluster existant parce qu'il se déplace en dehors de son propre cluster ou parce qu'il détecte d'autres clusters plus puissants (voir section 4.3.7), soit devenir « *clusterhead* » parce qu'il détient un meilleur poids dans le cluster où il est membre (voir section 4.3.8).

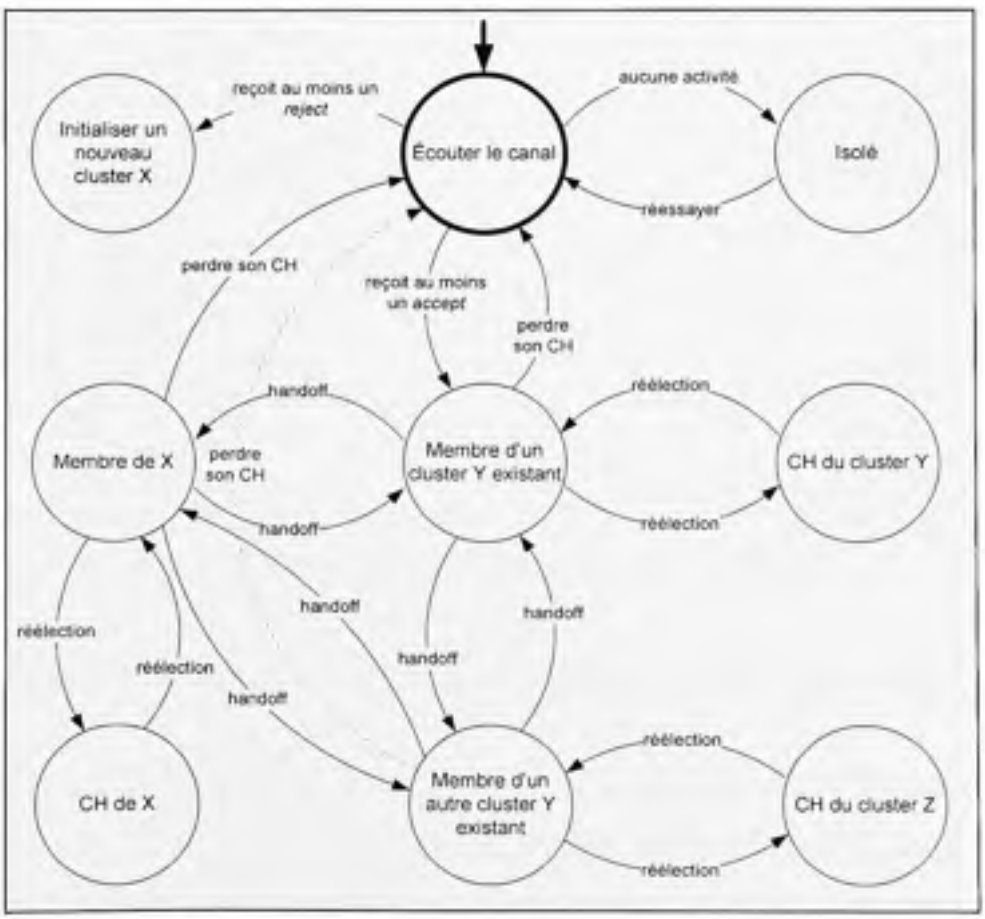

Figure 4.1 ***Diagramme d'états d'un nœud lors de la maintenance.***

Les schémas (1), (2) et (3) de la figure 4.2 montrent un exemple illustratif de découverte de voisinage, de détermination et d'interconnexion des CH.

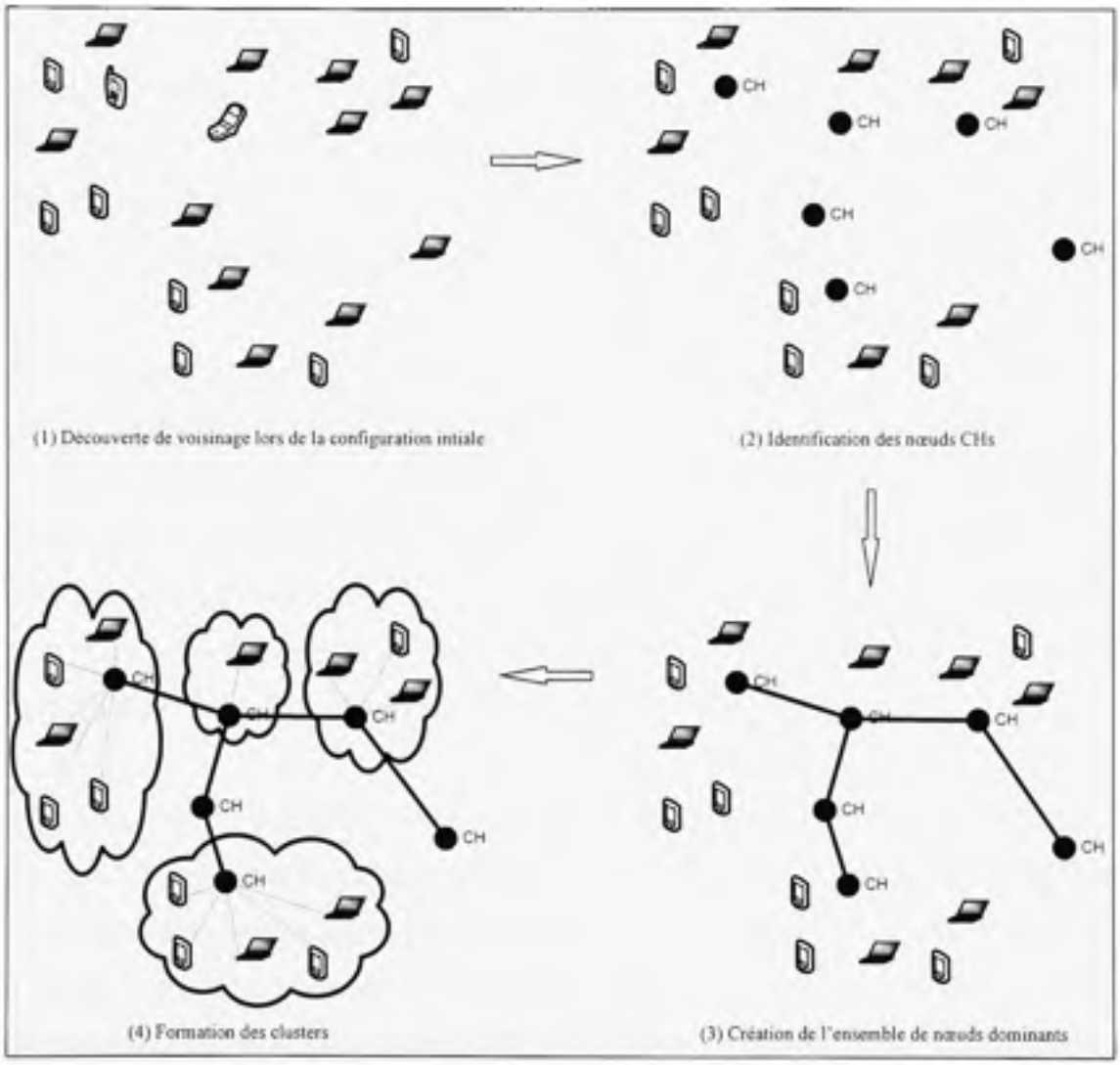

Figure 4.2 ***Panorama de construction des 1-clusters.***

4.3.3 Procédure d'adhésion de nouveaux nœuds à des clusters existants

Tout nœud v_i non CH doit chercher un cluster pour s'y attacher. Il commence par diffuser une requête *Join* (ayant un $TTL = 1$) sur le canal en utilisant le code de signalisation CDMA « *sig_code* » afin de joindre le *clusterhead* le plus puissant et situé à un saut de lui. Puisqu'il n'appartient à aucun cluster, le nœud doit surveiller et écouter constamment le

canal afin de détecter les réponses à sa requête. Durant cette période de temps, le nœud est dans un état instable (*écoute*) : son état n'est pas complètement identifié en attendant de remplir les champs manquants ($id_{CH}, intracluster\_code, compteur$).

Seuls les *clusterhead* ont la responsabilité de traiter les requêtes *Join*. Les nœuds membres doivent ignorer toute requête *Join* interceptée sur le canal. Si le nœud non CH ne reçoit aucune réponse positive (*accept*) ou négative (*reject*) pendant un laps de temps, il considère le milieu désert et se déclare « isolé » avant de recommencer après une période de temps.

À la réception d'une réponse (*accept* ou *reject*), le nœud doit se préparer à s'attacher à un des clusters candidats avant de prendre une décision définitive et immédiate. Ceci lui permet de collecter autant de réponses pendant un laps de temps afin qu'il puisse juger adéquatement au sujet du meilleur CH à joindre. Notons que la réception d'une réponse positive ne veut pas dire que le CH a complètement accepté l'adhésion de ce nœud (voir section 4.3.6). En effet, cette réponse est conditionnelle à la confirmation du nœud par l'envoi d'un message *Join_accept* signalant son choix définitif à ce sujet.

D'autre part, il est fort probable que le nœud reçoive plusieurs messages *accept*; dans ce cas il doit choisir celui ayant le meilleur poids w_i. Ensuite, il confirme son choix en envoyant au CH choisi un *Join_accept* sur le canal de signalisation et attend un message *CH_ACK* de la part de ce dernier. Le message *CH_ACK* n'est qu'une confirmation définitive au sujet de l'adhésion du nœud et comprend un code CDMA intraclusters que le nœud doit utiliser pour communiquer ses données avec le CH. C'est à ce stade que le CH crée une entrée pour ce nœud dans sa table « *intraclusters_table* » et incrémente son compteur. Le nœud entre dans l'état « membre » et identifie les champs ($id_{CH}, intracluster\_code, compteur$).

Nous voulons avec cet échange (4 *handshaking)* garantir qu'aucun CH autre que celui choisi comme meilleur candidat n'ait ajouté ce nœud dans sa table. Ce qui permet d'éviter tout conflit dans le réseau.

Dans le cas où le nœud ne reçoit que des réponses négatives *reject*, il doit sélectionner le CH source du message *reject* ayant le meilleur poids. Le nœud communique avec ce dernier par l'envoi d'une requête *CH_request* signalant son désir de se déclarer en tant que nouveau CH dans le réseau et attend pour une réponse *CH_response*. De cette façon, ce nœud peut jouer le rôle d'une passerelle (*gateway*) permettant à d'autres nœuds de s'y attacher. Le schéma (4) de la figure 4.2 montre un exemple illustratif de la procédure d'adhésion des nœuds. Plus formellement, cette procédure s'écrit :

```
for all (v_i ∈ V, id_CH(v_i) = NULL) do /* V étant l'ensemble de nœuds */
    Start monitoring the channel;
    if (no activity)
      The node declares itself as an isolated node and retries the algorithm later;
    else
      Broadcast Join (id_vi, other fields = NULL);
      Wait during a time interval for a message accept or reject;
      if (there is neither accept nor reject)
        The node declares itself as an isolated node and retries the algorithm later;
      else
        Store during a time interval all the received accept and reject in a vector [ ];
      end
    end
    Search in vector [ ] the accept which has the best weight;
    if (accept is found)
      Send Join_accept to the CH which is the accept's source and wait for CH_ACK from the chosen CH;
      if (CH_ACK is received)
        The node declares itself as a member of the chosen cluster and fulfills its state;
      else
        The node tries with next best weight CH selected from the vector [ ];
      end
    else
      The node chooses the reject which has the best weight and decides to declare itself as CH;
      Send CH_request to the CH which is the reject's source and wait for CH_response;
      if (CH_response is received)
        compteur = 1;
        id_CH = id_vi; /* The node declares itself as CH */
      else
        The node tries with another CH selected from the reject vector [ ];
      end
    end
end
```

4.3.4 Procédure de contrôle d'admission basé sur la QoS

La raison pour laquelle le CH accepte ou rejette une requête *Join* dépend de la capacité de ce dernier à satisfaire les besoins de ces membres. Nous avons proposé un mécanisme de contrôle d'admission basé sur le modèle analytique présenté dans le chapitre 3 afin d'estimer le débit et le délai du réseau. Ce mécanisme sera seulement implémenté sur les CH et appliqué durant la maintenance de sorte que ces derniers puissent prendre des décisions d'accepter et/ou de rejeter des membres dans le but de garantir un certain niveau de QoS aux membres déjà admis. Plus formellement, ce mécanisme s'écrit :

```
for all v_i ∈ C do /* C étant l'ensemble de nœuds clusterhead */
   estimate the intraclusters throughput T;
   estimate the end to end delay d;
   if (new Join is received from a node)
      if (T > seuil or d > seuil)
        reject the Join request;
        send a reject message to the node;
      else
        accept the Join request;
        send a accept message to the node and wait for a confirmation;
        if (Join_accept is received from the node)
            update the counter compteur;
        end
      end
   end
end
```

L'utilité du contrôle d'admission n'apparaît concrètement que lorsque nous faisons les simulations dans le chapitre 5. C'est là où nous aurons en main les vraies valeurs concernant le nombre de nœuds par cluster ainsi que le nombre de clusters produits dans un réseau réaliste tenant compte de la mobilité et de la distribution aléatoire des nœuds (réseau hétérogène).

4.3.5 Procédure de désabonnement des nœuds

Le désabonnement d'un nœud membre peut se faire de deux façons : implicite et explicite. En effet, un nœud membre peut explicitement envoyer un message de type *leave* signalant à son CH son intention de quitter le cluster pour des raisons quelconques. Le désabonnement peut également se faire implicitement et être déclenché par le CH. Dans ce cas, le CH ne recevant pas de messages de rafraîchissement de type *hello* de la part d'un nœud membre, doit enlever ce dernier de sa table « *intracluster_table* » après l'expiration du temporisateur associé. Dans le deux cas, le CH met à jour ses tables et son état et avertit ses membres et les clusters avoisinants de la nouvelle valeur du *compteur*.

4.3.6 Procédure de maintenance des nœuds *clusterhead*

Un nœud CH ayant ($id_{CH} = id_{v_i}$) calcule périodiquement son poids et l'envoie à ses membres et ses CH avoisinants par le biais des messages *hello* dans le but de mettre à jour les tables *intracluster_table* et *intercluster_table*. Un CH doit également traiter les messages *leave, hello, Join, CH_*request, *CH_response* et *CH_change*. À la réception d'un message *leave* en provenance d'un nœud membre, le CH met à jour la table *intracluster_table* et diffuse un message *hello* à tous ses membres pour leur informer que le *compteur* a été décrémenté.

À la réception d'un message *hello* de la part d'un CH avoisinant, le CH doit mettre à jour la table *intercluster_table*. Si le *hello* est en provenance d'un nœud membre, le CH met à jour sa table *intracluster_table* et compare le poids du *hello* avec le sien. Dans le cas où ce *hello* a un meilleur poids, le CH devra exécuter la procédure de réélection. Nous avons restreint cette procédure aux nœuds CH en vue de faciliter la maintenance et la gestion des clusters. Notons que dans le but de minimiser le nombre d'invocations de la réélection et de maximiser le niveau de stabilité des clusters, nous avons fait de sorte que la réélection n'aboutisse pas toujours à réélire un autre nœud même si ce dernier possède un meilleur poids. Cette procédure sera élaborée dans la section 4.3.8.

D'autre part, si le CH reçoit une requête de type *Join* de la part d'une nouvelle arrivée ($id_{CH} = NULL$) ou d'un nœud déjà membre ($id_{CH} = x$) d'un autre cluster x, le nœud devra exécuter la procédure de migration d'un nœud afin d'accepter ou de rejeter la requête en se basant sur le mécanisme de contrôle d'admission (section 4.3.4) ainsi que sur l'estimation de la mobilité relative du nœud migrateur par rapport au CH. Cette procédure sera élaborée dans la section 4.3.7.

Pour permettre un passage à l'échelle et maximiser la connectivité, si le CH reçoit un message *CH_request* de la part d'un nœud désirant devenir CH (car ce dernier n'a reçu que de messages *reject*), le CH doit accepter la requête sans avoir à vérifier ses capacités. Il envoie un *CH_response* à ce nœud et diffuse un message *hello* à tous les CH avoisinants pour leur annoncer un nouveau CH dans le réseau. Un lien interclusters sera ainsi établi sur le canal utilisant le code exclusif CDMA « *intercluster_code* ». Finalement, si une réélection dans un cluster a eu lieu, le CH doit émettre un message *CH_change* annonçant aux membres et aux CH voisins l'identité du nouveau CH qui vient d'être élu.

4.3.7 Procédure de migration d'un nœud (*handoff*)

Nous avons essayé d'équilibrer la charge entre tous les clusters du réseau en nous basant sur une procédure de gestion des *handoff* donnant plus de flexibilité aux membres en leur permettant de quitter un faible cluster et joindre un autre qui semble plus performant que les leurs. Cependant, les clusters sollicités doivent contrôler l'admission de ces nœuds de sorte que le *handoff* ne se fasse pas au détriment de la qualité de service de leurs membres. Nous pensons que le contrôle d'admission tout seul ne permet que de maintenir un niveau de QoS dans le cluster sans se soucier de leur niveau de stabilité en termes de minimiser le nombre d'invocations des procédures de maintenance suite à des *handoff*.

En effet, il est fort probable qu'un nœud, voulant migrer à un nouveau cluster, possède un poids beaucoup mieux que celui du CH de ce cluster. Par conséquent et dans le but de ne pas exécuter immédiatement la procédure de réélection avant de vérifier si ce nœud

s'approche ou bien continue à s'éloigner du CH, nous avons mesuré la mobilité relative de ce nœud par rapport au CH en nous basant sur l'historique de puissances des signaux reçus. Ainsi, si l'historique montre que la puissance reçue continue à décrémenter, ceci est une forte indication que le nœud est en train de faire un passage temporaire rapide dans le cluster et il vaut mieux l'ignorer en rejetant sa requête quelque soit la décision du mécanisme de contrôle d'admission.

Afin de ne pas créer de surcharge de traitement au niveau des nœuds, nous considérons le cas où les données récoltées sont raisonnablement espacées. L'obtention des données sur la puissance se fait toutes les cinq secondes. Les éléments qui influencent la puissance reçue sont la variation de la distance suite aux mouvements des nœuds ainsi que les caractéristiques du canal de propagation. Le simulateur que nous utilisons tient compte de ces paramètres. Plus formellement, la procédure de gestion des *handoff* s'écrit :

```
for all (v_i ∈ C , id_{v_i} ≠ x) do /* C étant l'ensemble de nœuds clusterhead */
    A = Perform admission control;
    if (Join is received from a new arrival node) /* (id_CH = NULL) */
            if (A is True)
               Send accept to the source of the message Join and wait during a time interval for Join_accept;
               if (no Join accept)
                  Ignore the request;
                  Break;
               end
            else /* A is False */
               Send reject to the source of the message Join;
               Break;
            end
    end
    if (Join is received from a node already member of another cluster x /* (id_CH = x) */
            B = Check if the historic indicates that the power of the received signals does not decrement rapidly;
            if (A is False or B is False)
               Ignore the request and send a reject to the source of the message Join;
               Break;
            end
    end /*Join is received from a member node (id_CH = x) or a new arrival node */
    compteur = compteur + 1;
    Add the node to the intraclusters_table and send a CH_ACK to the new added member;
    Update intraclusters_table and interclusters_table;
end
```

4.3.8 Procédure de réélection des *clusterhead*

La réélection se base sur la valeur des poids échangés dans les messages de contrôle *hello*. Il est primordial de n'exécuter la procédure de réélection que lorsque c'est nécessaire. Ceci permet de minimiser le trafic de contrôle et optimiser l'utilisation des ressources des nœuds (moins de traitement, moins de mémoire et d'énergie).

Si un nouveau CH doit être élu, la procédure doit être flexible et efficace de sorte qu'elle ne perturbe pas le fonctionnement du cluster durant le transfert des bases des données entre l'ancien et le nouveau CH. Les nœuds des autres clusters ne seront pas influencés par ces changements. Une mise à jour (*CH_change*) permet tout simplement d'annoncer l'adresse du nouveau CH à tous clusters avoisinants. Plus formellement, cette procédure s'écrit :

```
for all v_i ∈ C do /* C étant l'ensemble de nœuds clusterhead */
    Verify periodically all weights w_j ∈ intracluster_table;
    if (there is a w_j better than w_i & w_i ≤ threshold) // be ready for reelection;
        The old CH stays CH until the reception of a confirmation;
        Send database_info (old CH) to the new elected node;
        Wait during a time interval for a database_ACK from the new elected CH;
        if (no database_ACK)
          Don't perform the reelection and the CH continues playing its role;
          Break;
        else
          The old CH copies the id_vj in the field id_CH;
          Update intraclusters_table and interclusters_table;
          Send CH_info to the elected node;
          Broadcast CH_change () to the members and the neighboring CHs;
        end
    else
      Don't perform the re-election and the CH continues playing its role;
    end
end
```

4.3.9 Procédure de maintenance des nœuds membres

Après avoir joint un cluster, le nœud pourra remplir tous les champs de son état. Il calcule périodiquement son poids et l'envoie à son CH dans des messages *hello*. Un membre doit

être capable de traiter seulement les messages *hello*, *reject*, *CH_info*, *database_info* et *CH_change* reçus de la part des nœuds CH. Ceci permet d'optimiser les ressources du réseau (bande passante, batteries, traitement, etc.) et de réduire le nombre de tâches que les nœuds membres doivent exécuter.

Si le message *hello* provient de son nœud CH, le nœud devra mettre à jour sa table *intracluster_table*. Si ce message provient des autres CH avoisinants (il porte un id_{CH} différent), le nœud devra exécuter la procédure de *handoff* à un autre cluster dans le cas où la source du message *hello* détient un poids plus fort que celui du CH courant. Dans ce cas, le nœud doit envoyer une requête *Join* au nouveau CH candidat sans se détacher de l'ancien cluster tant qu'il n'a pas reçu un message CH_ACK du nouveau CH. À ce moment, le nœud envoie un message *Leave* à son CH pour lui annoncer explicitement son intention de quitter la zone. Si le nœud reçoit un *reject* comme réponse à sa requête de migration, il ignore ce message et maintient son abonnement avec son cluster actuel.

Comme nous allons le voir dans les simulations du chapitre 5, cette manière de gestion et de maintenance permettra sans doute de maximiser le niveau de stabilité, d'équilibrer la charge entre les différents clusters ainsi que de minimiser le nombre de clusters à former dans le réseau du fait que le nœud ait la possibilité de migrer à un cluster plus stable et puissant sans avoir à créer son propre cluster et à partitionner davantage la topologie.

Un nœud membre doit mettre à jour son état suite à la réception d'un message de type *CH_change* annonçant aux membres l'identité du nouveau CH élu dans le cluster. Ce nœud peut également recevoir des messages de type *CH_info* résultant de la procédure de réélection exécutée par le CH responsable de la gestion de sa zone. Dans ce cas, le nœud doit se préparer pour remplacer l'ancien CH.

Si un nœud membre ne reçoit pas de *hello* de la part de son CH pendant un laps de temps, il doit considérer que le CH est tombé brusquement en panne. Par conséquent, la procédure de

découverte doit être réinitialisée afin de trouver un nouveau cluster d'attache. Plus formellement, cette procédure s'écrit :

```
for all (v_i ∈ V, id_CH(v_i) ≠ NULL) do /* V étant l'ensemble de nœuds */
    Calculate periodically its weights w_i;
    Send periodically hello messages to its CH;
    if (no hello from the CH during a time interval)
      The CH is down;
      Perform Procédure de découverte;
    end
    if (CH_change is received from the CH)
      Copy the id of the new CH in the field id_CH;
    end
    if (CH_info is received from the CH)
       Copy the id_vi in the field id_CH; /* The node became a CH */
    end
    if (database_info is received as a result of the reelection procedure)
      Prepare itself to become a new CH in the cluster;
      Store received database_info received from the old CH;
      Send database_ACK to the old CH and wait for CH_info;
    else
      if ( id_CH(hello) = id_CH(v_i)) /* the hello is received from its CH */
        Update intraclusters_table;
      else /* the hello is received from another CH */
        if weight (hello from another CH) > weight (last hello from current CH)
          Ignore this hello;
        else /* Possibility to migrate to a stronger CH */
          Send Join to that stronger CH while staying as a member of the current CH;
          Wait for CH_ACK from that stronger CH during a time interval;
          if (reject is received from the stronger CH)
            Ignore the hello;
          end
          if (no CH_ACK is received)
            Ignore the hello;
          else
            Send a leave message to the actual CH;
            Copy the id of that CH in the field id_CH; /* The node became member of a new CH */
          end
        end
      end
    end
end
```

4.4 Conclusion

Un réseau *Ad hoc* est formé d'un ensemble de nœuds libres de se mouvoir dans un environnement dynamique. La clustérisation doit fournir une solution générique pour simplifier le déploiement de tout service réseau. Les clusters doivent être stables dans le temps pour remplir tous les objectifs de gestion, de routage, d'adressage, d'allocation de bande passante et de réduction des inondations, formant une structure virtuelle qui doit dans la mesure du possible rester inchangée. Cette structuration des réseaux *Ad hoc* doit être complétée par une maintenance adéquate des clusters, répondant aux différentes contraintes de ces réseaux.

Dans ce chapitre, nous avons présenté une nouvelle procédure de maintenance des *1-clusters*. Nous avons décrit tous les algorithmes employés durant la maintenance. Pour réagir de façon appropriée, il est primordial que cette procédure soit robuste à la dynamique du réseau et permette un passage à l'échelle dans le sens où elle doit faciliter l'implémentation des protocoles de routage hiérarchiques et garantir une connectivité maximale. La maintenance et les informations requises à une décision étant locales, la réactivité des procédures sera accélérée, permettant une meilleure adaptation aux changements de topologie.

Toutefois, la maintenance pour être efficace, ne doit pas permettre une baisse drastique de performances quand le nombre de nœuds constituant le cluster augmente. Ce nombre évoluant dans le temps, nous devons donc trouver des moyens pour paramétrer les métriques utilisées et mieux les adapter aux conditions environnementales. Cette adaptation doit être transparente et dynamique. Ainsi, nous avons implémenté le contrôle d'admission dans nos procédures pour que les structures soient passables à l'échelle et la qualité de service soit respectée vis-à-vis de la taille des clusters. Dans le chapitre suivant, nous allons implémenter notre modèle et faire une étude de performances tenant compte de la mobilité des nœuds afin de vérifier l'efficacité et la stabilité de nos structures dans un environnement réaliste.

CHAPITRE 5

IMPLÉMENTATION ET ÉTUDE DES PERFORMANCES

5.1 Introduction

Nous avons présenté jusqu'à présent le modèle de clustérisation des réseaux sans fil *Ad hoc* que nous proposons pour répondre aux besoins spécifiques expliqués dans la problématique de recherche. Un des paramètres essentiels à considérer dans ces réseaux est la mobilité. Il est primordial que les solutions proposées tiennent compte du modèle de mobilité qui décrit mieux le mouvement des nœuds.

Un nœud est libre de se déplacer, les liens inter-nœuds changent continuellement selon leur position. Les *1-clusters* étant formés d'un regroupement de nœuds situés à un saut de leur *clusterhead*, il peut y avoir beaucoup de changements dans la topologie intraclusters, provoquant ainsi des périodes d'instabilité dans tout le réseau. Un autre facteur à considérer et qui a été pris en considération dans notre modèle est celui du niveau d'énergie des nœuds. Un nœud peut disparaître à tout moment de son cluster non seulement parce qu'il a migré vers un autre cluster, mais également à cause d'une batterie morte. Certes, la situation s'empirerait si c'était un nœud *clusterhead*.

Nous croyons que le facteur de stabilité est un élément primordial pour mesurer les performances d'un modèle de clustérisation. Nous allons présenter dans ce chapitre, une étude complète des performances de notre modèle basée tant sur des analyses numériques que sur des simulations réalisées dans un environnement réaliste. Finalement, nous allons faire une comparaison entre notre modèle et un autre modèle *1-clusters* existant dans la littérature.

5.2 Modélisation

5.2.1 Préliminaires

Actuellement, IEEE 802.11 constitue *de facto* le standard le plus utilisé pour gérer les accès sur le canal radio, présentant un débit et un rayon de portée acceptables pour de nombreux types d'applications. De ce fait, nous avons opté pour la couche MAC du standard IEEE 802.11b en mode DCF et sans RTS/CTS, utilisant une technique de modulation de type DSSS dans la bande ISM. Cependant, il est à noter que rien n'interdira dans le futur d'intégrer une couche MAC différente, plus adaptée aux réseaux sans fil multisauts. Ainsi, nous nous sommes basés sur le standard 802.11b pour choisir les paramètres de la couche physique et MAC, illustrés dans le tableau 5.1.

Tableau 5.1

Paramètres utilisés dans l'analyse numérique

Paramètres	Valeur
MAC Header	272 bits
PHY Header	192 bits
ACK Length	112 bits
Data Transmission Rate ϑ	1 Mbit/s
Propagation Delay (ε)	1 μsec
ACK_Timeout	212 μsec
SIFS	10 μsec
DIFS	50 μsec
Time Slot γ	20 μsec
CW_{min}	32 slots
CW_{max}	2048 slots
***backoff* stage** (m)	6

En effet, ce standard fournit des débits allant jusqu'à 11 Mbit/s dans environnement sans fil ouvert. Ces débits chutent de 11 Mbit/s (portée de 30 m) à 1 Mbit/s (portée de 90 m) si nous tenons compte des obstacles et des différentes perturbations qui peuvent influencer les

ondes électromagnétiques. Nous avons supposé dans nos calculs que le débit maximal ϑ offert dans un environnement non ouvert est de 1 Mbit/s. Il est à noter que nous avons évité l'utilisation du mécanisme RTS/CTS pour ne pas introduire plus de congestion dans le réseau *Ad hoc* comme l'ont montré [Ray *et al.* (2003)].

5.2.2 Outil de simulation

La génération des scénarios est un vrai défi dans un réseau *Ad hoc*, surtout lorsque nous voulons intégrer la mobilité et un environnement réaliste caractérisant le canal de propagation radio. Nous avions le choix entre différents outils de simulation. OPNET est un des outils les plus répandus dans le monde de simulation des réseaux de télécommunications. Toutefois, [Agba *et al.* (2006b)] ont montré que la couche physique d'OPNET, ne tenant compte de tous les phénomènes du canal radio, ne permet de simuler un environnement *Ad hoc* assez réaliste.

Dans le but de mesurer l'efficacité de notre modèle de clustérisation basé sur le contrôle d'admission ainsi que de la procédure de maintenance employée, nous avons choisi le simulateur SMGen de [Agba *et al.* (2006a)]. Ce simulateur nous fournit un outil de simulation très puissant facilitant la spécification et la définition des scénarios *Ad hoc*. Il intègre une variété de modèles de mobilité de la littérature dans un environnement à trois dimensions (3D) tenant compte différentes perturbations qui peuvent avoir lieu sur le canal radio. La figure 5.1 illustre un cas d'utilisation de ce simulateur ainsi que les paramètres employés pour la génération d'un scénario.

Nous pensons que le domaine d'application des réseaux *Ad hoc* serait de plus en plus dans des environnements fermés tels que des salles de conférence et des campus universitaires. Dans ce cas, les nœuds se déplacent sur une surface de simulation de dimension définie suivant un modèle de mobilité à définir dans la section 5.2.3. Nous avons choisi une topologie de simulation carrée de $500\ m \times 500\ m$ où les nœuds se déplacent à une vitesse

de 3 à 10 Km/h pour réfleter un comportement très réaliste de la vitesse aléatoire des nœuds constituant le réseau.

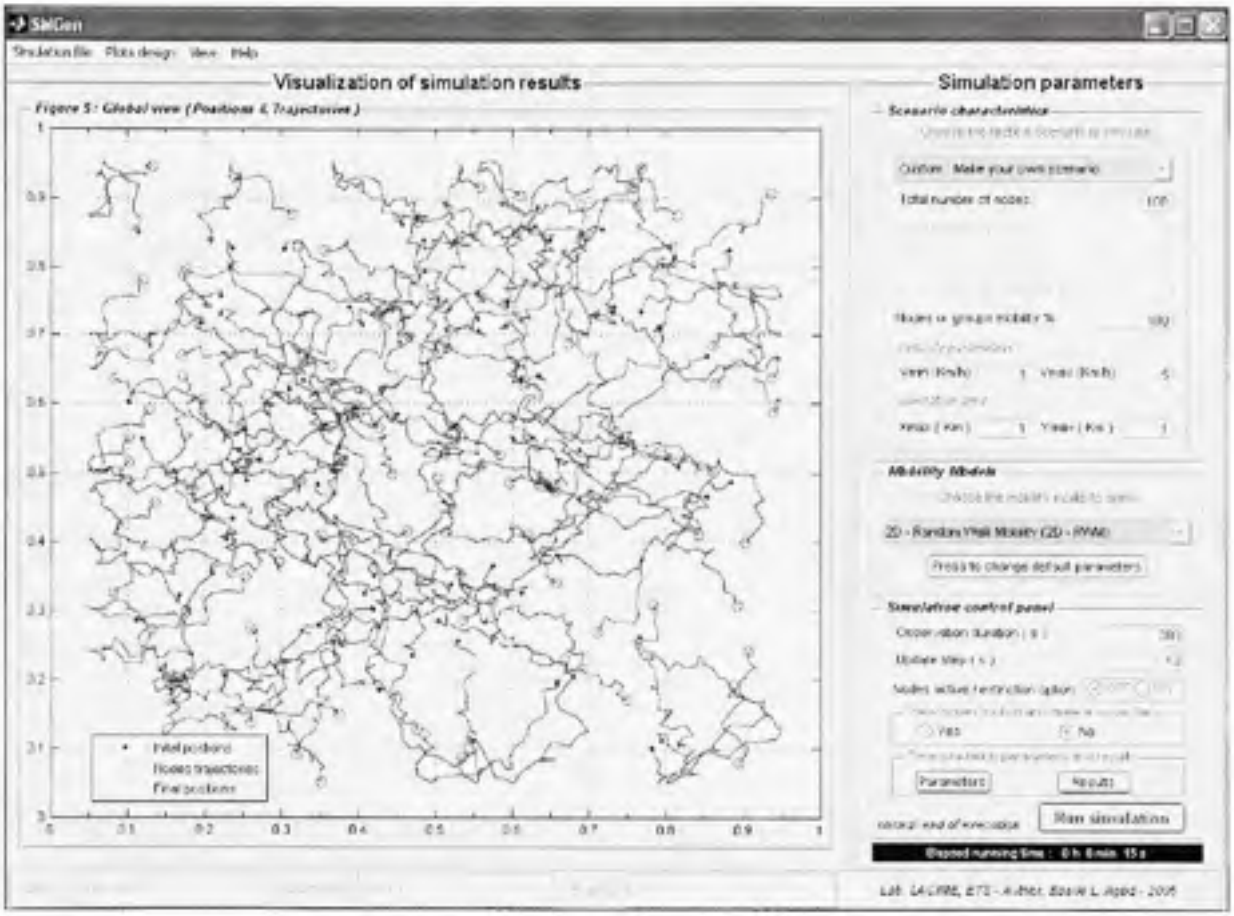

Figure 5.1 ***Exemple de génération d'un scénario mobile Ad hoc dans*** **SMGen.**

La majorité des produits 802.11 de nos jours possèdent une portée radio de 30 à 300 mètres selon la distance et les objets interférents entre les nœuds. Nous avons varié le nombre de nœuds n dans tout le réseau entre 20 et 100. La durée de la simulation est de 300 secondes. La mise à l'échelle (*scalability*) est mesurée en termes de connectivité lorsque nous augmentons le nombre de nœuds dans le réseau.

5.2.3 Modèle de mobilité

Comme nous le savons, il existe dans la littérature une variété de modèles de mobilité permettant de simuler la façon à laquelle les nœuds se déplacent dans un réseau mobile *Ad hoc*. Il n'est pas question ici de présenter tous ces modèles. Cependant, nous référons le lecteur aux travaux de [Camp *et al.* (2002), Zheng *et al.* (2004)] qui ont synthétisé une dizaine de modèles de mobilité tout en décrivant les avantages, les inconvénients et les champs d'applications de chacun d'eux.

Vu sa facilité d'implémentation, le *Random Waypoint Model* (RWM) de [Johnson *et al.* (2007)] est devenu un des modèles les plus utilisés par la communauté de recherche et de simulation. Toutefois, il nous semble impossible de déterminer le modèle de mobilité le plus réaliste pour les applications *Ad hoc* étant actuellement très peu développées. Le jour où ces applications seraient très populaires, il serait beaucoup plus facile de définir les champs d'applications visés par cette technologie et par la suite de déterminer le modèle de mobilité le plus pertinent qui permet de refléter le comportement des nœuds dans le réseau.

Pourtant, nous pensons que le domaine d'application des réseaux *Ad hoc* se développerait de plus en plus dans les applications militaires, les salles de conférence et les campus universitaires. C'est la raison pour laquelle nous avons choisi le modèle RWM pour nos simulations étant donné que nous visons des environnements fermés ayant une surface bien définie. Nous avons ainsi intégré le modèle RWM dans le simulateur afin de simuler un environnement *Ad hoc* très réaliste répondant à nos besoins. Les paramètres qui ont été pris en considération sont la densité ou le nombre total de nœuds dans le réseau, le pourcentage de nœuds mobiles (par défaut à 100%), la taille du réseau, la vitesse minimale et maximale d'un nœud, l'intervalle *hello* des mises à jour, le temps de repos ainsi que la durée de simulation (voir tableau 5.2 dans la section 5.4).

Comme illustré sur la figure 5.2, un nœud, suivant le modèle RWM, choisit d'une façon aléatoire une destination quelconque et une vitesse uniforme (entre 3 et 10 Km/h dans notre

cas). Rendu à cette destination, le nœud s'arrête pendant un temps de repos aléatoire (*Pause Time*) uniformément distribué entre 0 et *max_time*. Par après, le nœud choisit une nouvelle destination aléatoire et se dirige vers elle avec une nouvelle valeur de vitesse uniforme.

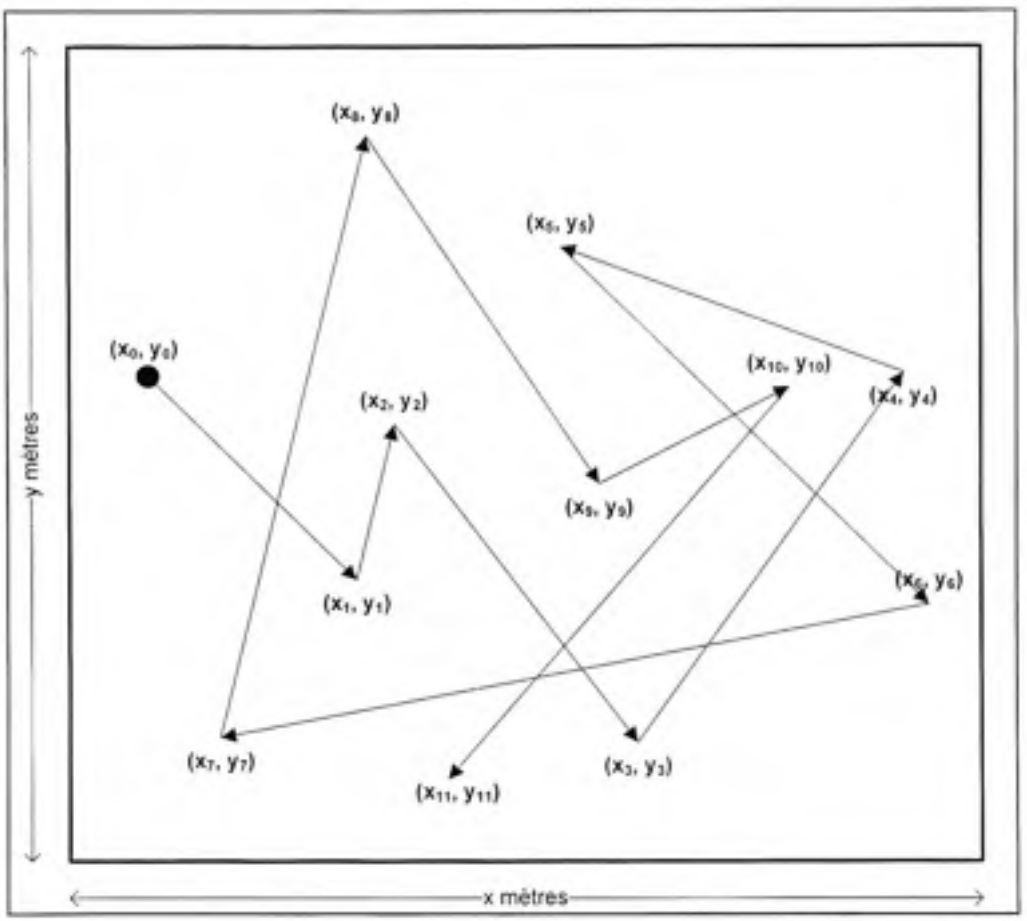

Figure 5.2 ***Mouvement des nœuds en Random Waypoint Mobility Model.***

5.3 Étude de performances des clusters

Dans cette section, nous présentons une analyse numérique basée sur le modèle analytique présenté dans le chapitre 3. L'objectif est de montrer l'effet de la variation du nombre de nœuds par cluster sur les performances des communications intraclusters et sur le niveau de la qualité de service. Ainsi, une bonne estimation de la densité des clusters en termes de nombre de membres nous conduira à ajuster les paramètres de notre modèle de clustérisation. En nous basant sur des simulations, nous allons voir dans la section 5.4 une

étude complète des performances globales du modèle ainsi que de la robustesse des procédures de maintenance présentées dans le chapitre 4.

5.3.1 Étude du débit intraclusters

Nous avons varié le nombre de nœuds par cluster en choisissant des paquets ayant une taille moyenne de 8184 bits (définie dans le standard 802.11). La figure 5.3 prouve que le débit intraclusters dépend fortement du nombre de membres *N* à l'intérieur du cluster.

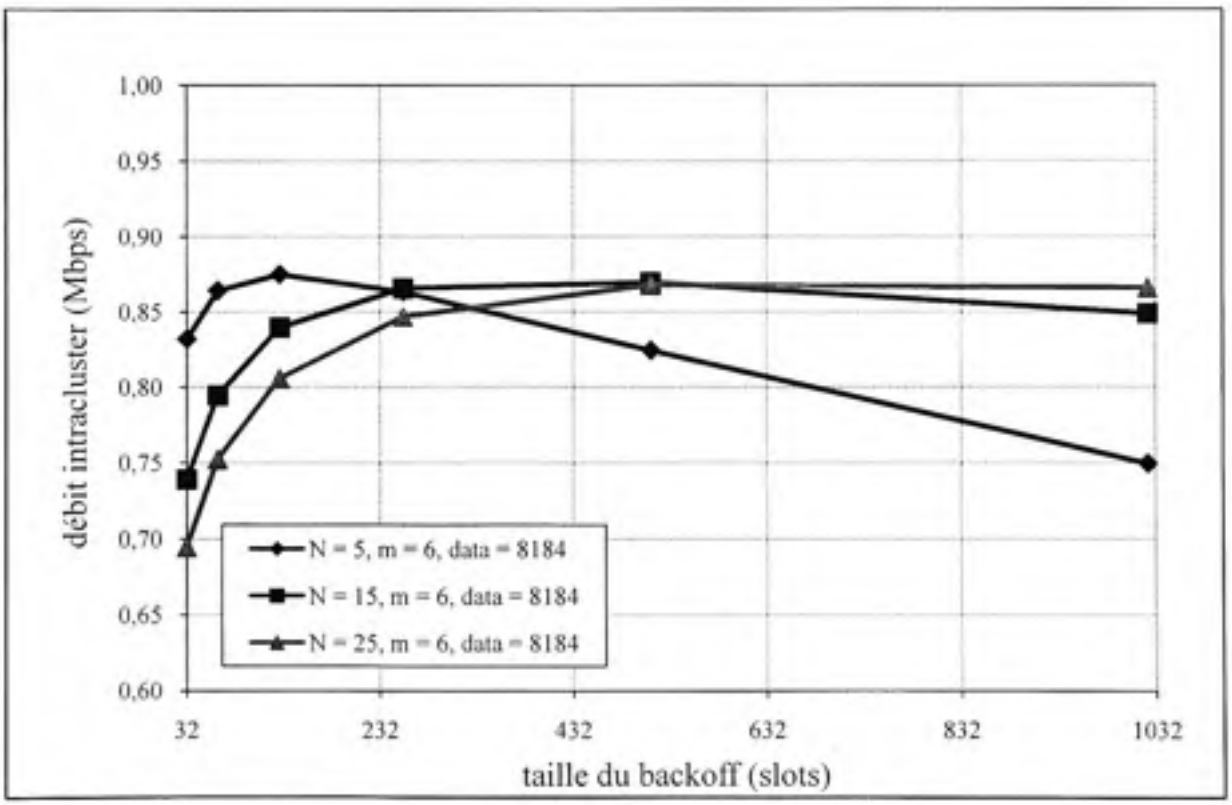

Figure 5.3 ***Débit intraclusters versus taille du backoff.***

En effet, lorsque la taille du *backoff* est petite, la probabilité que les nœuds transmettent simultanément est très grande. Dans ce cas, les nœuds doivent réessayer jusqu'à 6 fois (valeur de *m*) de retransmettre le même paquet (Tous les nœuds du même cluster utilisent le même code CDMA). Lorsque le *backoff* augmente, les risques d'avoir des interférences sur les transmissions sont plus bas. Ce qui permet aux nœuds de transmettre leurs paquets plus

rapidement, et par la suite d'augmenter le débit par nœud. Toutefois, nous remarquons que lorsque nous continuons à augmenter le *backoff*, les nœuds vont passer plus de temps à faire écouler les slots d'attente aléatoire même si le canal est libre. La situation s'empire lorsque le nombre de nœuds est très petit dans le cluster (cas où $N = 5$). Ce qui provoque une sous utilisation de la bande passante disponible dans le réseau.

Nous nous rendons compte que, dans la plupart des cas, le débit se détériore radicalement lorsque le nombre de membres actifs commence à augmenter. Cette condition demeure valide jusqu'à ce que la valeur de la fenêtre W soit approximativement égale à 500 slots. Nous remarquons qu'une valeur plus élevée de W permet d'améliorer le débit dans le cas où le nombre de membres est assez grand, alors qu'elle le pénalise rigoureusement dans le cas où le nombre de membres est petit.

D'autre part, il est clair que lorsque le nombre de membres est petit, il est évident que la probabilité de choisir de larges fenêtres W devient négligeable, puisque la contention est minimisée au maximum. C'est pour cela que nous pouvons ignorer cette situation.

5.3.2 Étude du délai intraclusters

Sous les mêmes conditions, le même comportement peut aussi être remarqué sur la figure 5.4. Lorsque la taille du cluster est grande et la valeur des fenêtres choisies par les nœuds membres est petite, un nœud membre passera beaucoup plus de temps avant de réussir à transmettre correctement son paquet. Lorsque la fenêtre commence à s'élargir, les interférences, dues aux transmissions simultanées, seront réduites; ce qui diminue également le délai d'attente pour transmettre correctement un paquet sur le canal.

D'autre part, nous remarquons que lorsque la fenêtre devient très large, le nœud doit tout d'abord écouler tous les slots avant d'essayer une transmission. Ce qui augmente évidemment le délai. Par contre, la probabilité qu'un nœud choisisse une fenêtre très large est top petite, étant donné que le nombre de membres dans le cluster est assez petit.

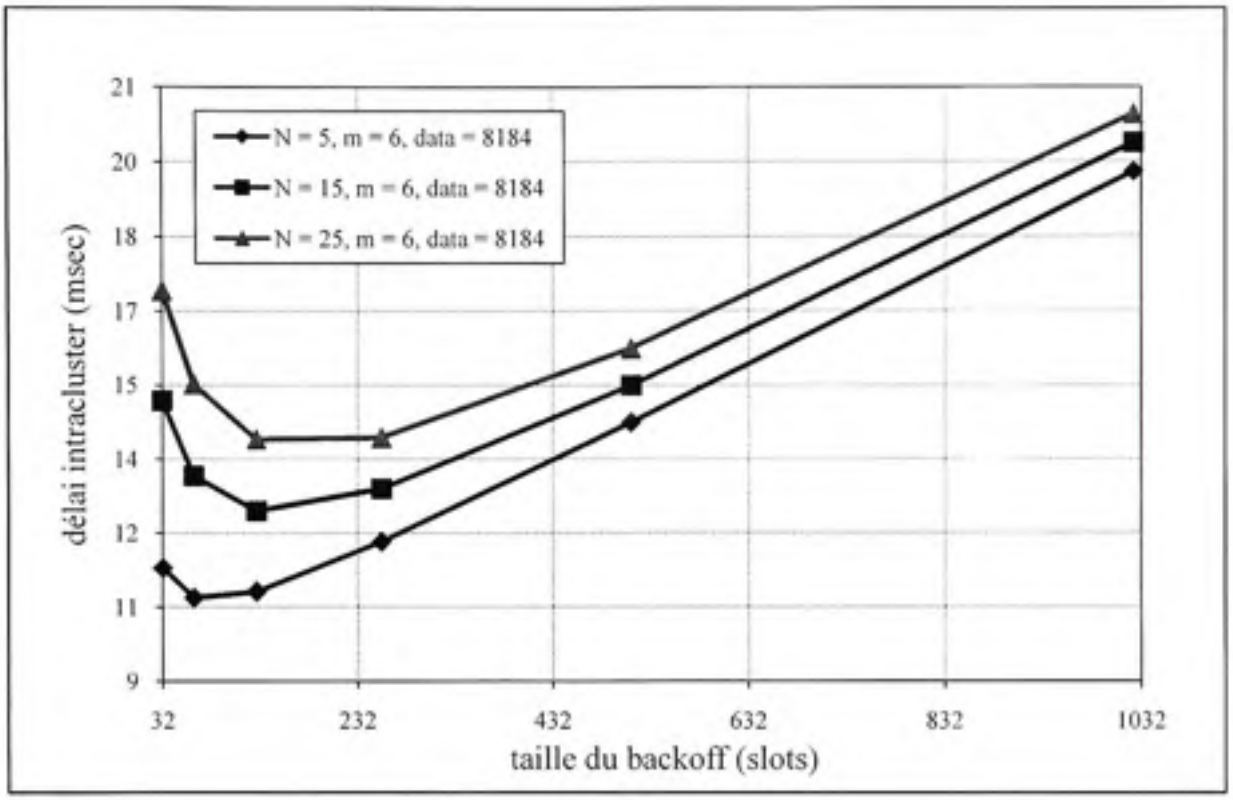

Figure 5.4 ***Délai intraclusters versus taille du backoff.***

5.3.3 Étude du taux de perte de paquets par cluster

Sous les mêmes conditions, figure 5.5 montre le nombre moyen de retransmissions qui ont eu lieu sur le paquet avant qu'il soit correctement reçu; ceci nous donne une indication sur le taux de perte de paquets sur le canal radio. Nous remarquons que le taux de perte augmente rapidement lorsque la taille de la fenêtre de contention décrémente.

La situation est plus pire dans le cas de grands clusters. Ceci est dû à un nombre élevé de retransmissions qu'un paquet doit subir avant qu'il soit correctement transmis étant donné que la probabilité de transmettre dans le même slot soit trop grande lorsque la fenêtre est très petite. La situation se détériore lorsque le cluster est plus dense.

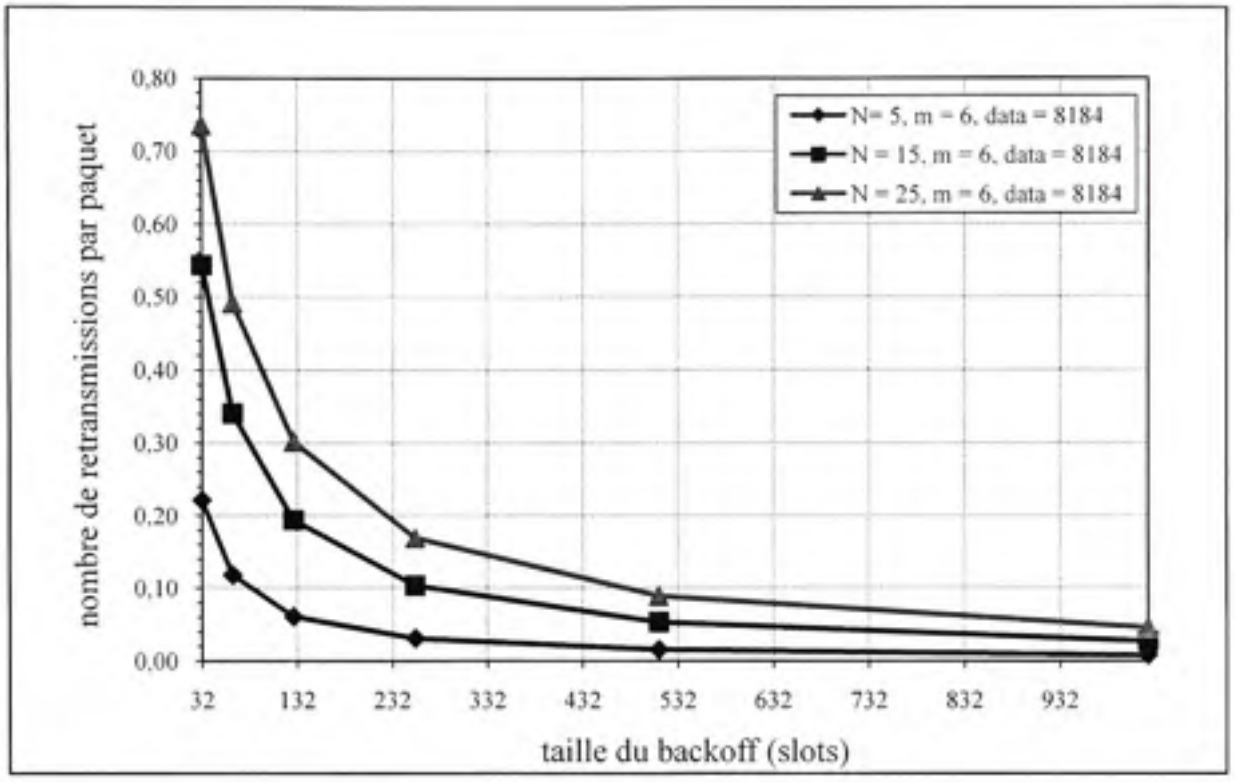

Figure 5.5 ***Nombre de retransmissions par paquet versus taille du backoff.***

5.3.4 Effet de la grandeur des paquets sur les performances des clusters

Il est primordial de vérifier le niveau de qualité de service qui sera offert à des applications temps réel. Nous savons que la plupart des codeurs de voix génèrent des paquets de petite taille au niveau applicatif (de l'ordre de centaines de bits). Nous avons varié à cet effet la taille des paquets au niveau canal entre 500 bits et 11000 bits pour refléter une variété de paquets de différents types d'applications.

Nous avons choisi une fenêtre de 32 slots afin de considérer le pire cas lorsque tous les nœuds du cluster décident de transmettre simultanément après un certain temps de repos. L'objectif est de mesurer les paramètres de qualité de service qui seront offerts au sein des clusters.

5.3.4.1 Effet de la grandeur des paquets sur le débit intraclusters

Comme illustré sur la figure 5.6, nous remarquons qu'en augmentant la taille des paquets, nous commençons à optimiser l'utilisation de la bande passante en s'approchant du débit maximal du canal, fixé à 1 Mbit/s.

Cette situation s'explique par une réduction considérable de temps de transmission des paquets : une fois le nœud a réussi à avoir le canal, il peut commencer à transmettre son paquet quelque soit sa taille. L'*overhead* de temps passé sur DIFS, SIFS et les slots de la fenêtre demeure le même. Finalement, nous pouvons ainsi conclure que plus le cluster est dense, moins le débit est élevé.

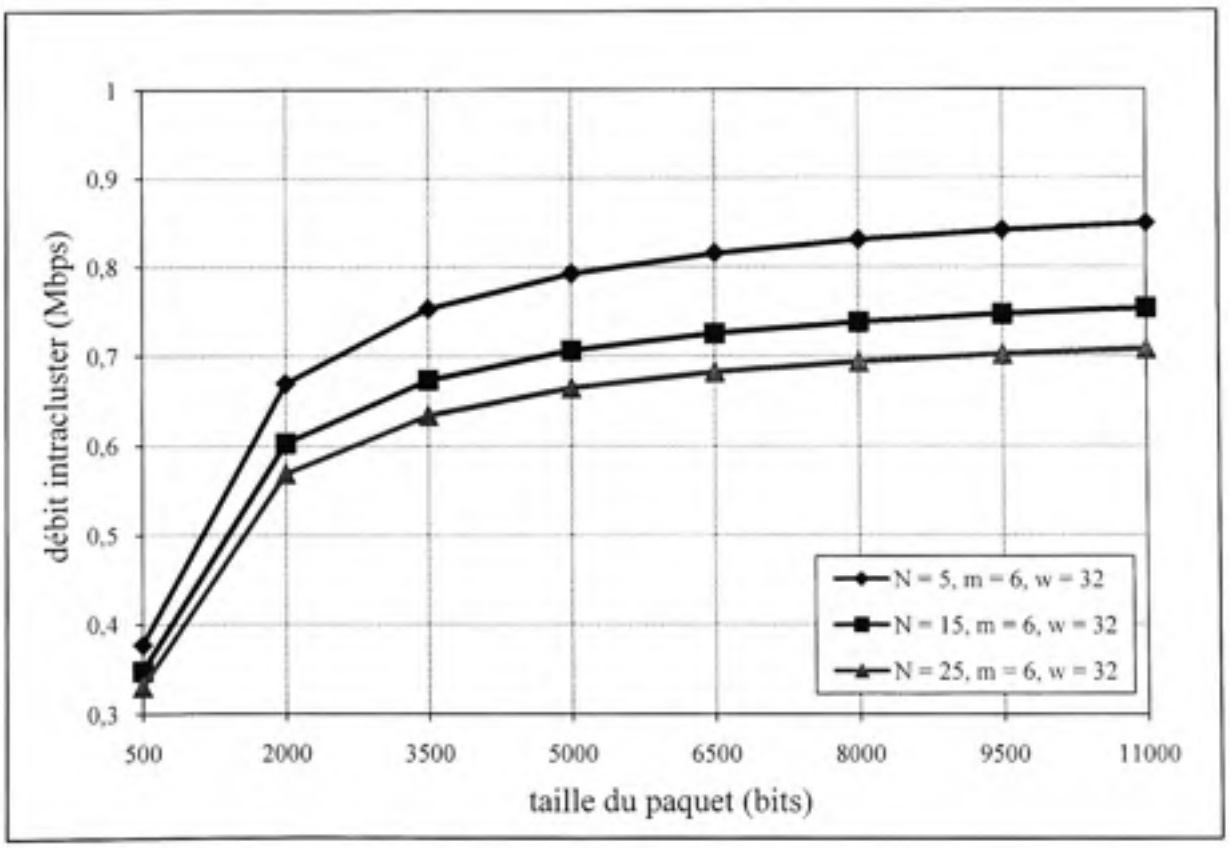

Figure 5.6 ***Débit intraclusters versus taille du paquet.***

5.3.4.2 Effet de la grandeur des paquets sur le délai intraclusters

Le même phénomène peut être vu sur la figure 5.7 en termes de délai qu'un nœud doit attendre avant de réussir à transmettre son paquet. Nous remarquons que le délai incrémente en augmentant la taille des paquets.

Ceci est expliqué par le fait que le temps passé à attendre des slots libres (*backoff*) est beaucoup plus long dans le cas de grands paquets étant donné qu'un long paquet peut monopoliser le canal pendant des périodes plus longues. Nous remarquons également que la situation s'aggrave lorsqu'on choisit des clusters de grande taille.

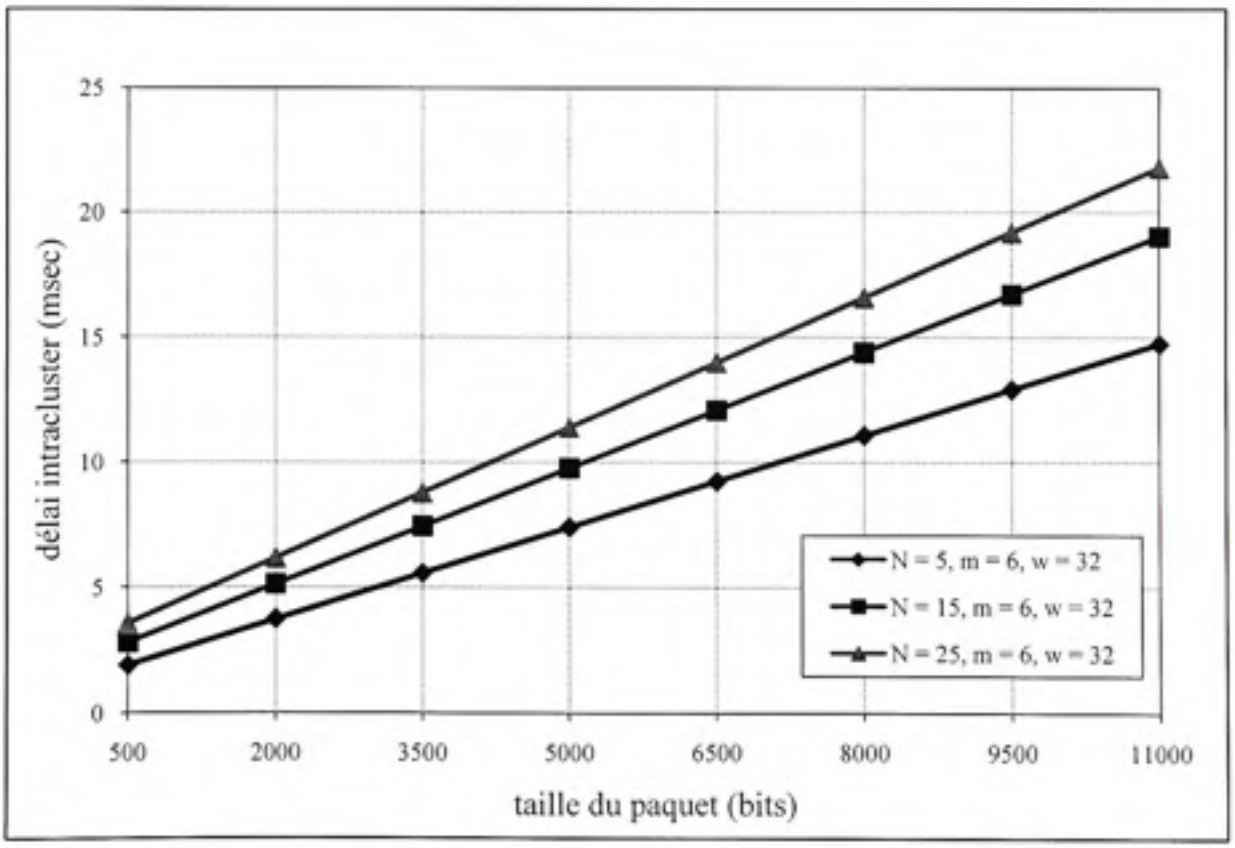

Figure 5.7 ***Délai intraclusters versus taille du paquet.***

5.3.4.3 Effet de la grandeur des paquets sur le taux de perte par cluster

Nous remarquons sur la figure 5.8 que lorsque le *backoff* est petit (32 slots), les chances que les nœuds transmettent simultanément sont grandes. Éventuellement, la situation s'empire lorsque les clusters sont plus denses; mais nous considérons que ce cas ne pourrait pas survenir qu'au début des transmissions simultanées. Après avoir détecté un problème de transmission, les nœuds vont doubler leurs *backoff*, ce qui réduit énormément les interférences et par suite le taux de perte à des valeurs négligeables comme illustré sur la figure 5.5 de la section 5.3.3.

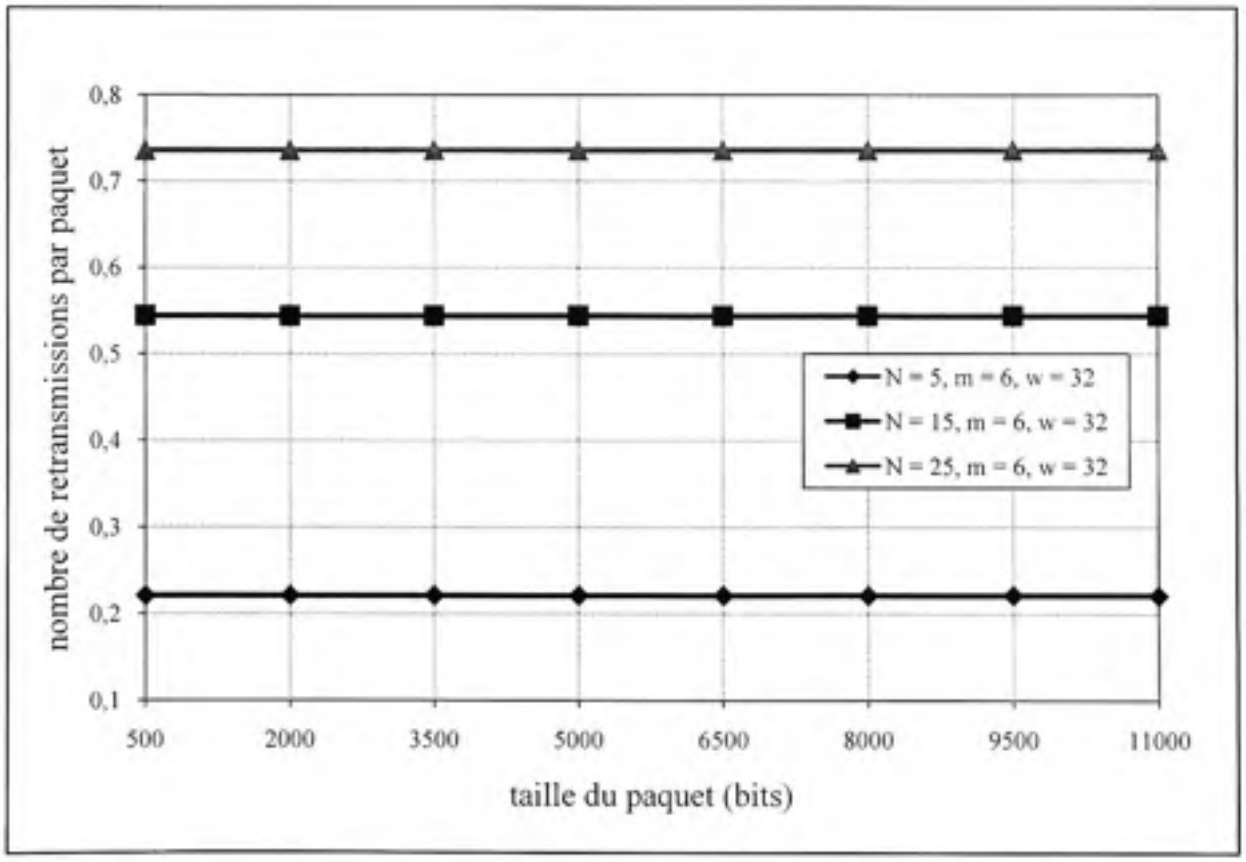

Figure 5.8 ***Nombre de retransmissions par paquet versus taille du paquet.***

En observant la figure 5.8, nous remarquons que le taux de perte n'est pas vraiment influencé par la taille des paquets, mais plutôt par le nombre de nœuds dans le cluster qui essayent de transmettre leur trafic. Lorsque le *backoff* est très petit, les paquets transmis,

utilisant le même code CDMA « *intracluster_code* », vont s'interférer; ce qui augmente énormément le taux de perte selon le nombre de membres actifs par cluster.

5.3.5 Estimation du nombre optimal de nœuds dans un cluster

Comme illustré sur la figure 5.9, nous remarquons que le débit commence à diminuer lorsque la taille du cluster augmente. Toutefois, nous pouvons observer que la courbe reste plus stable quand la taille de la fenêtre est de l'ordre de 512 slots et le nombre de nœuds actifs est entre 15 et 20 nœuds par cluster. Ceci est expliqué par le fait que la probabilité que tous les nœuds choisissent un même slot, parmi les 512 slots, soit trop petite; ce qui augmente le débit jusqu'au point où le cluster devient plus dense (15 nœuds) et la probabilité d'occasionner des interférences commence à augmenter.

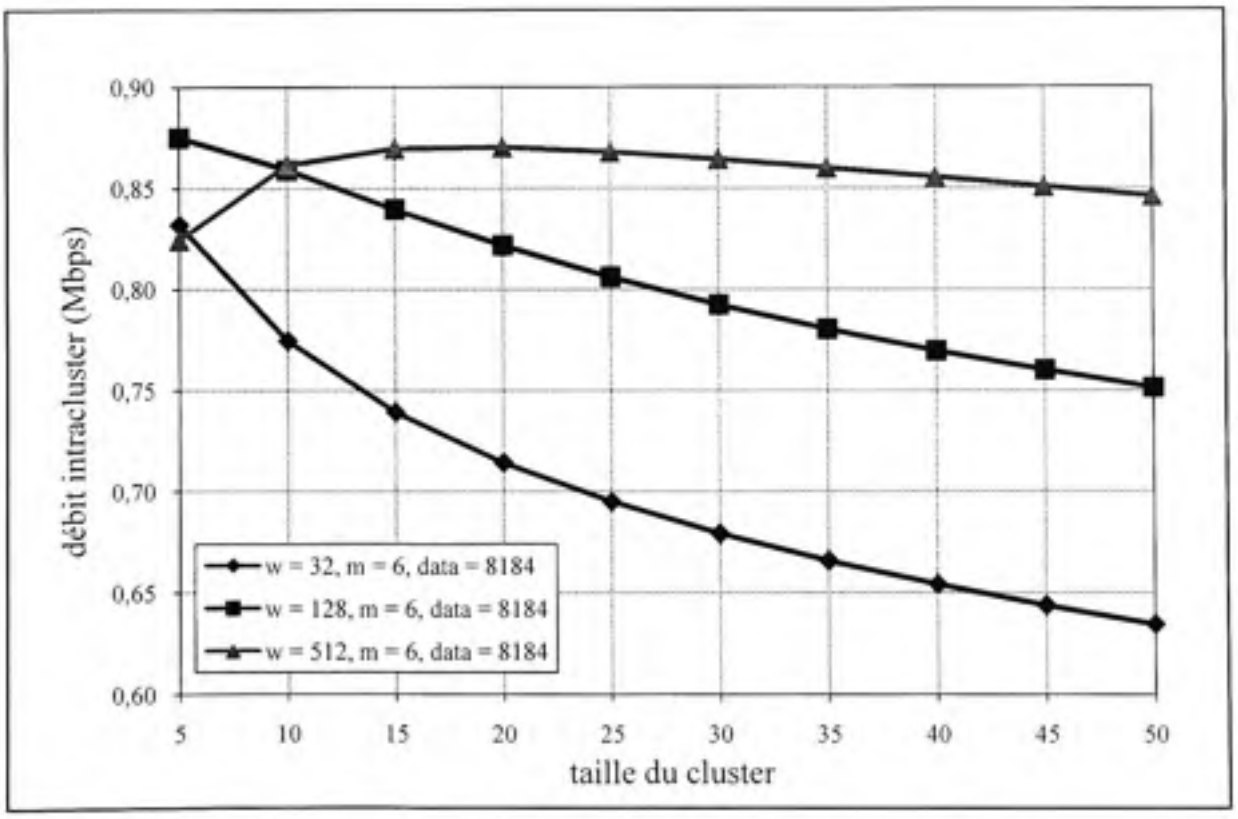

Figure 5.9 ***Effet de la taille du cluster sur le débit intraclusters.***

Pour les mêmes raisons auparavant décrites, nous pouvons observer sur les figures 5.10 et 5.11, que le délai et le nombre de retransmissions augmentent radicalement lorsque le cluster devient plus dense.

Nous remarquons qu'une valeur raisonnable d'une fenêtre de 512 slots permet de maintenir une meilleure stabilité dans les différentes courbes étant donné que les risques d'interférences seront minimisés. De plus, lorsque la taille des clusters est de l'ordre de 15 nœuds, le délai est très acceptable et le taux de perte est presque négligeable pour des paquets ayant une taille de 8184 bits.

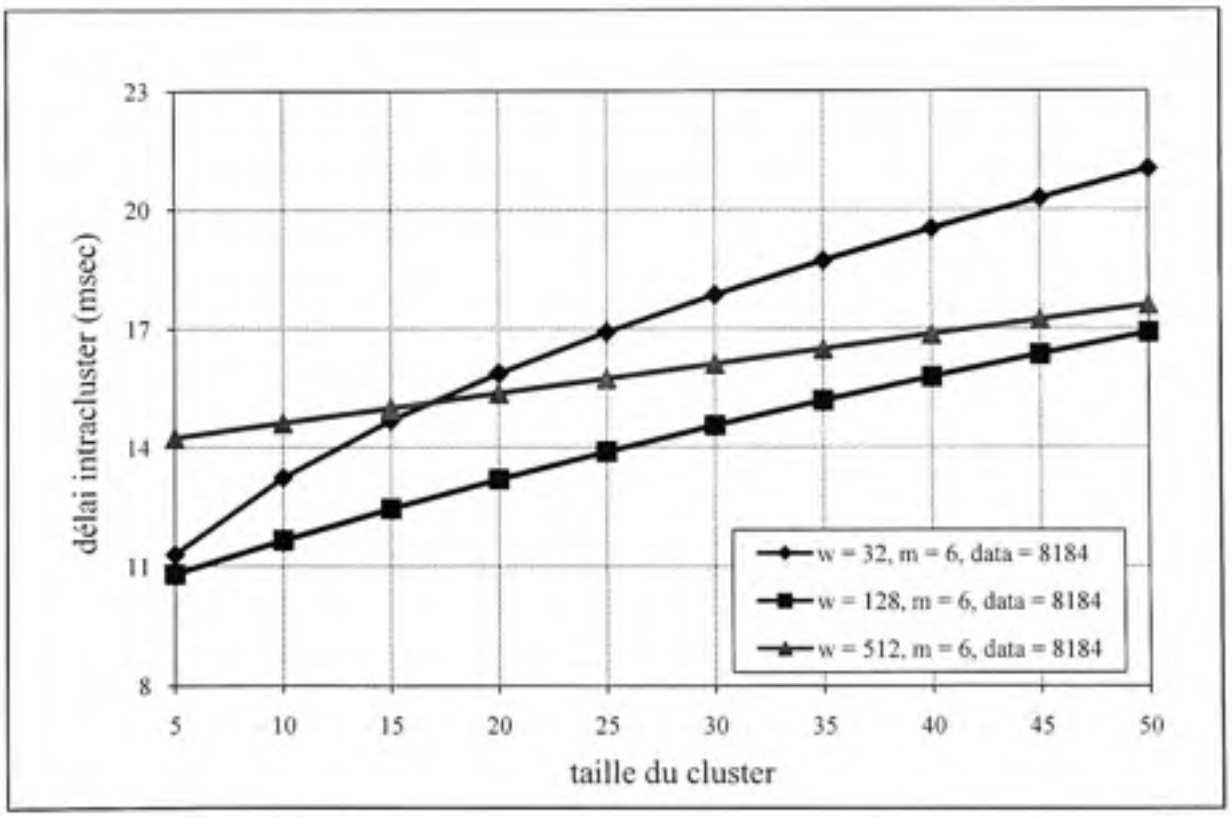

Figure 5.10 ***Effet de la taille du cluster sur le délai intraclusters.***

Il est clair que la clustérisation d'un réseau *Ad hoc* ne doit en aucun cas être limitée à un découpage de la topologie sans tenir compte des spécificités des applications. Il s'agit ainsi d'optimiser le nombre de nœuds par cluster de sorte que nous puissions satisfaire les paramètres de qualité de service des applications utilisées dans les clusters. Cette

optimisation permet en quelque sorte d'équilibrer la charge entre les différents clusters et d'éviter la congestion sur les CH.

Par conséquent, un nœud voyant sa demande de connexion refusée par un *clusterhead*, doit chercher un autre cluster d'attache ayant assez de capacités et pouvant l'admettre. Si aucun *clusterhead* n'accepte sa requête, le nœud doit former son propre cluster et crée un lien interclusters avec un des meilleurs *clusterhead* qui ont refusé sa requête. Rappelons que les communications interclusters se font sur un code CDMA spécifique « *intercluster_code* ».

Ceci permet, d'une part, de minimiser les interférences entre les canaux intraclusters et les canaux interclusters; et d'autre part, d'étendre le réseau et de maximiser la connectivité étant donné qu'en formant un cluster ayant un seul nœud, ce dernier peut être vu comme une passerelle à laquelle d'autres nœuds lointains peuvent s'attacher.

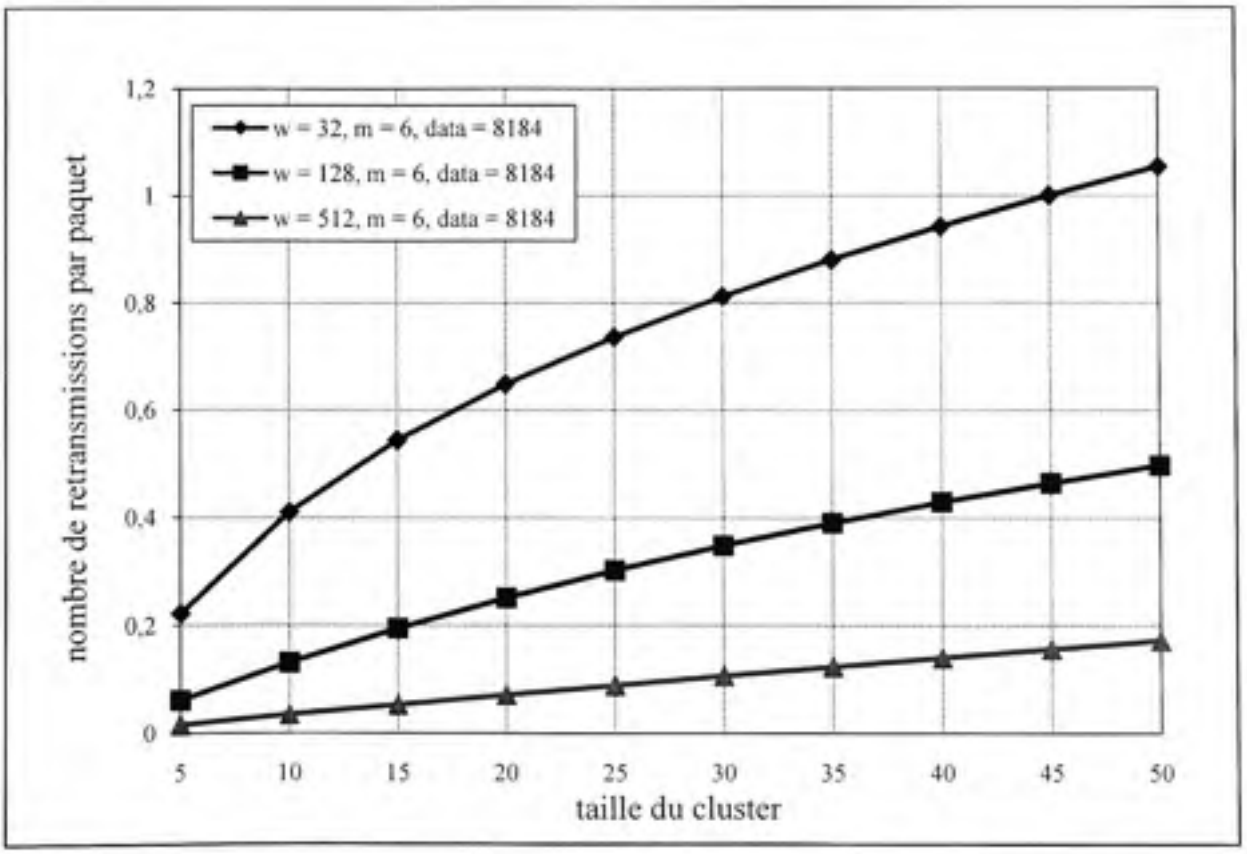

Figure 5.11 ***Effet de la taille du cluster sur le nombre de retransmissions par paquet.***

Le modèle de clustérisation proposée est flexible, il ajuste ses paramètres pour tenir compte des besoins des applications en termes de débit, de délai et de taux de perte. Toutefois, nous allons considérer dans la suite de ce chapitre que le mécanisme de contrôle d'admission va essayer d'établir des clusters ayant un maximum de 15 nœuds actifs à la fois, afin de garantir à nos applications les paramètres de QoS désiré.

5.4 Étude de performances des procédures de maintenance

Nous présentons dans cette section les résultats des simulations effectuées en générant des scénarios avec l'outil SMGen. Nous commençons par présenter les métriques d'évaluation que nous jugeons pertinentes à l'étude de l'efficacité et de stabilité de nos *1-clusters*. Nous avons varié la vitesse des nœuds dans toutes les simulations entre 3 et 10 Km/h. Par conséquent, tous les graphiques obtenus reflètent la présence de la mobilité dans le réseau. Le tableau 5.2 illustre les valeurs des différents paramètres employés dans le simulateur.

Tableau 5.2

Paramètres utilisés dans le simulateur

Paramètres	Valeur
Nombre de nœuds dans le réseau *n*	20 - 100
Taille du réseau	500 $m \times 500\ m$
Vitesse des nœuds *S*	3 à 10 Km/h
Portée de transmission *R*	30 à 300 mètres
Temps de repos (*PT*)	0 sec
Intervalle des *hello*	5 secondes
Bande de fréquence	2.4 GHz
Durée de simulation	300 secondes

5.4.1 Métriques d'évaluation des performances

Afin de réaliser une étude des performances reflétant le niveau de stabilité des clusters et le passage à l'échelle, nous nous sommes intéressés à étudier les métriques suivantes :

- **nombre moyen de clusters :** définit le nombre moyen de clusters qui seront créés dans tout le réseau et nous donne une idée sur le nombre moyen de sauts qu'un paquet va prendre étant donné que le routage est hiérarchique et se fait exclusivement entre les *clusterhead*;

- **nombre moyen d'occurrences à l'état CH :** définit le nombre moyen de transitions qu'un nœud change d'état CH à un état membre; c'est-à-dire le nombre de réélections du nœud en tant que CH durant toute la simulation. Ceci est une forte indication sur la stabilité de l'algorithme de clustérisation utilisé et des procédures de maintenance. En effet, moins de changements sur les CH implique plus de stabilité et moins de coupures sur les routes;

- **nombre moyen de ré-affiliations :** définit le nombre moyen de *handoff* ou le nombre de différents clusters qu'un nœud va joindre durant toute la simulation. Un nœud *Ad hoc* étant mobile et se déplaçant à une vitesse aléatoire, ce dernier peut changer du cluster d'attache à tout moment. Minimiser le nombre de ré-affiliations implique une réduction au niveau des coupures des liens avec le CH et par conséquent une meilleure stabilité des routes;

- **équilibrage de charge entre les différents clusters :** définit le taux d'occupation des clusters dans le temps. En implémentant notre mécanisme de contrôle d'admission, les clusters seront équilibrés. Un CH peut ainsi refuser l'adhésion d'un nœud; ce dernier voyant sa requête rejetée peut chercher un autre cluster d'attache moins chargé. Cette métrique nous donne également une idée sur le degré moyen des CH ou bien le nombre

des membres gérés à tout moment dans le cluster. Ainsi, les paramètres de qualité de service des clusters seront facilement mesurables;

- **connectivité :** définit la probabilité que chaque nœud puisse atteindre tout autre nœud dans le réseau. Cette métrique nous fournit une très bonne indication sur le niveau de connexité globale des routes dans le réseau;

- **mise à l'échelle (*Scalability*)** : Pour mesurer le facteur de mise à l'échelle, nous avons vérifié le comportement global du réseau lorsque le nombre de nœuds grandit. Nous avons remarqué que le modèle réagit bien et les performances sont respectées en termes de nombre de clusters formés, nombre d'occurrences à l'état CH, connectivité des nœuds, et équilibrage de charge entre les clusters;

- **débit, délai et taux de perte :** définit les paramètres de qualité de service à offrir aux applications utilisées dans le réseau. Ayant le vrai nombre moyen de clusters formés dans le réseau, nous allons dorénavant mesurer ces paramètres de bout en bout et non seulement au sein des clusters.

5.4.2 Effet de la densité des nœuds en présence de la mobilité

Nous présentons dans cette section quelques graphiques obtenus par les simulations. Le facteur de mise à l'échelle du modèle est mesuré en variant la densité des nœuds dans le réseau et la portée de transmission des nœuds. Rappelons que la mobilité est toujours prise en considération dans notre modèle. Les nœuds se déplacent constamment à des vitesses aléatoires variant de 3 à 10 Km/h selon le modèle de mobilité *Random Waypoint Model*. Le tableau 5.2 récapitule toutes ces hypothèses.

5.4.2.1 Effet de la portée de transmission sur le nombre moyen de clusters

Figure 5.12 montre qu'en variant la densité des nœuds dans le réseau, la plupart des nœuds ne sont pas dans le rayon de portée l'un de l'autre lorsque la portée de transmission est très limitée (30 mètres). De ce fait, le réseau risque d'être fortement déconnecté surtout dans le cas où les nœuds sont très dispersés dans l'environnement. Ceci oblige les nœuds de déclarer leurs propres clusters, engendrant ainsi un nombre élevé de clusters dans le réseau.

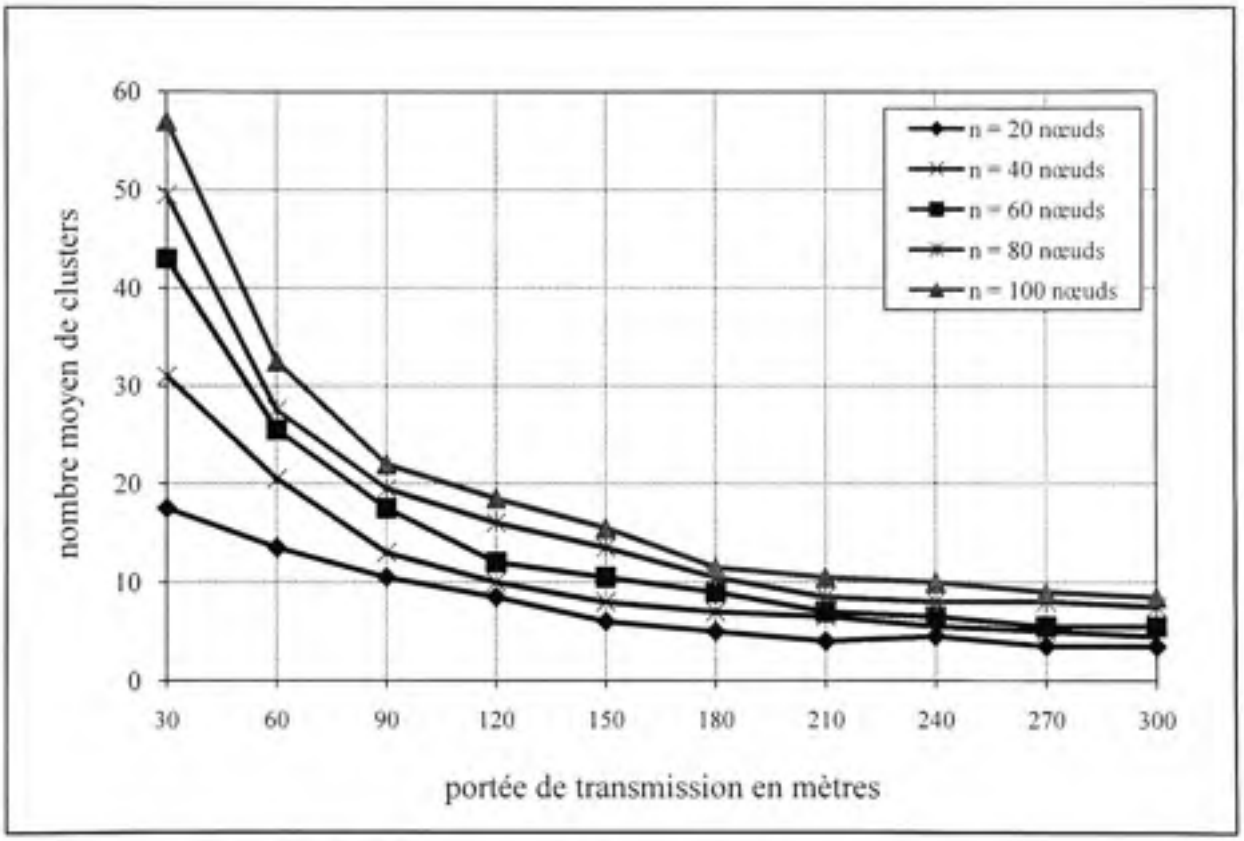

Figure 5.12 ***Nombre moyen de clusters versus portée de transmission.***

En augmentant la portée de transmission, les rayons de portée se chevauchent et les nœuds peuvent se communiquer facilement, réduisant ainsi le nombre de clusters formés étant donné que chaque *clusterhead* va regrouper plusieurs nœuds dans sa portée tant que le mécanisme de contrôle d'admission le permet. Le fait de réduire le nombre de clusters

permet également de minimiser le nombre de sauts dans les routes et par conséquence la quantité des tables de routage à échanger entre les *clusterhead*.

En réalité, une portée de 150 mètres est considérée très raisonnable dans la plupart des appareils de la norme 802.11. Nous remarquons qu'à partir de cette valeur, le nombre de clusters formés tend à devenir stable quelque soit la densité des nœuds dans le réseau. Cette stabilité est expliquée par la meilleure qualité du CH en termes d'énergie, de connectivité, de mobilité, de qualité de service et également par la réduction du nombre de réélections des CH.

5.4.2.2 Effet de la portée de transmission sur les occurrences à l'état CH

Comme illustré sur la figure 5.13, lorsque la portée commence à augmenter, nous remarquons quelques fluctuations très restreintes et négligeables en termes de nombre d'occurrences à l'état CH. Ce qui nous permet de conclure que le modèle est stable et passable à l'échelle même lorsque nous augmentons le nombre de nœuds dans le réseau.

Un des plus importants facteurs de stabilité est celui du nombre de réélections d'un nœud en tant que CH. Ceci nous donne également des indications sur le nombre d'invocations des procédures de maintenance employées dans le modèle. Le moindre d'invocations implique sans doute moins de traitement à faire par les nœuds et une économie extrême de l'énergie consommée.

Ce facteur permet également de mesurer le niveau de stabilité des clusters. La conception du modèle permet aux nœuds de migrer vers d'autres clusters soit volontairement parce qu'ils changent de zone, soit parce qu'ils détectent un autre cluster avoisinant fournissant de meilleures performances.

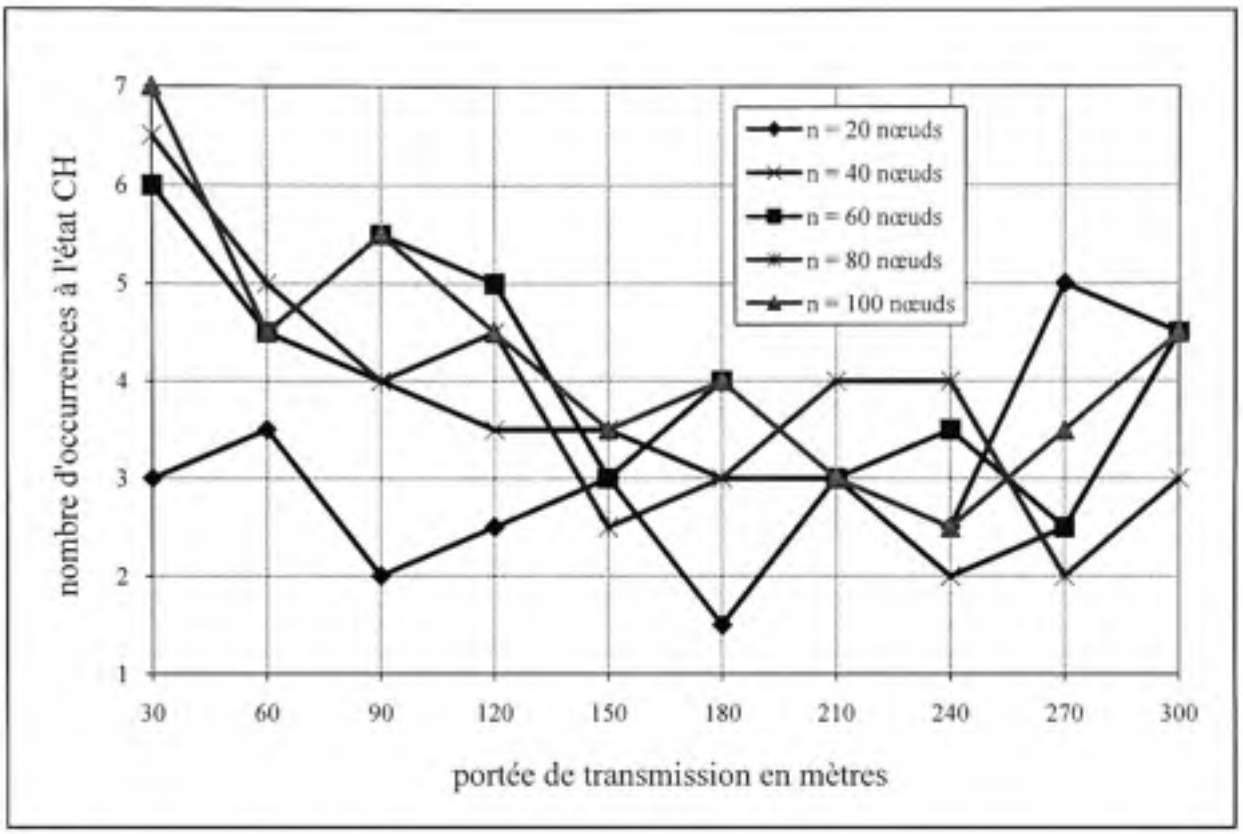

Figure 5.13 ***Nombre d'occurrences à l'état CH versus portée de transmission.***

5.4.3 Passage à l'échelle ou connectivité du réseau

Un modèle de clustérisation est considéré efficace si tous les nœuds du réseau peuvent se communiquer via un ou plusieurs sauts afin de garantir un meilleur acheminement des paquets de bout en bout. Cependant, la mobilité des nœuds et la portée de transmission constituent un obstacle pour le maintien de la connectivité, elle influe directement sur les liens radios et ainsi sur les routes.

Augmenter la portée de transmission permet de garantir une connectivité excellente au détriment d'une consommation élevée des batteries des nœuds, réduisant ainsi leur durée de vie et contribuant en quelque sorte à rendre la topologie moins dense et moins connexe. En revanche, diminuer la portée de transmission permet de réduire les interférences excessives

et les consommations énergétiques, mais provoque la division du réseau en des îlots plus petits et moins connexes.

Pour toutes ces raisons, il est important de trouver un compromis entre ces deux approches sans sacrifier les performances de l'un en faveur de l'autre. La robustesse est garantie par nos procédures en permettant de contourner ces problèmes. Nous remarquons sur la figure 5.14 que la connectivité est meilleure lorsque la topologie est dense. En augmentant la portée de transmission, les CH seront capables d'établir des communications interclusters, améliorant ainsi la connectivité globale du réseau.

Nous pouvons aussi remarquer que la connectivité commence à se stabiliser lorsque le nombre de nœuds augmente dans le réseau. Ceci est aussi une bonne indication sur le niveau de stabilité du modèle et permet un passage à l'échelle des réseaux *Ad hoc*.

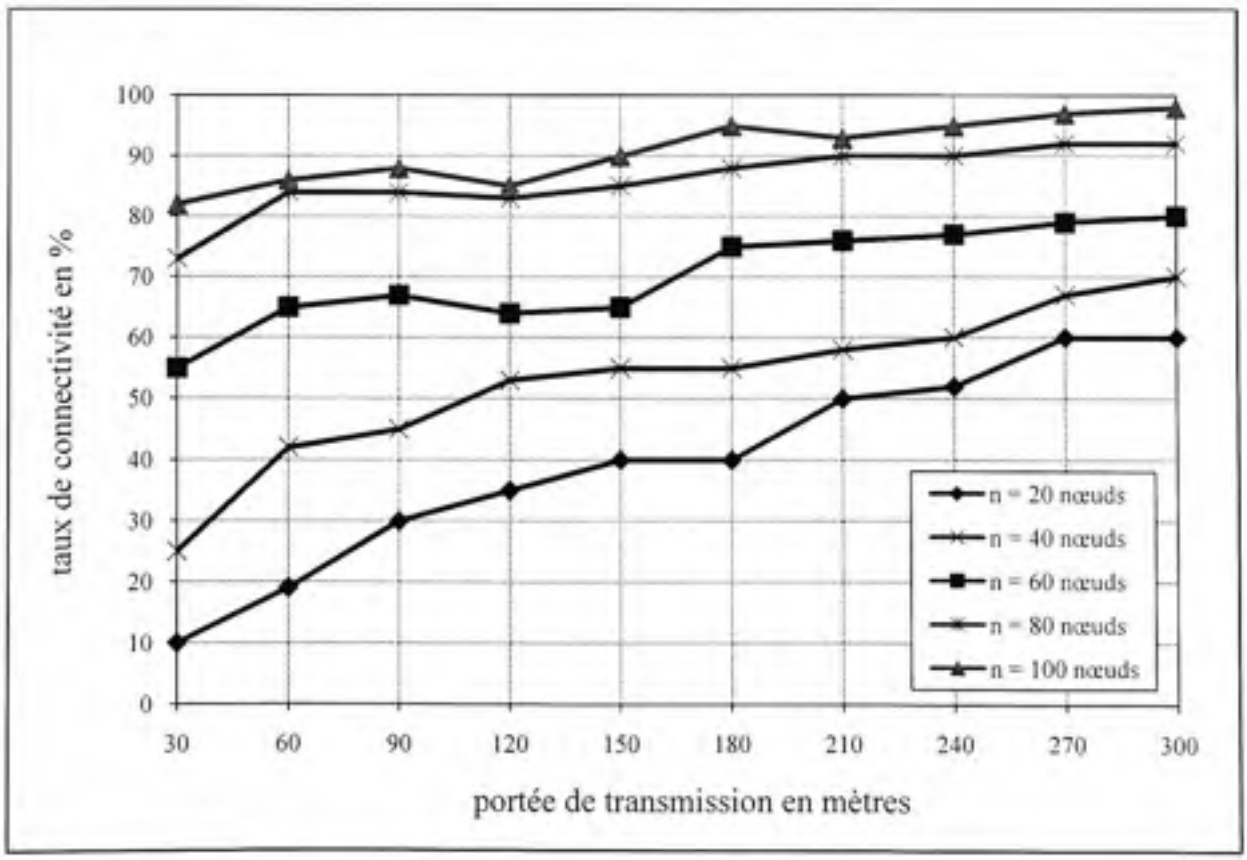

Figure 5.14 ***Taux de connectivité versus portée de transmission.***

5.4.4 Équilibrage de charge entre les clusters

Le modèle de clustérisation proposé essaye de repartir et équilibrer la charge sur tous les clusters afin de garantir un débit, un délai et un taux de perte acceptable par les applications. Nous nous sommes basés sur les résultats des performances intraclusters de la section 5.3 afin de décider du nombre de nœuds optimaux par cluster. Nous avons pris 15 nœuds actifs (ayant du trafic) par cluster comme un paramètre de clustérisation.

Les résultats illustrés sur la figure 5.15 montrent qu'aucun des clusters n'a atteint un taux de 100 % d'occupation. La charge est ainsi équilibrée entre les différents clusters même avec l'augmentation du nombre de nœuds dans le réseau. Ce qui permet d'éviter les congestions sur les CH et garantit également un passage à l'échelle.

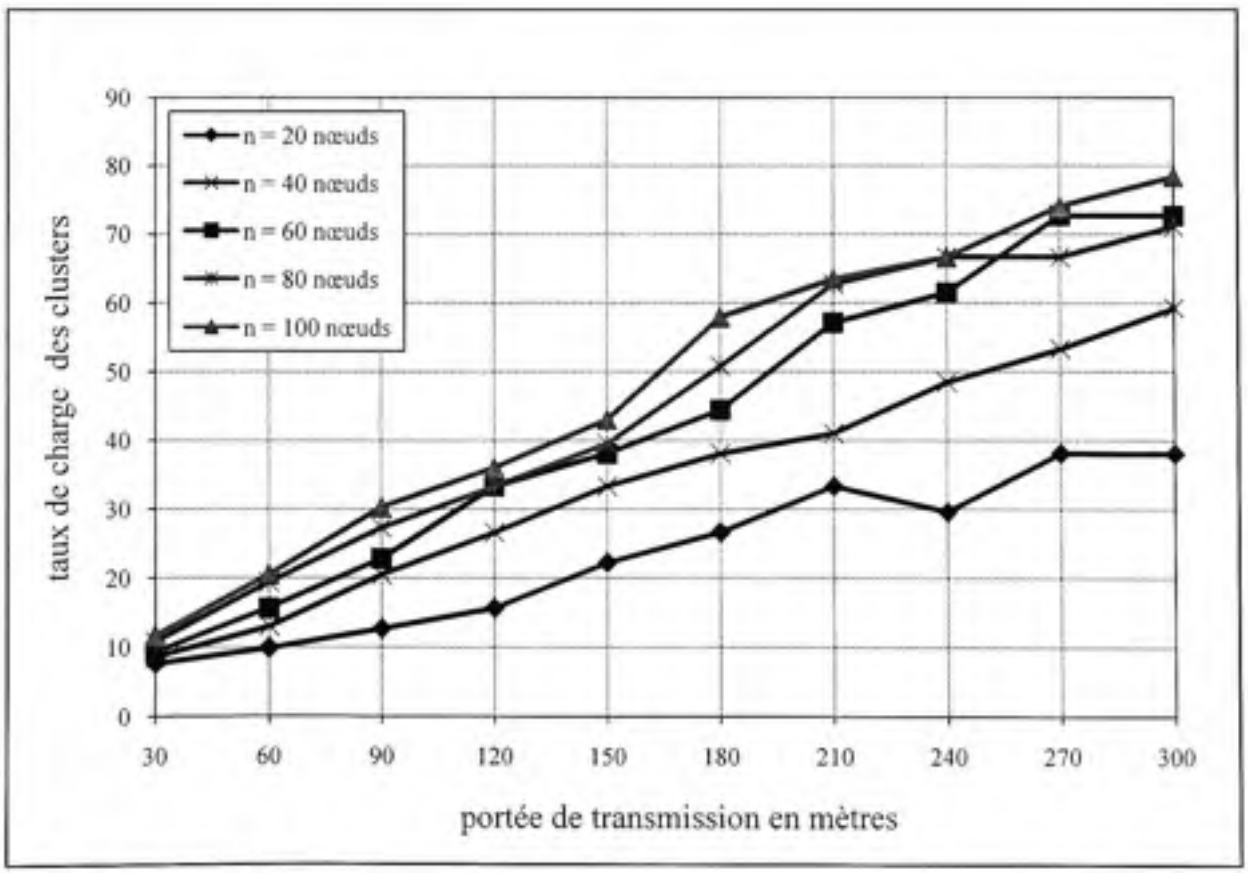

Figure 5.15 ***Taux de charge des clusters versus portée de transmission.***

5.4.5 Étude des *handoff* dans le réseau

Rappelons que les nœuds sont toujours mobiles et se déplacent à une vitesse de 3 Km/h (piéton) à 10 Km/h (promeneur rapide) pour refléter autant que possible le cas des nœuds que nous trouvons dans des environnements tels que des campus universitaires et des salles de conférence.

Sur la figure 5.16, nous pouvons observer le niveau de stabilité de nos clusters en variant le nombre de nœuds dans le réseau. Nous pouvons facilement voir que les nœuds vont faire plus souvent de *handoff* lorsque le réseau est plus dense; ceci est expliqué par le fait que le nombre de clusters formés augmente lorsqu'il y a plus de nœuds dans le réseau. En effet, les nœuds membres tendent à détecter d'autres CH plus stables et plus puissants; ils essayeront de s'attacher à ces derniers.

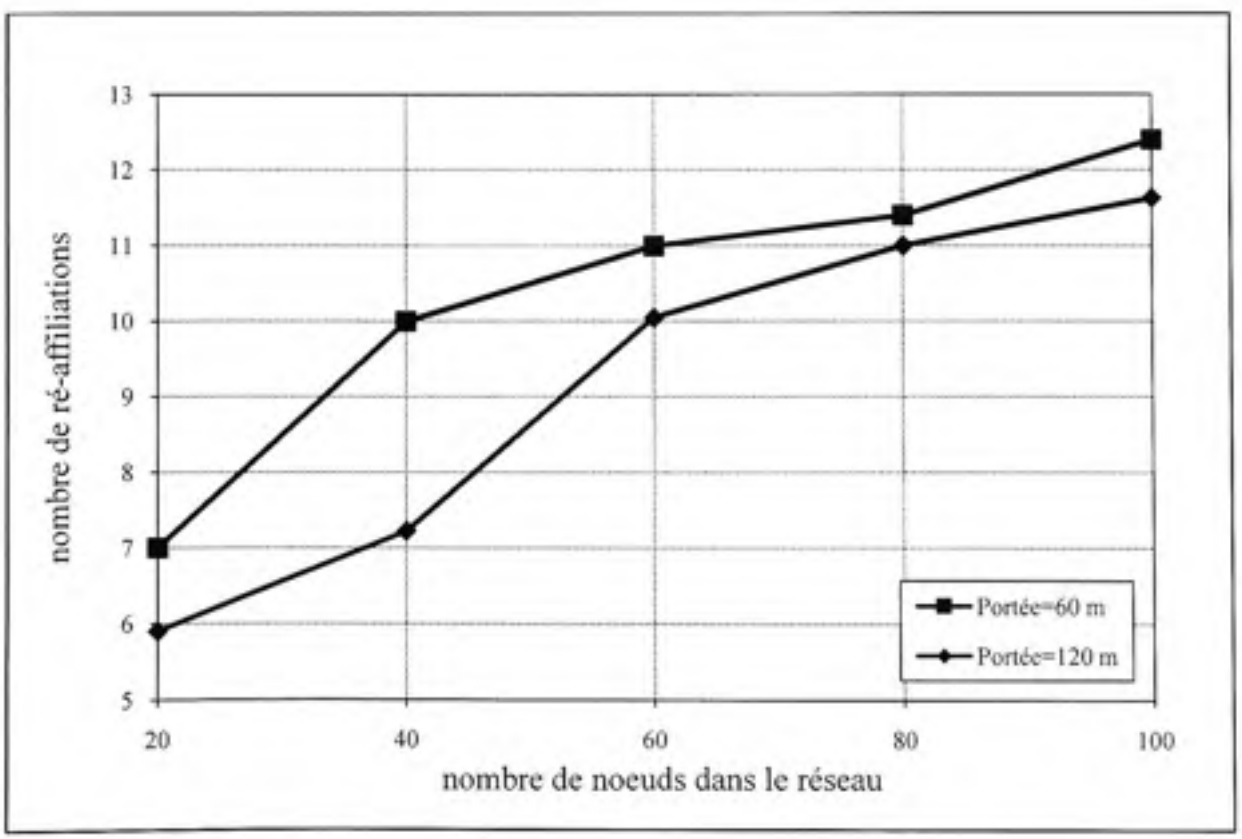

Figure 5.16 ***Nombre de ré-affiliations versus densité du réseau.***

Nous nous intéressons à étudier la robustesse du modèle ainsi que sa flexibilité en termes de *handoff* qui auront lieu dans le réseau. Il est évident que le fait de réduire les *handoff* est une indication sur la stabilité et la durée de vie des CH. Les CH stables et puissants durent plus longtemps dans leur cluster, ceci permet de stabiliser les liens membres-CH à moins que les nœuds soient très mobiles et changent assez souvent de clusters.

La même situation peut être observée sur la figure 5.17 lorsque nous varions la portée de transmission des nœuds. Dans ce cas, le nombre de clusters produits va se stabiliser surtout lorsque le rayon de portée est plus large (voir figure 5.12); ce qui minimise également le nombre d'élections de nouveaux CH. C'est la raison pour laquelle nous remarquons moins de fluctuations dans les courbes de la figure 5.17; les clusters sont plus stables et les nœuds tendent à ne pas changer de cluster à moins qu'ils se déplacent très loin au-delà de la portée de leurs CH.

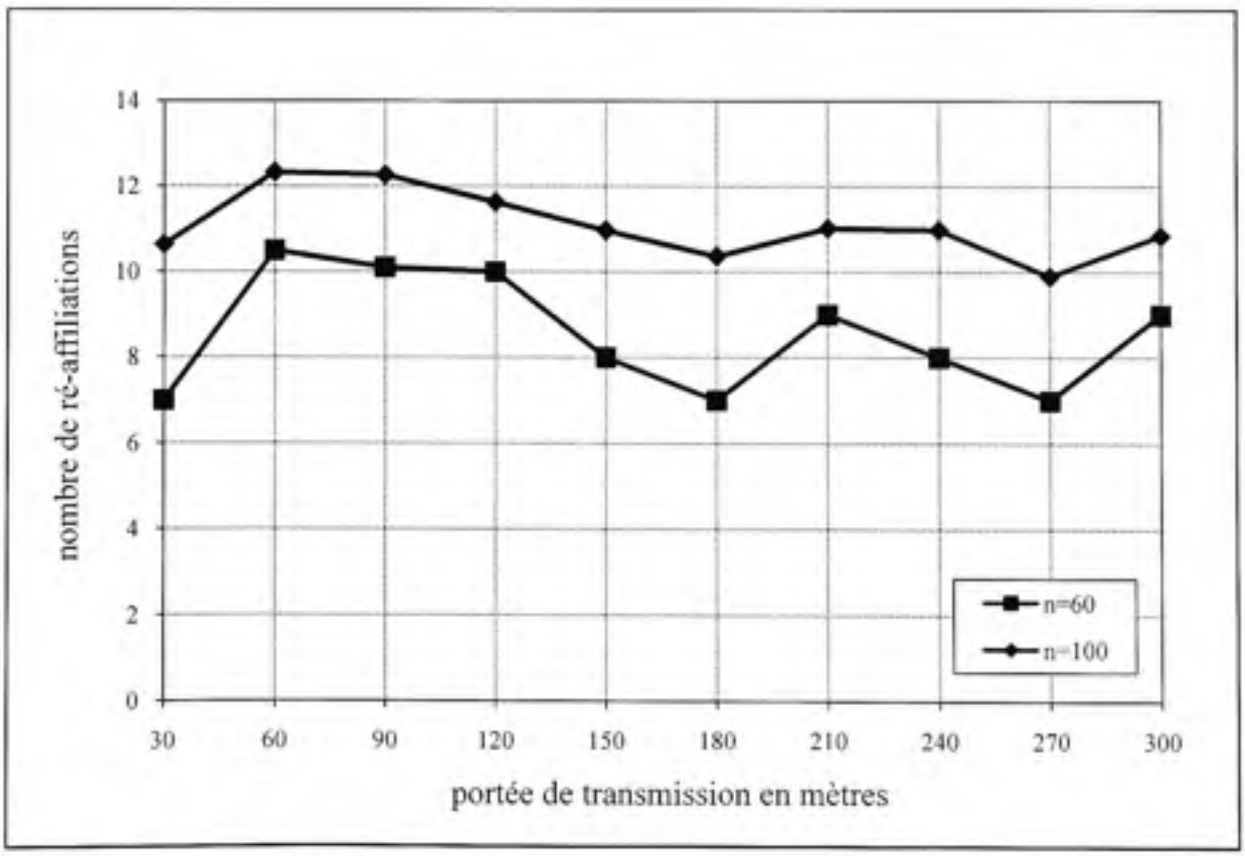

Figure 5.17 ***Nombre de ré-affiliations versus portée de transmission.***

5.4.6 Analyse des paramètres de qualité de service

Il s'agit ici d'étudier le niveau de qualité de service globale de tout le réseau. Rappelons que pour des fins de simplifications, nous avons supposé que les *clusterhead* sont toujours saturés en ayant des paquets à servir pour les membres et les clusters avoisinants. Nous nous sommes basés sur l'étude des performances intraclusters présentée dans la section 5.3.5 pour choisir le nombre optimal des nœuds qui garantit une qualité de service acceptable dans les clusters.

Comme nous l'avons remarqué, les paramètres du modèle sont ajustables et dépendent des applications utilisées dans le réseau. Une connaissance approfondie de l'environnement ainsi que des applications utilisatrices nous conduit à mieux paramétrer les métriques du poids combiné ainsi que le nombre optimal de nœuds par cluster. Toutefois, nous considérons dans cette section des clusters ayant 15 nœuds actifs et des paquets ayant une taille de l'ordre de 1023 octets afin de mesurer les paramètres de la qualité de service.

Nous remarquons sur la figure 5.18, l'effet de l'augmentation de la portée sur le débit interclusters. Ce débit tend à diminuer lorsque la portée augmente; un phénomène très fréquent dans les communications sans fil, causé par les différentes perturbations et les objets interférents. Cependant, nous observons que ce débit reste élevé par rapport à la bande passante maximale du canal (par hypothèse 1 Mbit/s). Ceci permet une mise à l'échelle lorsque le réseau devient plus dense.

Sur les figures 5.19 et 5.20, nous illustrons respectivement le délai interclusters et le délai de bout en bout. Quand la portée de transmission est petite, les clusters sont dispersés et moins connectés. Le délai d'attente d'établissement des liens interclusters est ainsi plus long. Ce délai commence à diminuer en augmentant la portée de transmission. Dans ce cas, la connectivité est plus élevée et le délai interclusters est ainsi réduit.

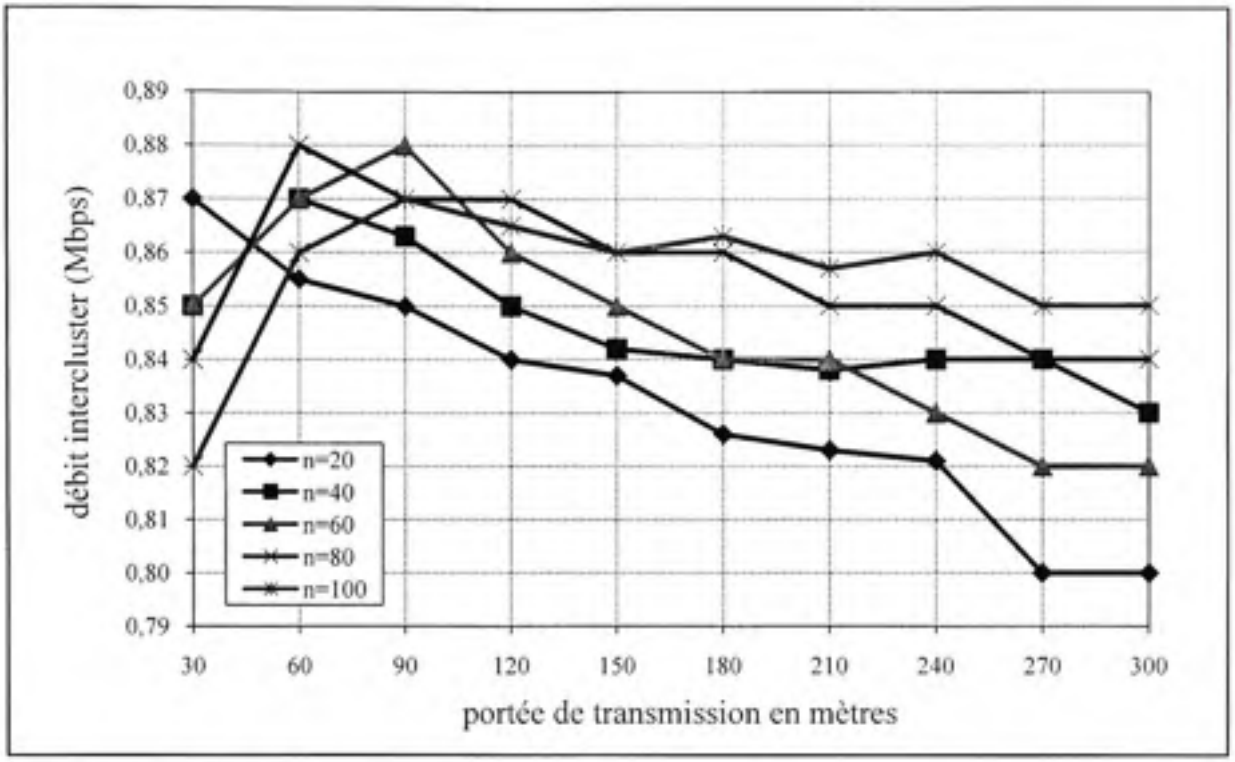

Figure 5.18 ***Débit moyen des liens interclusters versus portée de transmission.***

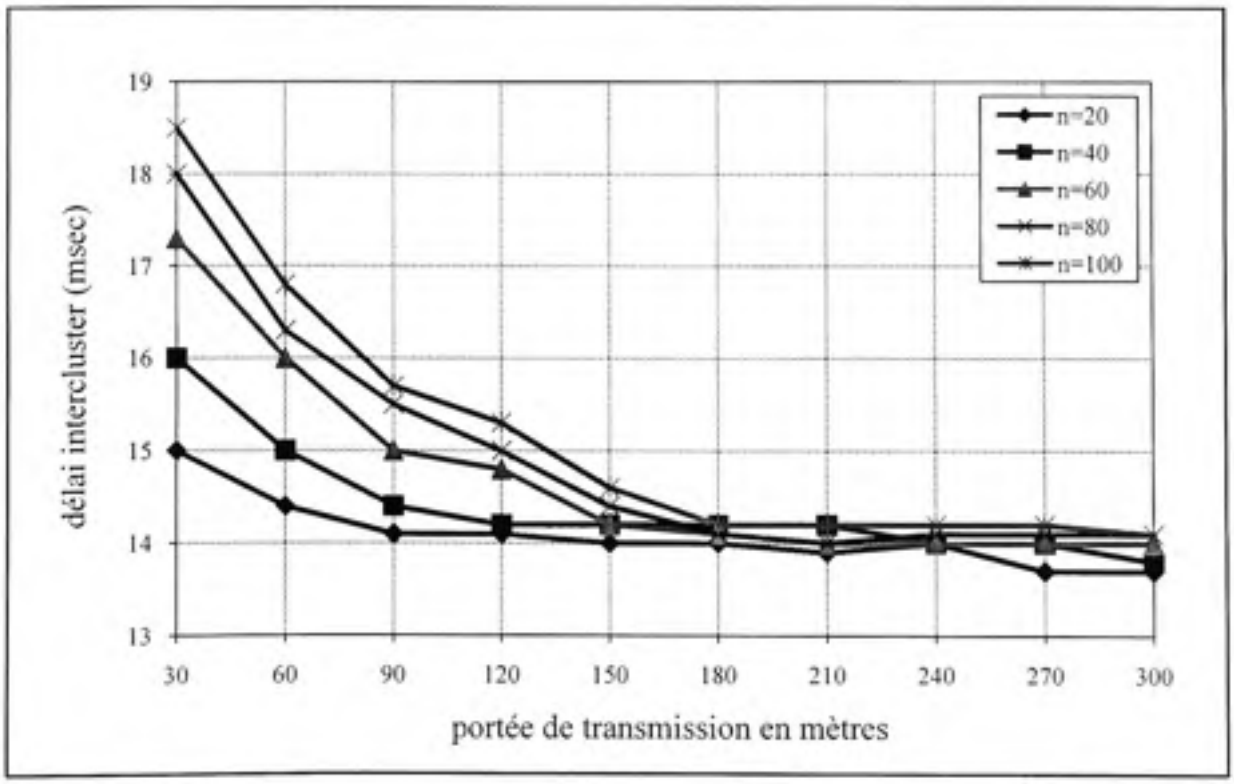

Figure 5.19 ***Délai moyen des liens interclusters versus portée de transmission.***

Le même raisonnement s'applique également sur la figure 5.20. Dans ce cas, le délai de bout en bout tient compte des délais intraclusters et du nombre moyen de sauts qu'un paquet va prendre dans le réseau.

Nous pouvons aussi remarquer qu'à partir d'une portée de 150 mètres et plus, le délai de bout en bout commence à se stabiliser (de l'ordre de 150 msec). Ceci est normal étant donné que le nombre de sauts (ou nombre de clusters formés) tend à devenir constant et que la connectivité s'approche d'une valeur de 100 %. Notons que des valeurs de délai de bout en bout de l'ordre de 50 à 1000 msec sont très acceptables pour les paquets de données ayant une taille de 1023 octets; ce délai diminue considérablement lorsque nous tenons compte des paquets temps réel de taille de centaines de bits, étant donné que le délai intraclusters diminue rapidement lorsque la taille des paquets diminue (voir figure 5.7).

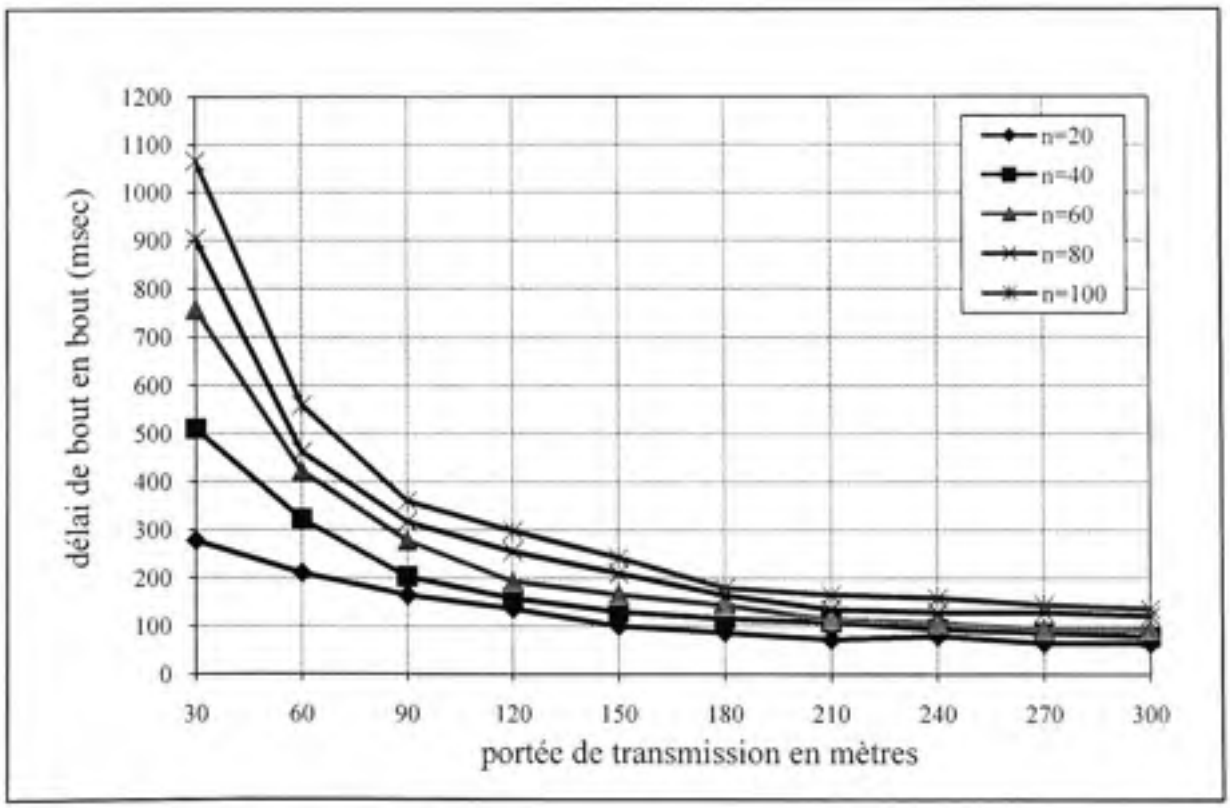

Figure 5.20 ***Délai de bout en bout versus portée de transmission.***

5.5 Comparaison avec d'autres modèles de clustérisation

Dans cette section, nous présentons une étude comparative des performances de notre modèle ECA (*Efficient Clustering Algorithm*) et celle d'un autre modèle existant basée sur l'assignation des poids aux nœuds et permettant de construire des *1-clusters*. Nous avons choisi le modèle WCA *(Weighted Clustering Algorithm)* de [Chatterjee *et al.* (2002)]; ce modèle considéré dans la littérature comme une des meilleures références, il a prouvé ses performances sur d'autres algorithmes basés sur des métriques comme *Lowest-id* [Lin et Gerla (1997)], MOBIC [Basu *et al.* (2001)].

Le poids dans WCA tient compte d'une variété de paramètres dont le degré, la somme de distance entre le CH et ses voisins, la vitesse des nœuds ainsi que le temps passé par un nœud à l'état CH. Pour faire notre étude, nous avons implémenté les deux approches dans le même environnement en utilisant le même scénario généré par le simulateur SMGen de [Agba *et al.* (2006a)].

5.5.1 Comparaison du nombre moyen de clusters

Figure 5.21 illustre le nombre moyen de clusters produits en fonction du nombre de nœuds dans le réseau tout en variant la portée de transmission. Nous constatons que WCA génère moins de clusters que notre modèle; par contre, la différence est trés minime. Cependant, nous nous attendions à ce résultat, car WCA n'implémente aucun mécanisme d'équilibrage de charge et ne restreint pas le nombre de nœuds par cluster du fait qu'il ne tient pas compte de la qualité de service. Notre modèle utilise un mécanisme de contrôle d'admission permettant d'équilibrer la charge entre plusieurs clusters en limitant le nombre de nœuds actifs par cluster.

Toutefois, nous pensons que le nombre de clusters formés ne nous fournit pas un outil pour la mesure des performances des clusters et qu'il est primordial d'autres métriques afin de juger la stabilité de ces structures (voir les sections suivantes).

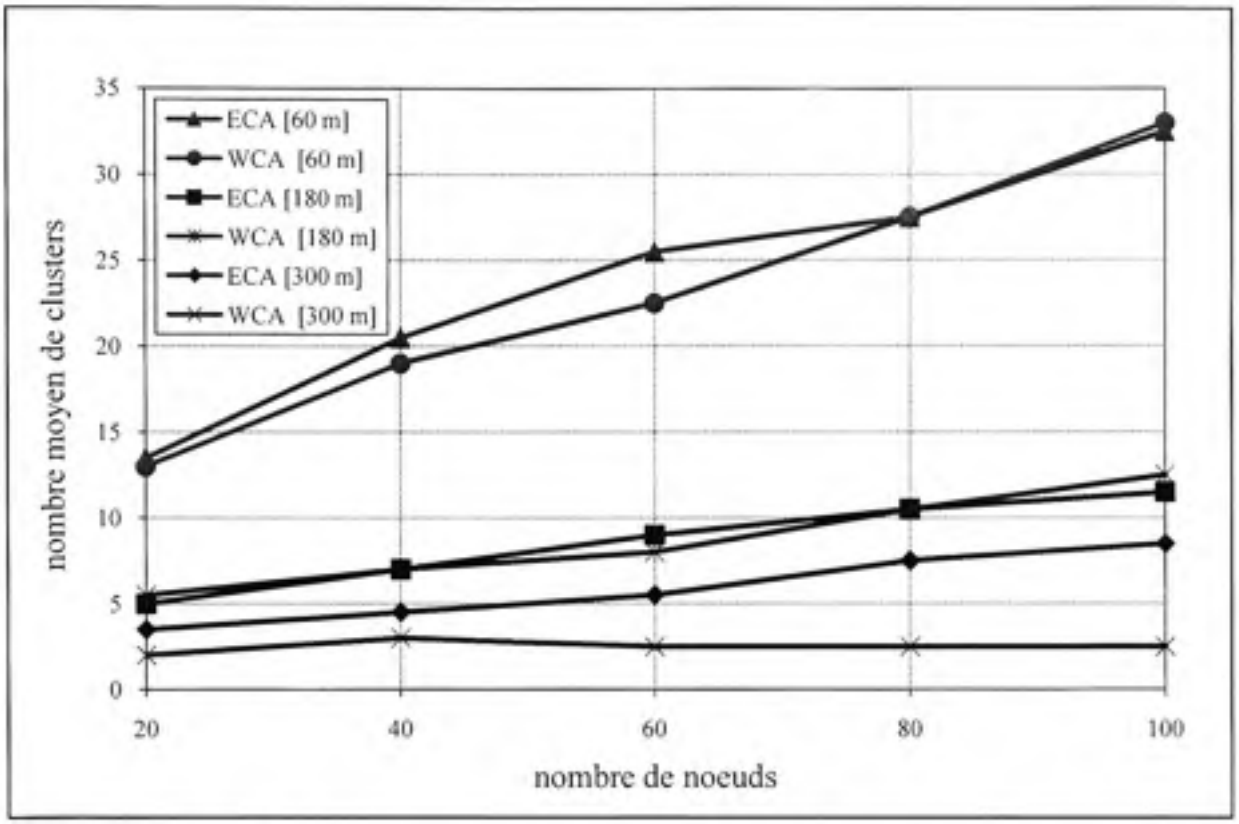

Figure 5.21 ***ECA versus WCA en termes de nombre moyen de clusters.***

5.5.2 Comparaison du nombre moyen d'occurrences à l'état CH

Dans WCA, les CH continuent à changer même lors des moindres changements de topologie ou des conditions des nœuds. Le CH renonce à son rôle dès qu'un nouveau nœud ayant un meilleur poids arrive dans le cluster ou parce qu'il épuise rapidement sa batterie étant donné qu'il dessert sans restriction un nombre élevé de nœuds. Ceci se traduit par un nombre élevé de transitions sur les CH comme illustré sur la figure 5.22 comparativement à ECA.

En effet, dans notre modèle, le CH est plus stable et dure plus longtemps dans le cluster. Nous remarquons qu'avec 100 nœuds dans le réseau et pour une portée de 180 mètres, notre modèle produit de 50 à 83.3 % moins de transitions sur chaque CH qu'avec WCA. Ainsi, la stabilité est fortement priorisée dans notre modèle.

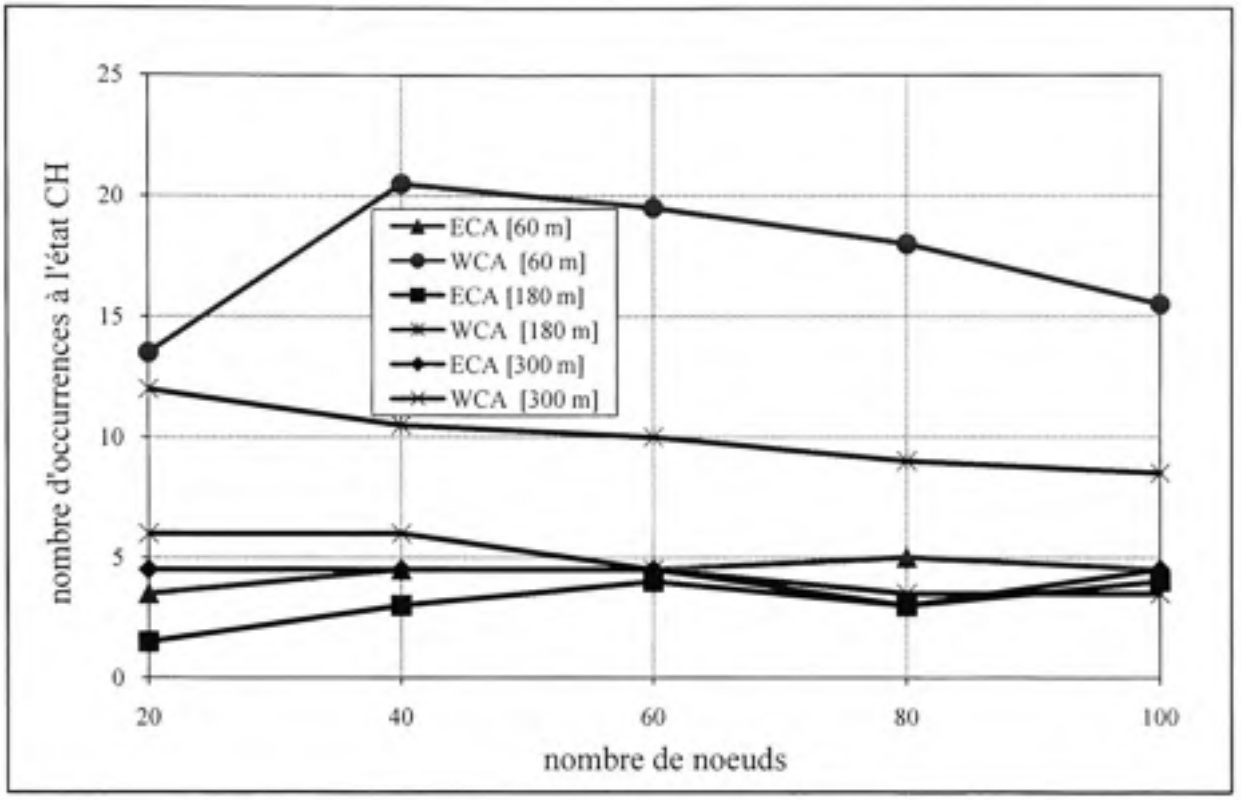

Figure 5.22 ***ECA versus WCA en termes de nombre d'occurrences à l'état CH.***

5.5.3 Comparaison du nombre de *handoff*

Nous remarquons sur la figure 5.23 que lorsque la portée est de l'ordre de 120 mètres, le nombre de ré-affiliations augmente dans les deux modèles en augmentant le nombre de nœuds dans le réseau. Toutefois, le taux d'augmentation tend à ralentir et à se stabiliser dans notre modèle, ce qui n'est pas le cas dans WCA.

En effet, lorsque nous avons 20 nœuds dans le réseau, ayant une portée de 120 mètres et se déplaçant à une vitesse de 3 à 10 Km/h, notre modèle produit 61.5 % moins de ré-affiliations que WCA. Ce taux augmente à 66.5 % lorsque le nombre de nœuds augmente dans le réseau.

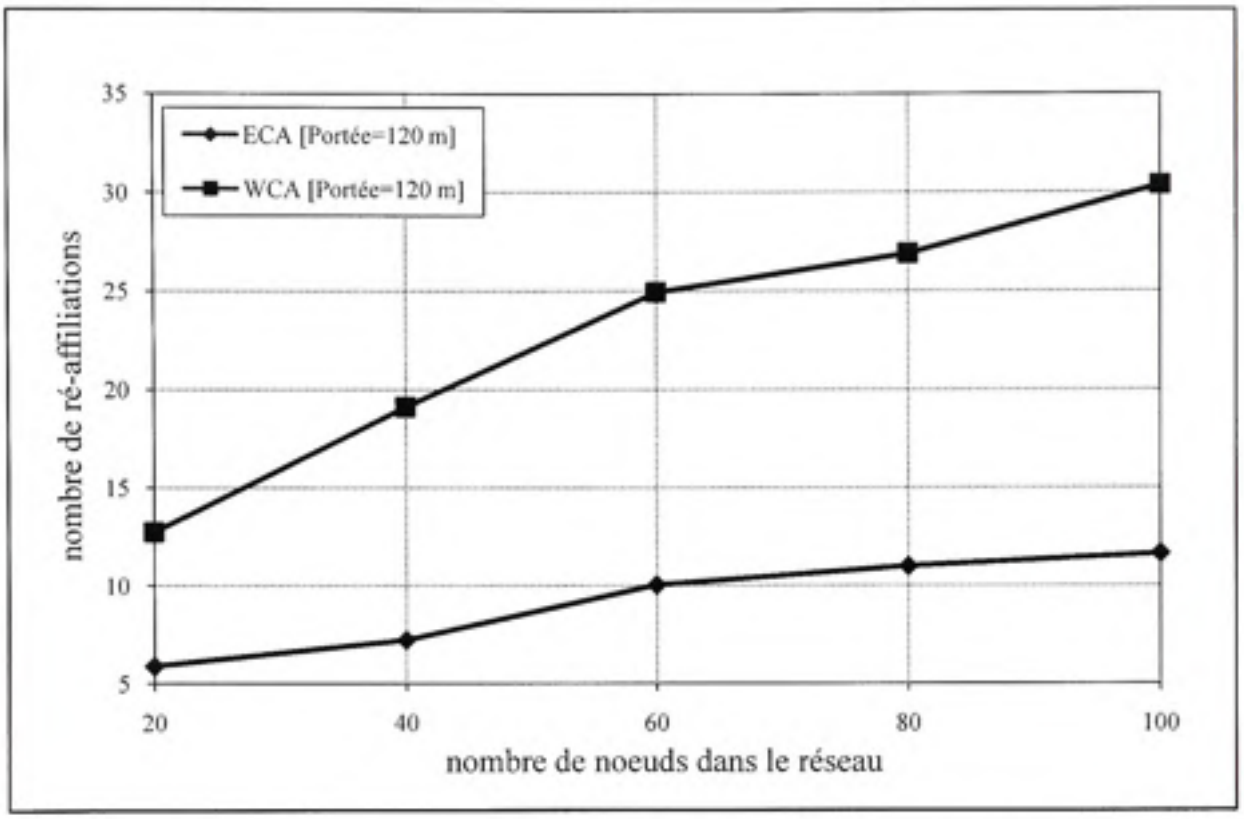

Figure 5.23 ***ECA versus WCA en termes de ré-affiliations pour multiples densités.***

D'autre part, le même phénomène peut être observé sur la figure 5.24 lorsque nous varions la portée de transmission des nœuds. Notre modèle produit 37.5 % et 66.5 % moins de ré-affiliations que WCA pour des portées de 30 et de 300 mètres respectivement.

Nous observons également beaucoup moins de fluctuations dans le cas de notre modèle; ceci se traduit par une meilleure stabilité que WCA où le nombre de fluctuations (*handoff*) ne cessent pas de varier rapidement.

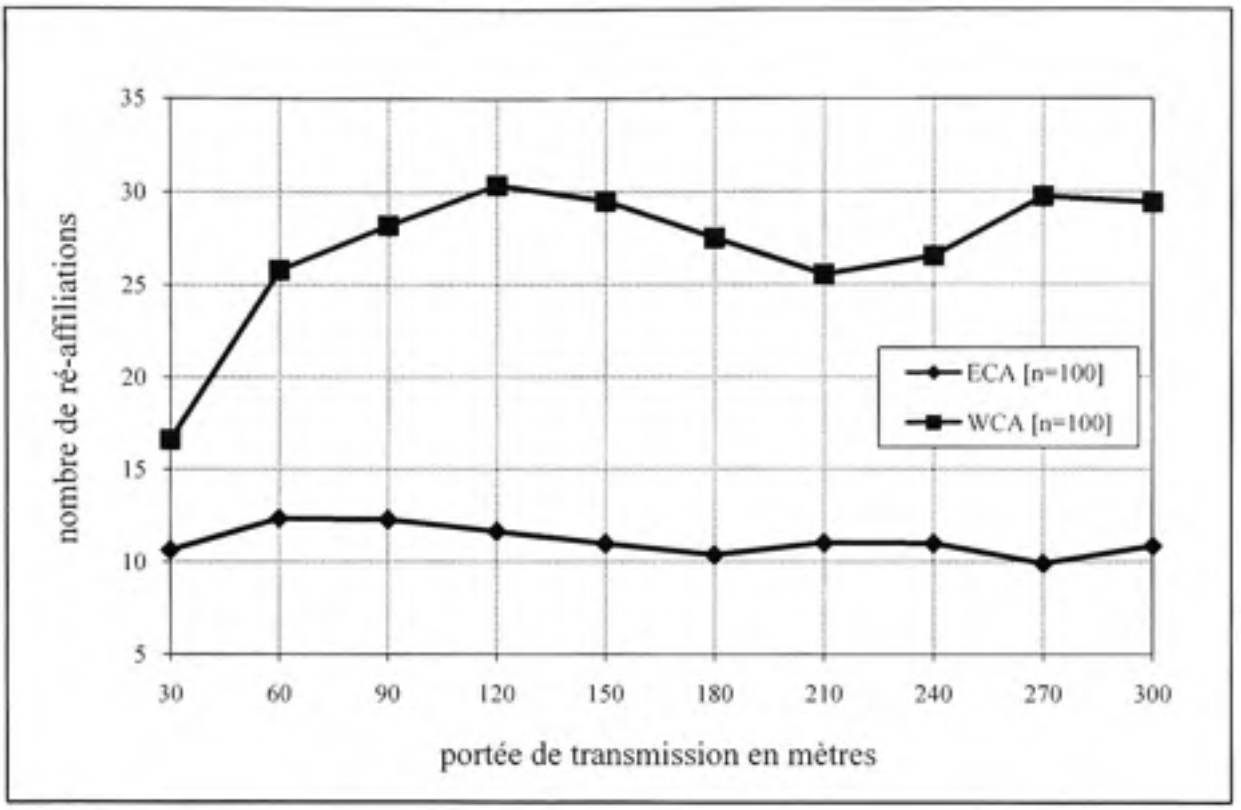

Figure 5.24 ***ECA versus WCA en termes de ré-affiliations pour multiples portées.***

5.5.4 Comparaison de la connectivité (mise à l'échelle)

Pour comparer le facteur de mise à l'échelle, nous étudions le taux de connectivité des nœuds lorsque nous augmentons le nombre de nœuds dans le réseau. Nous remarquons sur la figure 5.25 que la probabilité qu'un nœud puisse établir une route avec tout autre nœud dans le réseau via un lien direct ou un lien multisauts, soit plus élevée en ECA qu'en WCA.

Nous observons également plus de fluctuations dans WCA qu'en ECA; ce dernier tend à se stabiliser plus rapidement. Ceci s'explique par le nombre plus élevé de transitions sur les nœuds CH, occasionnant des coupures plus fréquentes des routes et réduisant le niveau de stabilité globale du réseau.

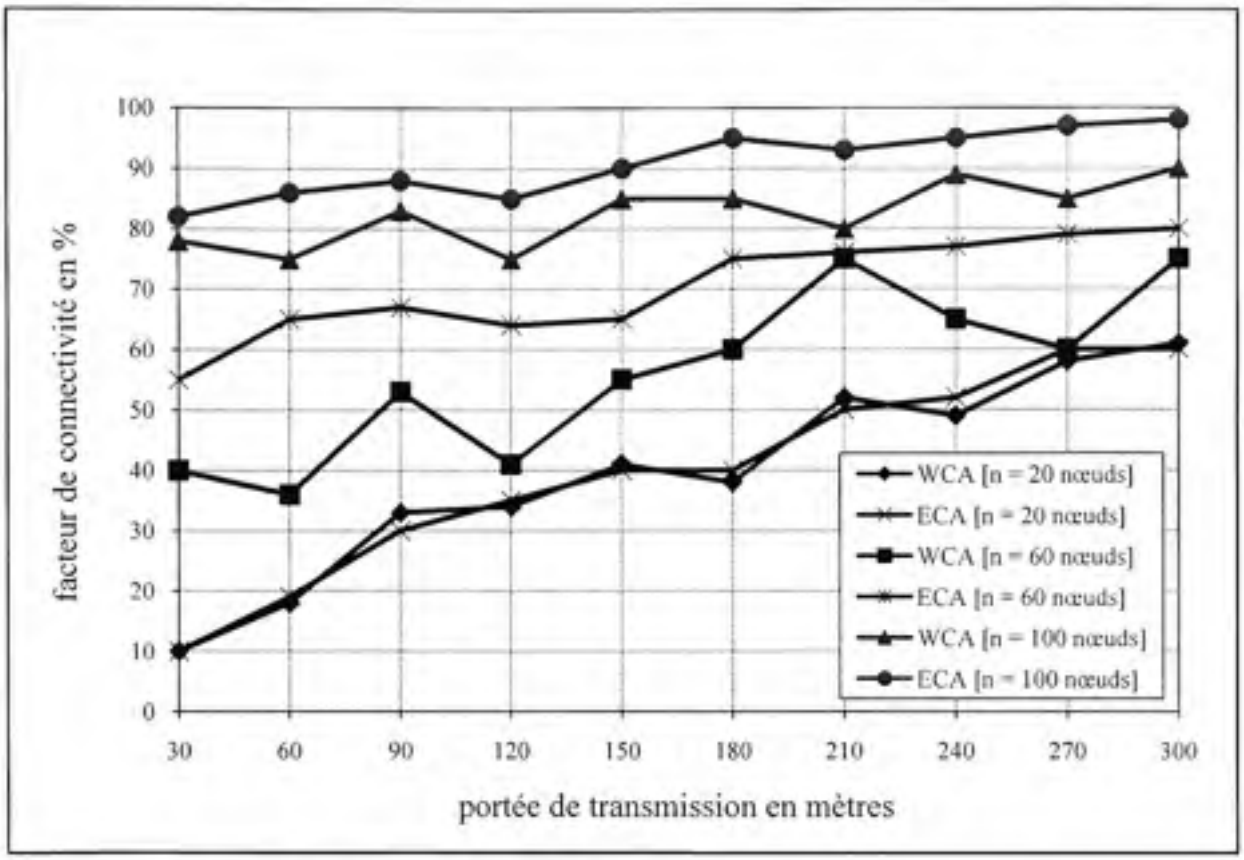

Figure 5.25 ***ECA versus WCA en termes de taux de connectivité.***

5.5.5 Comparaison des paramètres de qualité de service

Il est à noter que WCA n'implémente aucun mécanisme de contrôle d'admission qui permet de garantir un service de qualité dans les clusters. Les auteurs utilisent une métrique nommée « différence de degré » qui permet d'assigner un poids à un nombre prédéfini de nœuds qu'un clusterhead devrait servir; par contre, cette métrique ne fournit aucun moyen de limiter le nombre de nœuds par cluster. Dans l'objectif de comparer WCA avec notre modèle, nous avons paramétré les clusters de WCA de façon que chaque d'eux ne devrait pas supporter plus qu'un nombre prédéfini de 15 nœuds.

Comme illustré sur les figures 5.26, 5.27 et 5.28, nous remarquons que notre modèle fournit de meilleures performances en termes de débit, de délai et de nombre de retransmissions par paquet. Nous nous attendions à ce résultat du fait que tous les nœuds du réseau dans WCA

utilisent le même code CDMA; les communications intraclusters sont certainement affectées par les clusters avoisinants et le débit de chaque nœud va dépendre du nombre de voisins qui sont dans le même rayon de portée de transmission.

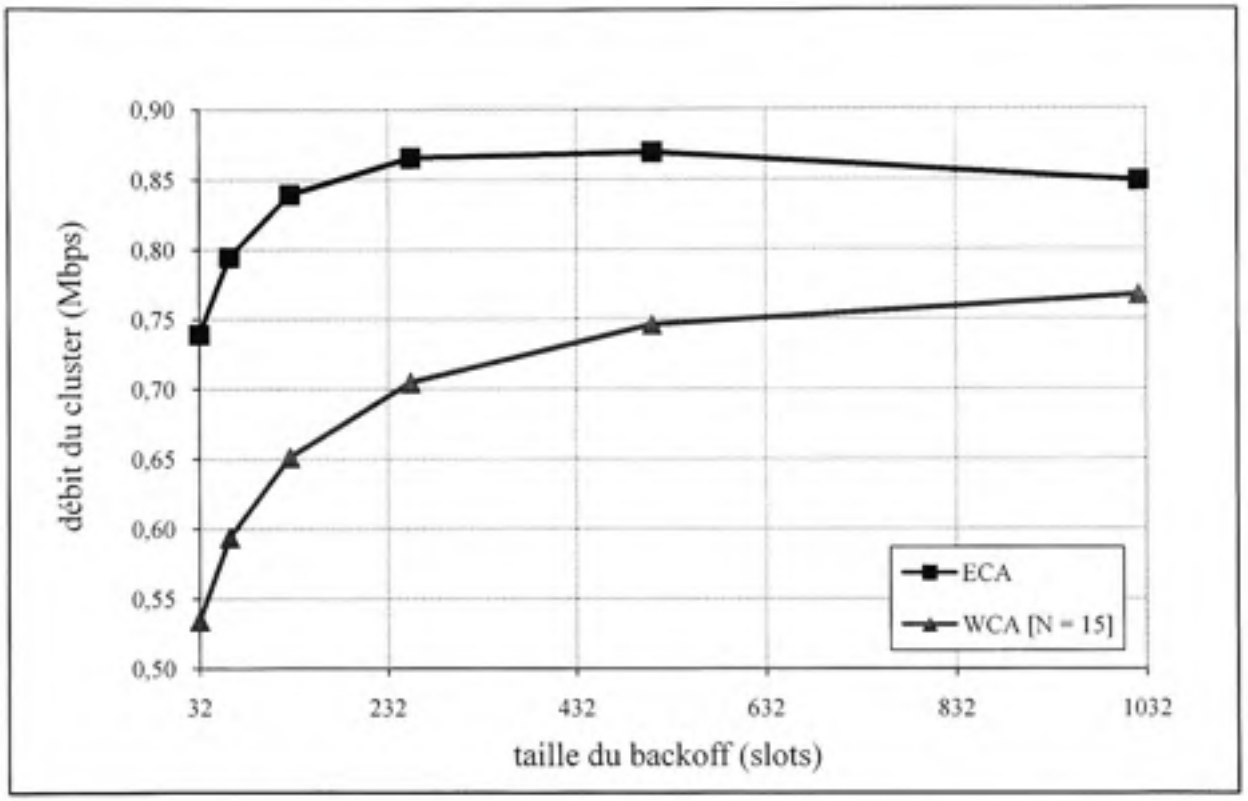

Figure 5.26 ***ECA versus WCA en termes de débit des clusters.***

De ces figures, nous pouvons facilement observer l'effet de l'utilisation de plusieurs codes CDMA sur les paramètres de la qualité de service. En effet, nous pensons que la clustérisation doit non seulement fournir des moyens pour créer une dorsale virtuelle servant à faciliter l'implémentation de routage, mais également elle doit prendre en considération les besoins des applications en termes de QoS tout en garantissant un niveau de stabilité élevé et une mise efficace à l'échelle.

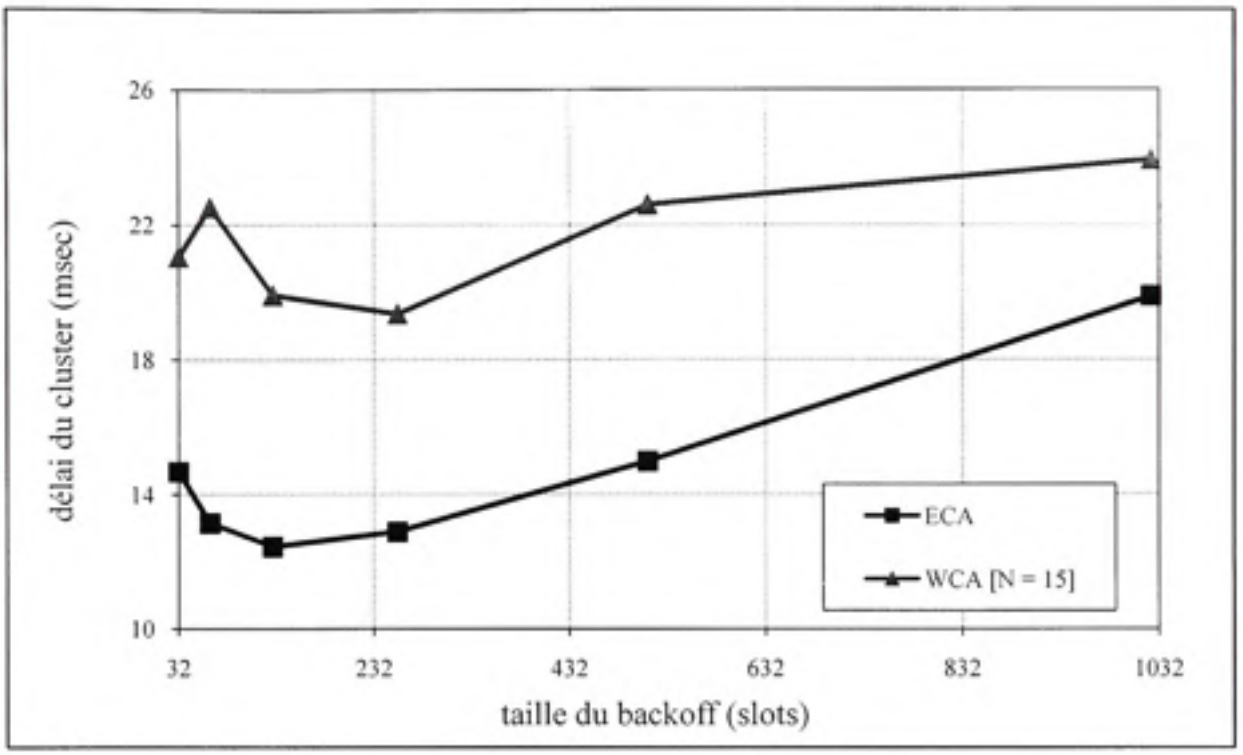

Figure 5.27 *ECA versus WCA en termes de délai des clusters.*

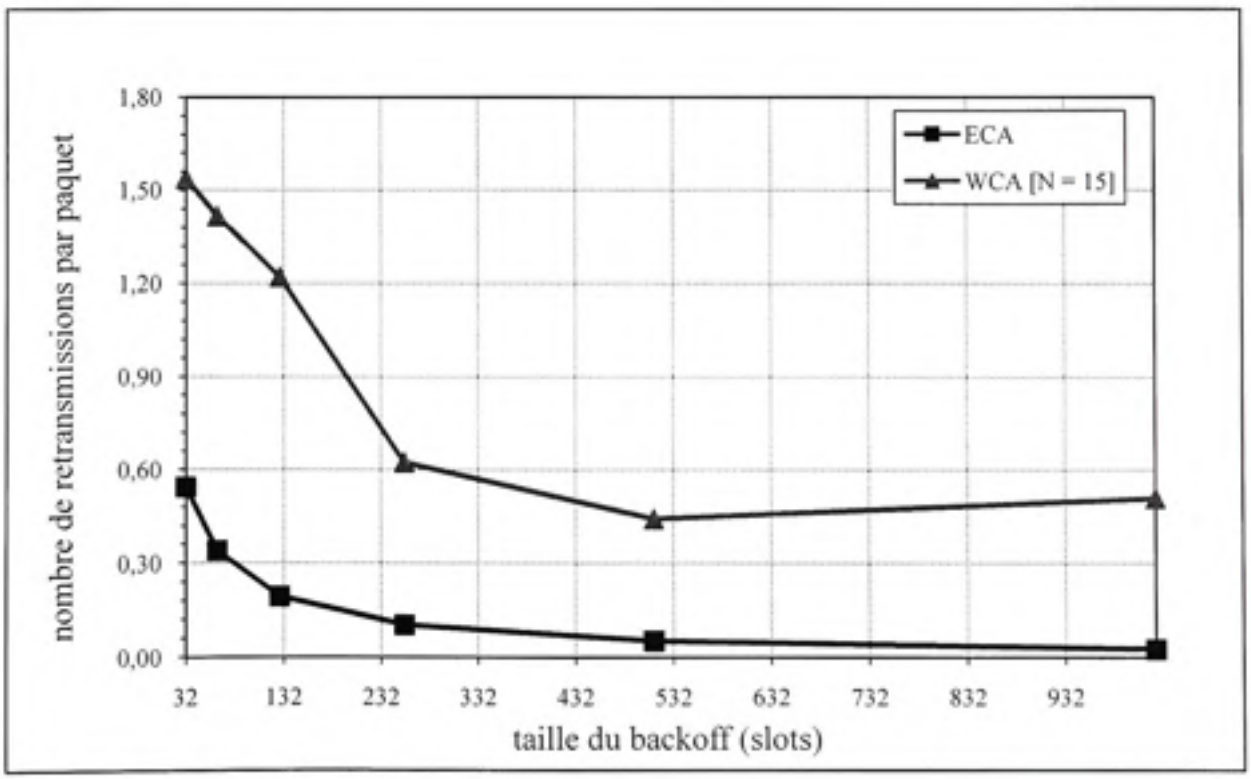

Figure 5.28 *ECA versus WCA en termes de nombre de retransmission par paquet.*

5.6 Conclusion

Dans ce chapitre, nous avons commencé par la présentation des outils et des paramètres de simulation qui ont servi à la validation du modèle et des approches proposés dans cette recherche. L'outil de simulation que nous avons utilisé permet de générer des scénarios mobiles très réalistes tenant compte de différents facteurs de la couche physique sans fil et intégrant une multitude de modèles de mobilité répondant aux besoins spécifiques de l'environnement.

Notre choix des paramètres et du modèle de mobilité, décrit auparavant, s'applique au champ d'application des réseaux *Ad hoc* que nous visons. En effet, nous nous sommes intéressés à étudier des réseaux ayant une taille limitée et reflétant les environnements de salles de conférence et de campus universitaires. Cependant, rien n'empêche d'essayer d'autres scénarios en choisissant un paramétrage adéquat des métriques de clustérisation et des mécanismes d'optimisation de la qualité de service par cluster.

Dans ce contexte, les résultats de simulation démontrent que notre modèle est utilisable dans des réseaux à grande échelle, aussi bien à mobilité réduite qu'à mobilité forte. Les objectifs ont été validés par des simulations réalistes. Ils s'articulent sur les points suivants :

- **mesure de la stabilité et de la persistance de la topologie :** nous avons remarqué que les clusters produits sont stables dans le temps. Une hiérarchie formée d'un ensemble de nœuds CH peut être utilisée ultérieurement pour des fins d'adressage, d'économie d'énergie, de routage, d'allocation de bande passante et d'intégration de toute autre fonctionnalité des réseaux filaires. Le modèle peut par exemple être exploité par les protocoles de routage; une dorsale virtuelle servira à faciliter l'acheminement des paquets, à minimiser les inondations ainsi qu'à réduire le nombre de sauts en minimisant le nombre de clusters formés.

De ce fait, nous pouvons imaginer des nœuds désignés qui s'échangent exclusivement les informations de mise à jour et de routage. De même, il se peut qu'un changement de topologie crée des changements au niveau des routes; les nœuds de cette dorsale doivent dans la mesure du possible rester inchangés et jouer leur rôle durant des longues périodes. Ceci a été également prouvé par la minimisation du nombre d'occurrences à l'état CH;

- **optimisation dynamique du choix des *clusterhead* et robustesse à la dynamique du réseau :** puisque chaque nœud est libre de se déplacer aléatoirement et indépendamment des autres, la topologie ne cesse jamais de changer. En conséquence, nous avons observé le nombre de *handoff* occasionnés par les nœuds dans le réseau; nous avons remarqué que le modèle s'adapte aux changements. Les nœuds tendent à ne pas changer de cluster assez fréquemment même en présence d'une forte mobilité. Ceci s'explique par le fait que la charge est toujours équilibrée entre les clusters de sorte que les chances de quitter son cluster soient très minimes à moins que le nœud se déplace très loin de sa zone.

  Les fluctuations dans les graphiques sont normales, car il n'y a aucune solution parfaite dans les réseaux *Ad hoc* vu que la nature et les conditions des nœuds sont incontrôlables. Toutefois, nous avons remarqué que le rendement du modèle ne se dégrade que pendant un laps de temps nécessaire à la mise à jour des données du modèle. Il est normal que plus l'intervalle des mises à jour est faible, plus le modèle réagit rapidement aux changements, et meilleure est sa robustesse. Pour éviter de surcharger les nœuds par des traitements aussi fréquents, nous avons jugé qu'un intervalle de cinq secondes serait raisonnable dans le cas des réseaux que nous traitons. Dans d'autres cas, il est primordial de mieux examiner ce paramètre tout en tenant compte de l'environnement et des capacités des nœuds;

- **flexibilité d'adaptation du modèle à l'environnement :** les nœuds dans un réseau *Ad hoc* se meuvent dans un environnement aléatoire. Par conséquent, la densité du réseau et la mobilité des nœuds peuvent évoluer avec le temps. Le modèle s'adapte bien et peut

être aisément paramétrable afin de suivre les conditions environnementales. Les résultats démontrés ont été récoltés dans un environnement pré-paramétré reflétant le cas des réseaux ayant une taille limitée (campus universitaire). Cependant, il est idéalement souhaitable que l'adaptation se fasse d'une manière automatique et dynamique;

- **localisation des décisions :** le modèle proposé permet à tout nœud de n'échanger du trafic de contrôle qu'avec d'autres nœuds à une distance limitée de lui (1 saut). L'information sur le réseau étant partielle, moins d'*overhead* est requis, remplissant la contrainte en termes de réduction des interférences sur les canaux radio et de traitement à faire par les nœuds. En outre, les données nécessaires à une décision de routage étant locales au niveau des CH, la réactivité du routage est grandement accélérée;

- **mise à l'échelle :** comme nous l'avons déjà vu dans le chapitre 3, aucun des protocoles de routage actuels ne passe totalement à l'échelle. Nos simulations ont clairement démontré que les performances ne baissent pas radicalement lorsque le nombre de nœuds constituant le réseau augmente : la connectivité s'améliore grandement lorsque le réseau est plus dense. De plus, moins de fluctuations sont observées dans les graphiques; ce qui montre une forte indication sur le niveau de stabilité des clusters et par conséquent sur la persistance des routes à travers le réseau;

- **garantie de la qualité de service :** à notre connaissance, aucun des modèles de clustérisation existants n'a pris en considération les paramètres de QoS des applications. L'introduction d'une nouvelle approche de distribution de codes CDMA nous a permis de tirer profit de la capacité des systèmes CDMA. Le trafic de contrôle étant transmis sur le canal avec un code de signalisation commun, ceci permet de minimiser son impact sur le trafic des données utiles et également de soulager les CH. De même, l'utilisation des codes *intracluster_code* et *intercluster_code* permet de réduire les interférences entre ces structures.

Dans les scénarios considérés pour la simulation, nous avons voulu garantir un certain niveau de QoS aux nœuds du cluster. Idéalement, une connaissance approfondie des applications utilisatrices et du champ d'application du réseau *Ad hoc* permettra d'ajuster les paramètres de clustérisation et du mécanisme de contrôle d'admission pour maximiser les performances à nos besoins;

- **comparaison avec d'autres modèles :** nous avons terminé par une étude comparative des performances de notre modèle par rapport à celle d'un modèle de référence existant dans la littérature. Dans les mêmes conditions topologiques et environnementales, les résultats ont montré que notre modèle fournit de meilleurs résultats, tout particulièrement en termes de qualité de service, de stabilité et de passage à l'échelle.

CONCLUSION

Le perfectionnement des dispositifs de transmission sans fil a connu une croissance accrue aux cours des dernières années, donnant naissance à de nombreuses applications sans fil prometteuses. Un réseau *Ad hoc* est un exemple de ces applications. C'est un réseau facile à déployer sans l'aide d'une infrastructure préexistante ou d'une entité centralisée. Ayant des caractéristiques dynamiques et trop changeantes, ces réseaux ont attiré l'attention des chercheurs et nombreuses sont les approches qui ont été proposées pour répondre à des problématiques particulières.

Nous avons commencé par une étude approfondie des réseaux *Ad hoc* afin de mieux décortiquer ses problèmes. À part la dynamique et la mobilité de ces réseaux, nous avons conclu que le manque d'une hiérarchie est la cause principale qui empêche d'une part le passage à l'échelle, et d'autre part, l'implémentation délicate des fonctionnalités courantes des réseaux filaires. C'est dans ce cadre que se focalise cette dissertation.

En effet, les réseaux *Ad hoc* ont souvent été considérés à plat, dans le sens où tous les nœuds contribuent d'une façon égalitaire aux différentes tâches. L'acheminent ou le routage des paquets se fait soit proactivement où les nœuds doivent toujours annoncer leur présence dans le réseau, soit à la demande où la recherche d'une route vers une destination est déclenchée au moment où le nœud source a du trafic à transmettre. Dans le deux cas, le trafic de contrôle est inondé dans tout réseau, consommant autant de bande passante que de ressources physiques au niveau des nœuds (batterie, mémoire, capacité de traitement, etc.).

Nous sommes alors partis de l'hypothèse que nous devrions améliorer les mécanismes de contrôle de topologie dans les réseaux *Ad hoc* pour permettre aux protocoles de routage de mieux performer et d'optimiser l'utilisation des ressources radio et des capacités des nœuds. L'originalité de nos travaux réside dans les techniques utilisées pour construire et maintenir des structures hiérarchiques servant de dorsale virtuelle et créant une vue logique différente

de la topologie radio. Cette vue peu changeante et robuste, peut être exploitée d'une manière plus performante par tout protocole des couches supérieures.

Nous avons commencé par découper la topologie du réseau en clusters. Chaque cluster étant géré par un nœud *clusterhead* servant d'une station de base temporaire pour les autres nœuds du cluster. Les nœuds non *clusterhead* sont ainsi soulagés, ils ont moins de tâches à exécuter tant que le *clusterhead* est aussi puissant. Ceci permet une économie d'énergie considérable et une réduction avantageuse des interférences. Le choix des nœuds *clusterhead* est un élément clé dans tout algorithme de clustérisation. Nous avons proposé une nouvelle façon d'élire les nœuds les plus stables en tant que *clusterhead*, ces nœuds doivent garder leur rôle de chef pendant un laps de temps suffisant. Nous avons combiné plusieurs métriques que nous jugeons pertinentes pour créer un poids à assigner aux nœuds du réseau. Les métriques reflètent la mobilité, le niveau d'énergie restante, la puissance de transmission et les capacités de chaque nœud. Un nœud ayant le meilleur poids est élu *clusterhead* jusqu'à temps où un autre nœud vient le remplacer.

La dorsale virtuelle est ainsi constituée d'un ensemble de nœuds *clusterhead* suffisamment stables. Selon nous, la force et l'originalité de notre proposition résident dans l'adaptation des bénéfices possibles de cette dorsale dans les protocoles de routage existants. Il serait possible de reprendre ces protocoles existants et les appliquer directement entre les clusters, augmentant ainsi leur passage à l'échelle. Cette hiérarchisation permet notamment de séparer le routage intraclusters du routage interclusters. À l'intérieur de nos *1-clusters*, les communications passent par le biais du *clusterhead* qui relaye le trafic entre les membres. Entre les clusters, un routage de type DSR pourrait être utilisé. Cependant, tout protocole réactif ou proactif conviendrait également.

Après avoir choisi les nœuds *clusterhead*, nous avons proposé des procédures tant pour la formation que pour la maintenance des clusters, profitant de toutes les lacunes que nous avons trouvées dans les approches de clustérisation de la littérature. En effet, les *clusterhead* intègrent des mécanismes de contrôle d'admission et de gestion des *handoff* permettant

d'équilibrer équitablement la charge entre les différents clusters de sorte que les paramètres de la qualité de service ne se détériorent radicalement pour les applications utilisatrices. Parallèlement, nous avons proposé un modèle analytique modélisant les différents paramètres de la couche MAC utilisée, et permettant de faire à tout moment des estimations sur le débit, le délai et le taux de perte aussi bien dans le cluster que dans tout le réseau. En outre, une autre contribution s'avère très importante par l'exploitation des capacités de technique CDMA et par l'introduction de plusieurs codes CDMA afin de minimiser les interférences intraclusters, maximiser les débits et réduire l'impact du trafic de contrôle qui utilise dorénavant un code de signalisation CDMA spécifique.

Par ailleurs, nous avons implémenté le modèle dans un environnement simulant la réalité d'un réseau *Ad hoc*. Ensuite, nous avons fait une étude des performances globales du réseau tant par des simulations que par des analyses numériques. Nous avons pu extraire les paramètres que nous jugeons pertinents pour prouver la stabilité et la robustesse des nos clusters suite à la mobilité, aux changements topologiques et aux défaillances éventuelles des conditions des nœuds. Lorsque le nombre de nœuds augmente dans le réseau, nous avons remarqué que plus de clusters avaient été produits, n'allongeant que modérément la longueur des routes en termes de nombre de sauts. Toutefois, la connectivité des nœuds s'approche de 100 % même pour des vitesses de l'ordre de 10 Km/h. Ainsi, nous pouvons conclure que le modèle proposé passe à l'échelle en nombre de nœuds tout en garantissant un niveau excellent de stabilité.

RECOMMANDATIONS ET PERSPECTIVES

Nombreuses sont les solutions qui ont été proposées pour les réseaux *Ad hoc*. Cependant, il reste beaucoup de travail à faire. La diversité des travaux de recherche permet sans doute d'avancer la progression dans ces réseaux. Toutefois, nous pensons que ceci pourrait, d'une part, désavantager les chercheurs eux-mêmes parce qu'il est extrêmement coûteux en temps de suivre cette rapidité de développement et de maintenir ses connaissances sur les plus récents travaux, et d'autre part, ceci ne permettrait pas d'unifier toutes les solutions dans une approche commune traitant à la fois toutes les problématiques rencontrées.

En effet, chaque problématique est traitée séparément et les tests effectués pour étudier les performances et mettre en perspective les propositions sont très différents. Les topologies ainsi que les types de trafic utilisés sont propres à chaque approche. Ainsi, il serait souhaitable d'avoir une plateforme unifiée à la portée de tous les chercheurs pour servir à la validation de leurs approches dans des environnements réalistes et communs.

Dans le but de limiter les incertitudes sur les performances réelles de notre proposition, nous avons tenté de faire la validation tant par des analyses numériques que par des simulations. Toutefois, il serait intéressant d'étudier analytiquement les caractéristiques de stabilité des clusters et de mettre en œuvre l'architecture proposée pour réaliser une évaluation réelle dans différents champs d'application. Nous avons également supposé qu'un intervalle de 5 secondes était suffisant pour collecter des informations sur les états des nœuds; nous pensons que ce point devrait être largement étudié afin d'optimiser la fréquence à laquelle les messages *hello* devraient être échangés sans avoir à pénaliser les performances du réseau.

Dans le futur, il serait intéressant de penser à des solutions plus avancées à base d'une dorsale constituée de plusieurs *clusterhead*. En effet, l'adressage, l'autoconfiguration ainsi que le routage avec qualité de service et le routage *multicast* représentent de forts candidats. Nous pensons que la stabilité de cette dorsale permettrait à tout protocole de niveau

supérieur de mieux performer. Il est aussi très intéressant de veiller sur l'avancement des techniques de modulation telles que la technologie MIMO qui est actuellement en forte croissance. Imaginons un nœud *clusterhead* utilisant MIMO, pouvant émettre et recevoir simultanément. Ceci permettrait de mieux gérer les communications intraclusters et interclusters, d'augmenter énormément les débits et d'éviter le délai qu'un *clusterhead* non MIMO doit subir étant donné qu'il est incapable de fonctionner en mode émission/réception simultané. Ainsi, les *clusterhead* auraient moins de données à faire attendre dans leurs buffers, surtout durant les périodes d'apogée de communications dans le réseau.

D'autre part, nous savons qu'au sein du cluster, les nœuds sont diversement éloignés de leur *clusterhead*, ce dernier doit normalement percevoir de la même manière tous les niveaux de puissances des nœuds membres émetteurs. Ces niveaux tels qu'ils sont reçus par le *clusterhead* varient en fonction de l'éloignement de chaque nœud membre. Dans nos simulations, nous avons choisi des nœuds ayant des puissances de transmission aléatoires. Toutefois, il serait intéressant de penser à éviter le problème du proche/éloigné (*Near Far Problem)*[13] rencontré dans des systèmes utilisant la technique CDMA. Il serait souhaitable de mesurer les performances en adoptant comme règle que les nœuds membres transmettent vers leur *clusterhead* avec un niveau de puissance inverse de celui en provenance du *clusterhead*. En d'autres termes, un nœud membre recevant un signal faible du *clusterhead* emploiera davantage de puissance qu'un autre nœud recevant un signal fort. Notons que dans le cas où le nœud perd sa connectivité avec son *clusterhead*, le nœud devrait réinitialiser sa puissance de transmission à la valeur par défaut.

Dans cette thèse, nous avons pris comme hypothèse que l'environnement est homogène : La topologie est soit une salle de conférence, soit un campus universitaire, où un modèle de mobilité comme le *Random Waypoint Model* serait acceptable. Dans ce contexte, nous avons paramétré les facteurs de poids pour refléter une topologie homogène en termes de

---

[13] Les communications des nœuds proches du *clusterhead* peuvent masquer celles des nœuds éloignés

besoins des nœuds. Ainsi, plus d'investigations doivent être faites lorsque la topologie est hétérogène : comment le modèle réagirait-il si nous disposions d'un groupe de nœuds ayant des contraintes totalement différentes en termes de modèles de mobilité, d'énergie, de capacité de traitement, etc.

D'ailleurs, l'interaction entre les couches ou le *Cross-Layer* connaît actuellement un fort essor dans les réseaux *Ad hoc*. Nous pensons qu'un modèle *Cross-Layer* appliqué à un cluster pourrait garantir une meilleure qualité de service aux applications et de maximiser la stabilité de ces structures. L'utilisation des mécanismes de prédiction de mouvement au sein de chaque cluster permettrait aux nœuds *clusterhead* de garder une trace permanente de l'emplacement de ses nœuds, d'avoir une vision globale sur l'état de chaque nœud (son énergie, sa mobilité, etc.) et de prédire le prochain saut sur le meilleur chemin vers la destination.

Finalement, une autre question pourrait se poser : qu'en est-il de l'exploitation des clusters dans d'autres domaines ? Les réseaux de capteurs sont considérés comme un cas spécial des réseaux *Ad hoc*. Ces réseaux souvent utilisés pour des fins de surveillance d'évènements et de détection d'incendies ont des contraintes spécifiques différentes. La topologie est plus redondante et l'économie d'énergie représente l'objectif primordial. Est-ce que ce type de réseaux nécessite-t-il une clustérisation ?

ANNEXE I

LISTE DES PUBLICATIONS

Journal

El Bazzal, Zouhair, Michel Kadoch, Basile Agba et François Gagnon. 2008. « Improve the Performance of a Clustered Mobile *Ad hoc* Networks using a CDMA Single Channel and Based on Admission Control Approach » In *WSEAS Journal of Transactions on Communications.* vol. 7, n° 2. p. 68-82. USA : WSEAS

ABSTRACT

Clustering has been proven to be a promising approach for mimicking the operation of the fixed infrastructure and managing the resources in multi-hop networks. In this paper, we propose an Efficient Clustering Algorithm (ECA) in Mobile *Ad hoc* Networks based on the quality of service's (QoS) parameters (cluster throughput/delay). The goals are yielding low number of clusters, maintaining stable clusters, and minimizing the number of invocations for the algorithm. The performance changes greatly for small and large clusters and depends strongly on the formation and maintenance procedures of clusters which should operate with minimum overhead, allowing mobile nodes to join and leave without perturbing the membership of the cluster and preserving current cluster structure as much as possible. In this manner, while QoS does not perform well under high traffic load conditions, admission control becomes necessary in order to provide and support the QoS of existing members. Based on the results from the proposed analytical model, we implement a new admission control algorithm that provides the desired throughput and access delay performance in order to determine the number of members inside an ECA cluster that can be accommodated while satisfying the constraints imposed by the current applications. Through numerical analysis and simulations, we have studied the performance of our model and compared it with that of WCA. The results demonstrate the superior performance of the proposed model.

Keywords: Clustering, Weight, Election, Throughput, Delay.

Conférences internationales

El Bazzal, Zouhair, Michel Kadoch, François Gagnon et Maria Bennani. 2007. « Optimizing the Performance of the Generated Clusters in Mobile *Ad hoc* Networks ». In *The 11th World Multi-Conference on Systemics, Cybernetics and Informatics WMSCI 2007.* (Orlando, 2007), p. 12-17. USA : IIS.

ABSTRACT

In this paper, we investigate an analytic model to optimize the performance of the clusters generated by our clustering algorithm FWCA (Flexible Weight based Clustering Algorithm) in which the *clusterhead* is at a single-hop wireless link from all its members. In particular, we show that in clustered networks, the size of the cluster in terms of number of members is important and can be expressed as function of the collision probability encountered on the channel, the average contention window and the current traffic inside the cluster. We also demonstrate how the parameters "achievable throughput, cluster delay and transmissions' number per packet" change greatly for small and large clusters. Thus, the results of our model will be used for providing probabilistic quality of service guarantees and determining the number of members inside a FWCA cluster that can be accommodated while satisfying the constraints imposed by the applications. We finally validate the results through a numerical analysis and simulations.

Keywords: *Ad hoc* Networks, Throughput, Delay, Capacity, Clusters.

El Bazzal, Zouhair, Michel Kadoch, Basile Agba, François Gagnon et Maria Bennani. 2006. « A Flexible Weight Based Clustering Algorithm in Mobile *Ad hoc* Networks ». *In IEEE International Conference on Systems and Networks Communication (ICSNC'06)*. (Tahiti, 2006). p. 50-57. USA : IEEE.

ABSTRACT

Clustering has been proven to be a promising approach for mimicking the operation of the fixed infrastructure and managing the resources in multi-hop networks. In order to achieve good performance, the formation and maintenance procedure of clusters should operate with minimum overhead, allowing mobile nodes to join and leave without perturbing the membership of the cluster and preserving current cluster structure as much as possible. In this paper, we propose a Flexible Weight Based Clustering Algorithm (FWCA) in Mobile *Ad hoc* Networks. The goals are yielding low number of clusters, maintaining stable clusters, minimizing the number of invocations for the algorithm and maximizing lifetime of mobile nodes in the system. Through simulations we have compared the performance of our algorithm with that of WCA in terms of the number of clusters formed, number of re-affiliations, number of states transitions on each *clusterhead* and number of *clusterhead* changes. The results demonstrate the superior performance of the proposed algorithm.

Keywords: Clustering Algorithm, Weight, Election.

El Bazzal, Zouhair, Michel Kadoch, Basile Agba, François Gagnon et Maria Bennani. 2006. « An Efficient Management Algorithm for Clustering in Mobile *Ad hoc* Networks ». In *Proceedings of the ACM international workshop on Performance Monitoring, Measurement, and evaluation of Heterogeneous Wireless and Wired Networks PM2HW2N'06*. (Spain, 2006). p. 25-31. ACM.

ABSTRACT

Clustering of mobile nodes among separate domains has been proposed as an efficient approach to mimic the operation of the fixed infrastructure and manage the resources in multi-hop networks. In this work, we propose a new clustering algorithm, namely Efficient Management Algorithm for Clustering (EMAC) based on weighting parameters. The goals are yielding low number of clusters, maintaining stable clusters, minimizing the number of invocations for the algorithm and maximizing lifetime of mobile nodes in the system. Through simulations we have compared the performance of our algorithm with that of WCA in terms of the number of clusters formed and number of states transitions on each *clusterhead*. The results demonstrate the superior performance of the proposed algorithm.

Keywords: MANET, Clustering Algorithm, Weight, Election.

LISTE DE RÉFÉRENCES

01net. 2008. « Mimo, la technique expérimentale qui accélère le sans-fil ». En ligne. <http://www.01net.com/article/212443.html>. Consulté le 8 mai 2008.

Aad, Imad et Claude Castelluccia. 2001. « Differentiation Mechanisms for IEEE 802.11 ». In *Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM 01*. Vol. 1, p. 209-218. USA : IEEE.

Adibi, Sasan et Shervin Erfani. 2006. « multipath routing survey for mobile ad-hoc networks ». In *3rd IEEE Consumer Communications and Networking Conference CCNC 2006*. vol. 2, p. 984- 988. IEEE.

Adjih, Cédric, Saadi Boudjit, Philippe Jacquet, Anis Laouiti et Paul Mühlethaler. 2005. *Un mécanisme de configuration et de détection d'adresse dupliquée pour OLSR fonctionnant avec des interfaces multiples*. Coll. « Rapport de recherche de l'INRIA », RR-5747. Sophia Antipolis (France) : Institut National de Recherche en Informatique et Automatique, 31 p.

Agba, Basile, François Gagnon et Ammar Kouki. 2006a. « Tactical ad hoc scenarios generator coupled with channel modelling ». In *IEEE wireless and microwave technology conference*. (Florida, 2006), p. 1-5. USA : IEEE.

Agba, Basile, Grace Amoussou, Zbiginew Dziong, Michel Kadoch et François Gagnon. 2006b. « Performances analysis of mobile ad hoc routing protocols under realistic mobility and power models ». In *OPNETWORK* 2006. (Washington, 2002). USA : OPNET.

Ahn, Gahng-Seop, Andrew T. Campbell, Andras Veres et Li-Hsiang Sun. 2002. « Supporting service differentiation for real-time and best-effort traffic in stateless wireless ad hoc networks (SWAN) ». In *IEEE TRANSACTIONS ON MOBILE COMPUTING*. vol. 1, n° 3, p. 192-207. USA : IEEE.

Alipour, Hamidreza, Maghsoud Abbaspour, Mostafa Esmaeili, Hamed Mousavi et Hamed Shahhoseini. 2007. « DACA: Dynamic Advanced Clustering Algorithm for Sensor Networks ». In *14th IEEE International Conference on Electronics, Circuits and Systems, 2007 (ICECS 2007)*. p. 518-525. USA : IEEE.

Alzoubi, Khaled, Peng-Jun Wan et Ophir Frieder. 2002. « Message-Optimal Connected Dominating Sets in Mobile *Ad hoc* Networks ». In *Proceedings of the 3rd ACM international symposium on Mobile Ad hoc networking & computing MobiHoc'02*. ACM.

Amis, Alan D. et Ravi Prakash. 2000. « Load-Balancing Clusters in Wireless *Ad hoc* Networks ». In *Proceedings 3rd IEEE Application-Specific Systems and Software Engineering Technology*. p. 25–32. USA : IEEE.

Amis, Alan D., Ravi Prakash, Thai H.P. Vuong et Dung T. Huynh. 1999. « Max-min d-cluster formation in wireless ad hoc networks » In *INFOCOM Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies Proceedings*. (Tel-Aviv, 1999), vol. 1, p. 32-41. IEEE.

An, Beongku et Symeon Papavassiliou. 2001. « A mobility-based clustering approach to support mobility management and multicast routing in mobile ad-hoc wireless networks ». In *International Journal of Network Management*. vol. 11, n° 3, p. 387-395. John Wiley & Sons, Inc.

Bannerjee, Suman et Samir Khuller. 2001. « A clustering scheme for hierarchical control in wireless networks ». In *INFOCOM*. (Alaska, 2002), p. 1028-1037. USA : IEEE.

Bao, Lichun et J. J. Garcia Luna Aceves. 2003. « Topology management in ad hoc networks ». In *Proceedings of the 4th ACM international symposium on Mobile Ad hoc networking & computing (MOBIHOC)*. (Anapolis USA, June 2003), p. 129-140. ACM.

Basagni, Stefano. 1999a. « Distributed clustering for ad hoc networks ». In *Proceedings of International Symposium on Parallel Architectures, Algorithms and Networks*. p. 310-315. IEEE.

Basagni, Stefano. 1999b. « Distributed and mobility-adaptive clustering for multimedia support in multi-hop wireless networks ». In *Proceedings of Vehicular Technology Conference (VTC)*. vol. 2, p. 889-893. IEEE.

Basagni, Stefano, Damla Turgut et Sajal K. Das. 2001. « Mobility-adaptive protocols for managing large ad hoc networks ». In *International Conference on Communications (ICC'01)*. (Helsinki Finland, June 2001), vol.5, p. 1539-1543, IEEE.

Basu, Prithwish, Naved Khan et Thomas D. C. Little. 2001. « A mobility based metric for clustering in mobile ad hoc networks ». In *International Conference on Distributed Computing Systems Workshop*. p. 413-418. IEEE.

Benzaid, Mounir, Pascale Minet et Khaldoun Al Agha. 2002. *Intégration de la gestion de la mobilité rapide dans le protocole de routage OLSR*. Coll. « Rapport de recherche de l'INRIA », RR-4510. Sophia Antipolis (France) : Institut National de Recherche en Informatique et Automatique, 12 p.

Bettstetter, Christian. 2004. « The cluster density of a distributed clustering algorithm in ad hoc networks ». In *IEEE International Conference on Communications*. vol. 7, p. 20-24. IEEE.

Bharghavan, Vaduvur, Alan Demers, Scott Shenker et Lixia Zhang. 1994. « MACAW: A media access protocol for wireless LAN's ». In *Proceedings of the SIGCOMM '94 Conference on Communications Architectures, Protocols and Applications*. p. 212-225. ACM.

Bianchi, Giuseppe. 2000. « Performance Analysis of the IEEE 802.11 Distributed Coordination Function ». In *IEEE Journal of Selected Areas in Communications (JSAC)*. vol. 18, n° 3, p. 535-547. IEEE

Butenko, S., X. Cheng, D.-Z. Du et P. M. Pardalos. 2003. « On the construction of virtual backbone for ad hoc wireless networks ». In *Cooperative Control: Models, Applications and Algorithms*. volume 1 of Cooperative Systems, p. 43-54. Kluwer Academic Publishers.

Camp, Tracy, Jeff Boleng et Vanessa Davies. 2002. « A survey of mobility models for ad hoc network research ». In *Wireless Communications and Mobile Computing (WCMC) : Special issue on Mobile Ad Hoc Networking : Research, Trends and Applications*, vol. 2, p. 483-502.

Carle, Jean et David Simplot-Ryl. 2004. « Energy efficient area monitoring by sensor networks ». In *IEEE Computer Magazine*, vol. 37, n° 2, p. 40-46. USA : IEEE.

Chao, Chih-Min, Jang-Ping Sheu et Cheng-Ta Hu. 2003. « Energy-Conserving GRID Routing Protocol in Mobile *Ad hoc* Networks ». In *Proceedings of the 2003 International Conference on Parallel Processing (ICPP'03)*. p. 265-272. IEEE.

Chatterjee, Mainak, Sajal K. Das, Damla Turgut. 2002. « WCA: A Weighted Clustering Algorithm for Mobile *Ad hoc* Networks ». In *Cluster Computing Journal*. vol. 5, n° 2, p. 193-204. Springer.

Chaudet, Claude, Dominique Dhoutaut et Isabelle Guérin Lassous. 2005. « Performance issues with IEEE 802.11 in *Ad hoc* networking ». In *IEEE Communications Magazine*. vol. 43, n° 7, p. 110-116. USA : IEEE.

Chelius, Guillaume et Éric Fleury. 2002. *Ananas : Un nouveau schéma architectural pour réseaux adhoc*. Coll. « Rapport de recherche de l'INRIA », RR-4354. Sophia Antipolis (France) : Institut National de Recherche en Informatique et Automatique, 22 p.

Chen, Geng, Fabian Garcia Nocetti, Julio Solano Gonzalez et Ivan Stojmenovic. 2002. « Connectivity based k-hop clustering in wireless networks ». In *Proceedings of the 35th Hawaii International Conference on System Sciences*. (Hawaii, January 2002), p. 2450-2459. USA : IEEE.

Chen, Yuanzhu Peter et Arthur L. Liestman. 2002. « Approximating minimum size weakly-connected dominating sets for clustering mobile *Ad hoc* networks ». In *Proceedings of the 3rd ACM international symposium on Mobile Ad hoc networking & computing (MOBICOM 02)*. (Lausanne, 2002), p. 165-172. ACM.

Cheng, Xiuzhen et Ding-Zhu Du. 2002. *Virtual backbone-based routing in multihop Ad hoc wireless networks*. Coll. « Technical Report », TR 02-002. University of Minnesota : USA, 23 p.

Chiang, Ching-Chuan, Hsiao-Kuang Wu, Winston Liu et Mario Gerla. 1997. « Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel ». In

Proceedings of IEEE Singapore International Conference on Networks (SICON). (Singapore, 1997), p. 197-211. USA : IEEE.

Choi, Jin-Young, Joon-Sic Cho, Seon-Ho Park et Tai-Myoung Chung. 2008a. « A Clustering Method of Enhanced Tree Establishment in Wireless Sensor Networks ». In *10th International Conference on Advanced Communication Technology, 2008. ICACT 2008.* vol. 2. p. 1103-1107. USA : IEEE.

Choi, Soon-Hyeok, Dewayne E. Perry et Scott M. Nettles. 2008b. « A Software Architecture for Cross-Layer Wireless Network Adaptations ». In *The Seventh Working IEEE/IFIP Conference on Software Architecture (WICSA 2008).* p. 281-284. USA : IEEE.

Clark, B. N., C. J. Colburn et D. S. Johnson. 1990. « Unit disks graphs ». In *Discrete Mathematics, 86.* p. 165-177.

Claussen, T. et P. Jacquet. 2003. *Optimized Link State Routing Protocol (OLSR).* RFC 3626. USA : Internet Engineering Task Force, 75 p.

Cordeiro, Carlos, Kiran Challapali, Dagnachew Birru et Sai Shankar N. 2005. « IEEE 802.22: the first worldwide wireless standard based on cognitive radios ». In *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks DySPAN 2005.* p. 328-337. USA : IEEE.

Costa, Luis Henrique M. K., Marcelo Dias De Amorim et Serge Fdida. 2004. « Reducing latency and overhead of route repair with controlled flooding ». In *Wireless Networks.* vol.10, n° 4, p. 347-358. Kluwer Academinc Publishers.

Das, Bevan et Vaduvur Bharghavan. 1997. « Routing in ad-hoc networks using minimum connected dominating sets ». In *IEEE International Conference on Communications (ICC 97).* (Montreal, 8-12 June 1997), vol. 1, p. 376-380. USA : IEEE.

Dharmaraju, Dinesh, Ayan Roy-Chowdhury, Pedram Hovareshti et John S. Baras. 2002. « INORA-a unified signaling and routing mechanism for QoS support in mobile *Ad hoc* networks ». In *Proceedings of the International Conference on Parallel Processing Workshops (ICPPW'02).* p. 86-93. USA : IEEE.

Ephremides, Anthony, Jeffrey E. Wieselthier et Dennis J. Baker. 1987. « A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling ». In *Proceedings of the IEEE*, vol. 75, no. 1, p. 56-73. USA : IEEE.

Fernandess, Yaacov et Dahlia Malkhi. 2002. « K-clustering in wireless ad hoc networks ». In *International Workshop on Principles of Mobile Computing (POMC)*. (Toulouse, 2002), p. 31-37. ACM.

Gerla, Mario, Xiaoyan Hong et Guangyu Pei. 2000. « Landmark routing for large *Ad hoc* wireless networks ». In *Global Telecommunications Conference (GLOBECOM)*. (San Francisco, 2000), vol. 3. p. 1105-1116. USA : IEEE.

Gerla, Mario et Jack Tzu-Chieh Tsai. 1995. « Multicluster, Mobile, Multimedia Radio Network ». In *ACM/Baltzer Wireless Networks Journal*. vol. 1, n° 3, p. 255-256. Kluwer Academic Publishers.

Günes, M. et J. Reibel. 2002. « An IP Address Configuration Algorithm for Zeroconf Mobile Multihop *Ad hoc* Networks ». In *Proceedings of the International Workshop on Broadband Wireless Ad-Hoc Networks and Services*. (Sophia Antipolis, 2002).

Haas, Zygmunt J. et Marc R. Pearlman. 2001. « The performance of query control schemes for the zone routing protocol ». In *IEEE/ACM Transactions on Networking (TON)*. vol. 9, n° 4, p. 347-358. USA : IEEE.

Hong, Xiaoyan, Mario Gerla, Guangyu Pei et Ching-Chuan Chiang. 1999. « A Group Mobility Model for *Ad hoc* Wireless Networks ». In *Proceedings of the 2nd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*. (Seattle USA, 1999), p. 53-60. ACM.

Hou, Ting-Chao et Tzu-Jane Tsai. 2001. « An access-based clustering protocol for multihop wireless *Ad hoc* networks ». In *IEEE Journal on Selected Areas in Communications*. vol. 19, n° 7, p. 1201-1210. IEEE.

Infonetics Research. 2008a. « ACCURATE. OBJECTIVE. PROVEN, Research and analysis you can rely on ». En ligne. <http://www.infonetics.com/pr/2007/ms07.ran.4q06.nr.asp>. Consulté le 25 avril 2008.

Infonetics Research. 2008b. « ACCURATE. OBJECTIVE. PROVEN, Research and analysis you can rely on ». En ligne. <http://www.infonetics.com/pr/2007/ms07.wl.1q07.nr.asp>. Consulté le 25 avril 2008.

Institute of Electrical and Electronics Engineers. 1999. *IEEE Std 802.11: Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. Norme Internationale. USA : IEEE, 528 p.

Institute of Electrical and Electronics Engineers. 2002. *IEEE Std 802.15.1: Standard for Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs)*. Norme Internationale. USA : IEEE, 580 p.

Jiang, Hai, Weihua Zhuang et Xuemin Sherman Shen. 2007. « Distributed Medium Access Control Next-Generation CDMA Wireless Networks ». In IEEE Wireless Communications Journal. vol. 14, n° 3, p. 25-31. USA : IEEE.

Jiang, Mingliang, J. Li et Y. C. Tay. 1999. *Cluster based routing protocol (CBRP)*. IETF Internet draft version 01. USA : Internet Engineering Task Force, 27 p.

Johansson, Per, Tony Larsson, Nicklas Hedman et Bartosz Mielczarek. 1999. « Scenario-based performance Analysis of Routing Protocols for Mobile *Ad hoc* Networks ». In *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*. p. 195-206. ACM.

Johnson, D. B., D. A. Maltz et Y.-C. Hu. 2007. *The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4*. RFC 4728. USA : Internet Engineering Task Force, 107 p.

Jurdak, Raja, Cristina Videira Lopes et Pierre Baldi. 2004. « A Survey, Classification and Comparative Analysis of Medium Access Control Protocols for *Ad hoc* Networks ». In *IEEE Communications Surveys and Tutorials FIRST QUARTER*. vol.6, n° 1. IEEE.

Kadoch, Michel. 2004. *Protocoles et réseaux locaux : l'accès Internet*. Montréal : École de technologie superieure. 529 p.

Karp, Brad et H. Kung. 2000. « GPSR : Greedy perimeter stateless routing for wireless networks ». In *The 6th annual International Conference on Mobile Computing and Networking (MOBICOM)*. (Boston, 2000), p. 243-254. ACM.

Ke, Bing Wen, Ying Jun Zhang et Soung Chang Liew. 2007. « Media Access Control with Spatial Correlation for MIMO Ad Hoc Networks ». In *IEEE International Conference on Communications ICC '07*. p. 3660-3665. USA : IEEE.

Ko, Young-Bae et Nitin H. Vaidya. 1998. « Location-aided routing (lar) in mobile *Ad hoc* networks ». In *Conference on Mobile Computing and Networking (MOBICOM)*. (Dallas, 1998), vol. 6, n° 4, p. 66-75. Kluwer Academinc Publishers.

Kozat, Ulas C., George Kondylis, Bo Ryu et Mahesh K. Marina. 2001. « Virtual dynamic backbone for mobile ad hoc networks ». In *International Conference on Communications (ICC'01)*. (Helsinki Finland, June 2001), vol. 1, p. 250-255. IEEE.

Kurose, James et Keith Ross. 2003. *Analyse structurée des réseaux : Des applications de l'Internet aux infrastructures de télécommunication*, 2e éd. Pearson Education. 790 p.

Kwon, Taek Jin et Mario Gerla. 2002. « Efficient flooding with passive clustering (pc) in ad-hoc networks ». In *ACM SIGCOMM Computer Communication Review*. vol. 32, p. 44-56. ACM.

Lavoie, Michel. 2007. *LOG-610 : Réseaux de télécommunication*. Notes du cours LOG-610 du programme de baccalauréat en génie logiciel et des TI, Montréal : École de Technologie Supérieure, premier cours, 90 acétates.

Lee, Seoung-Bum, Gahng-Seop Ahn, Xiaowei Zhang et Andrew T. Campbell. 2000. « INSIGNIA: An IP-Based Quality of Service Framework for Mobile *Ad hoc* Networks ». In *Journal of Parallel and Distributed Computing 60*. p. 374-406. Academic Press.

Lian, Jie, Kshirasagar Naik et Gordon B. Agnew. 2007. « A Framework for Evaluating the Performance of Cluster Algorithms for Hierarchical Networks ». In *IEEE/ACM TRANSACTIONS ON NETWORKING*. vol. 15, n° 6. USA : IEEE.

Liang, Ben et Zygmunt J. Haas. 2000. « Virtual backbone generation and maintenance in ad hoc network mobility management ». In *The Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM.* (Tel-Aviv Israel, March 2000), p. 1293-1302, IEEE.

Liao, Wen-Hwa, Yu-Chee Tseng et Jang-Ping Sheu. 2001. « GRID: A Fully Location-Aware Routing Protocol for Mobile *Ad hoc* Networks ». In *Telecommunication Systems*. vol. 18, n° 1, p. 37-60.

Lin, An-Tai et Shie-Jue Lee. 2003. « A Modified Distributed Coordination Function for Real Time Traffic in IEEE 802.11 Wireless LAN ». In Proceedings of the 17th International Conference on Advanced Information Networking and Applications (AINA'03). p. 794-797. IEEE.

Lin, Chunhung Richard et Mario Gerla. 1997. « Adaptive Clustering for Mobile Wireless Networks ». In *JOURNAL ON SELECTED AREAS IN COMMUNICATIONS JSAC*. vol. 15, n° 7, p. 1265-1275. IEEE.

Lin, J.-H., C.-R. Dow et S.-F. Hwang. 2003. « A distributed virtual backbone development scheme for ad-hoc wireless networks ». In *Wireless Personal Communications*. vol. 27, n° 8, p. 215-233.

Liu, Jian et Sing Suresh. 2001. « ATCP: TCP for Mobile *Ad hoc* Networks ». In *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS*. vol. 19, n° 7, p. 1300-1315. IEEE.

Madan, Ritesh, Shuguang Cui, Sanjay Lall et Andrea Goldsmith. 2006. « Cross-layer design for lifetime maximization in interference-limited wireless sensor networks ». In *IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS*. vol. 5, n° 11. IEEE.

McDonald, Bruce et Taib F. Znati. 1999. « A Mobility-Based Framework for Adaptive Clustering in Wireless *Ad hoc* Networks ». In *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS IEEE/JSAC*. vol. 17, n° 8. IEEE.

McDonald, Bruce et Taib F. Znati. 2001. « Design and Performance of a Distributed Dynamic Clustering Algorithm for Ad-Hoc Networks ». In *Proceedings 34th Annual Simulation Symposium*. p. 27-35.

Meraihi, Rabih, Gwendal Le Grand, Nicolas Puech, Michel Riguidel et Samir Tohmé. 2004. « Improving ad hoc network performance with backbone topology control ». In *Vehicular Technology Conference (VTC)*. (Los Angeles, 2004), vol. 4, p. 2829-2833. IEEE.

Min, Manki, Feng Wang, Ding-Zhu Du et Panos. M. Pardalos. 2004. « A Reliable Virtual Backbone Scheme in Mobile Ad-hoc Networks ». In *International Conference on Mobile Ad-hoc and Sensor Systems (MASS)*. (Fort Lauderdale USA, October 2004), p. 60-69. IEEE.

Mnif, Kais. 2006. « Construction et maintenance d'une dorsale virtuelle dans les réseaux *Ad hoc* mobiles ». Thèse de doctorat en génie électrique, Montréal, École de technologie supérieure, 205 p.

Mohsin, Mansour et Ravi Prakash. 2002. « IP Address Assignment in a Mobile *Ad hoc* Network ». In IEEE MILCOM 2002. (Anaheim, 2002), vol. 2, p. 856-861. IEEE.

Murthy, M. C. Siva Ram et B. S. Manoj. 2004. *Ad hoc Wireless Networks: Architectures and Protocols*. Prentice Hall PTR. 880 p.

Nesargi, Sanket et Ravi Prakash. 2002. « MANETconf: Configuration of Hosts in a Mobile *Ad hoc* Network ». In *Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM 2002*. (New York, 2002), vol. 2, p. 1059-1068. IEEE.

Nikaein, Navid et Christian Bonnet. 2004. « Topology management for improving routing and network performances in mobile *Ad hoc* networks ». In *ACM MONET*. vol. 9, n° 6, p. 583-594. Kluwer Academic Publishers.

Ohta, Tomoyuki, Shinji Inoue et Yoshiaki Kakuda. 2003. « An Adaptive Multihop Clustering Scheme for Highly Mobile *Ad hoc* Networks ». In *Proceedings of the Sixth International Symposium on Autonomous Decentralized Systems (ISADS'03)*. p. 293-300. USA : IEEE.

Oliveira, Rodolfo, Luis Bernardo et Paulo Pinto. 2006. « Performance Analysis of the IEEE 802.11 Distributed Coordination Function with Unicast and Broadcast Traffic ». In

The 17th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'06). p. 1-5. IEEE.

Parekh, Abhay K. 1994. « Selecting routers in ad-hoc wireless networks ». In *Proceedings of the SBT/IEEE International Telecommunications Symposium.*

Pearlman, Marc R. et Zygmunt J. Haas. 1999. « Determining the optimal configuration of the zone routing protocol ». In *IEEE Journal on Selected Areas in Communications*. vol. 17, n° 8, p. 1395-1414. IEEE.

Perkins, Charles E. 2001. *Ad hoc networking*, 1st ed. Addison-Wesley Professional, 384 p.

Perkins Charles E., E. Belding-Royer et S. Das. 2003. « *Ad hoc* On-Demand Distance Vector (AODV) Routing ». RFC 3561. USA : Internet Engineering Task Force, 37 p.

Perkins, Charles E. et Pravin Bhagwat. 1994. « Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers ». In *Proceedings of the conference on Communications architectures, protocols and applications (SIGCOMM)*. (London, 1994), p. 234-244. ACM.

Perkins, Charles E., Jari T. Malinen, Ryuji Wakikawa, Elizabeth M. Belding-Royer et Yuan Sun. 2001. *IP Address Autoconfiguration for Ad hoc Networks*. IETF Internet draft version 01. USA : Internet Engineering Task Force, 13 p.

Pong, Denis et Tim Moors. 2003. « Call Admission Control for IEEE 802.11 Contention Access Mechanism ». In *Proceedings of IEEE GLOBECOM'03*. (San Francisco, 2003), vol.1, p. 174-178. IEEE.

Pujolle, Guy. 2005, *LES RÉSEAUX*, 5e éd. France : Éditions Eyrolles. 1094 p.

Qin, Y. et C. Lee. 2003. « An Improved Dynamic Source Routing Algorithm for MANET ». In *IEEE ICC 2003 Personal Communication Systems and Wireless LANs*. IEEE.

Raisinghani, Vijay T. et Sridhar Iyer. 2006. « Cross-layer feedback architecture for mobile device protocol stacks ». In *IEEE Communications Magazine*, vol. 44, n° 1, p. 85-92. IEEE.

Rajaraman, Rajmohan. 2002. « Topology control and routing in ad hoc networks: a survey ». In *ACM SIGACT News*. vol. 33, n° 2, p. 60-73. ACM

Ray, Saikat, Jeffrey B. Carruthers et David Starobinski. 2003. « RTS/CTS-Induced Congestion in Ad-Hoc Wireless LANs » In Wireless Comm. and Networking Conference (WCNC). (New Orleans, 2003), p. 1516-1521. USA : IEEE.

Reddy, T. Bheemarjuna, I. Karthigeyan, B.S. Manoj et C. Siva Ram Murthy. 2006. « Quality of service provisioning in *Ad hoc* wireless networks : a survey of issues and solutions ». In *Ad hoc Networks*, vol. 4, n° 1, p83-124. Elsevier.

Rieck, M. Q., S. Pai et Sukesh ad Dhar. 2005. « Distributed routing algorithms for wireless ad hoc networks using d-hop connected d-hop dominating sets ». In *Computer Networks*. vol. 47, n° 6, p. 785-799. Amsterdam (PAYS-BAS) : Elsevier Science.

Romdhani, Lamia, Qiang Ni et Thierry Turletti. 2003. « Adaptive EDCF: Enhanced Service Differentiation for IEEE 802.11 Wireless Ad-Hoc Networks ». In IEEE WCNC'03 (Wireless Communications and Networking Conference). (New Orleans, 2003), vol. 2, p. 1373-1378,. IEEE.

Rubin, Ishak, Xiaolong Huang, Y. C. Liu et Huei-jiun Ju. 2003. « A distributed stable backbone maintenance protocol for *Ad hoc* wireless networks ». In *The 57th IEEE Semiannual Vehicular Technology Conference (VTC)*. (Jeju, Korea, 2003), vol. 3, p. 2018-2022. IEEE.

Ryu, Bo, Jason Erickson, Jim Smallcomb et Son Dao. 1999. « Virtual wire for managing virtual dynamic backbone in wireless ad hoc networks ». In *International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIAL'M)*. (Seattle USA, August 1999), 8 p. IEEE.

Ryu, Jung-Hee, Sanghwa Song et Dong-Ho Cho. 2001. « New Clustering Schemes for Energy Conservation in Two-Tiered Mobile Ad-Hoc Networks ». In *IEEE Communications International* Conference ICC'01. vol. 3, p. 862-866. USA : IEEE.

Safwat, Ahmad, Hossam Hassanein et Hussein T. Mouftah. 2001. « Power-aware fair infrastructure formation for wireless mobile *Ad hoc* communications ». In *IEEE Global Telecommunications Conference GLOBECOM'01*. vol. 5, p. 2832-2836. IEEE.

Santivanez, César A., Bruce McDonald, Ioannis Stavrakakis et Ram Ramanathan. 2002. « On the scalability of *Ad hoc* routing protocols ». In *Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM 02*. (New York, 23-27 June 2002), vol. 3, p. 1688-1697. IEEE.

Sheltami, Tarek et Hussein T. Mouftah. 2003. « An efficient energy aware *clusterhead* formation infrastructure protocol for MANETs ». In *Proceedings of The Eighth IEEE International Symposium on Computers and Communication (ISCC 2003)*. vol. 1, p. 203-208. IEEE.

Sinha, Prasun, Raghupathy Sivakumar et Vaduvur Bharghavan. 2001. « Enhancing *Ad hoc* routing with dynamic virtual infrastructures ». In *The Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM*. (Alaska, 22-26 April 2001), vol. 3, p. 1763-1772. IEEE.

Sivakumar, Raghupathy, Prasun Sinha et Vaduvur Bharghavan. 1999. « CEDAR: a core-extraction distributed *Ad hoc* routing algorithm ». In *IEEE Journal on Selected Areas in Communications*. vol. 17, n° 8, p. 1454-1465. IEEE.

Srivastava, Vineet et Mehul Motani. 2005. « Cross-layer design: a survey and the road ahead ». In *IEEE Communications Magazine*. vol. 43, n°12. p. 112-119. IEEE.

Stojmenovic, Ivan, Mahtab Seddigh et Jovisa Zunic. 2001. « Dominating sets and neighbor elimination based broadcasting algorithms in wireless networks ». In *IEEE Transactions on Parallel and Distributed Systems*. vol. 12, n° 12, p. 14-25. IEEE.

Tan, Haixia, Weilin Zeng et Lichun Bao. 2005. « Patm : Priority-based adaptive topology management for efficient routing in *Ad hoc* networks ». In *International Conference in Computational Science (ICCS)*. (Atlanta, 22-25 May 2005), vol. 3515, p. 485-492. Springer.

Tanenbaum, Andrew. 2003. *Réseaux*, 4e éd. Pearson Education France. 908 p.

Tay, Y. C. et K. C. Chua. 2001. « A capacity analysis for the IEEE 802.11 MAC protocol ». In *Wireless Networks*. vol. 7, n° 2, p. 159-171. Kluwer Academic Publishers.

Toh, C.K. 2001. *Ad hoc Mobile Wireless Networks: Protocols and Systems*. Prentice Hall PTR, 336 p.

Tseng, Yu-Chee., Sze-Yao Ni et Yuh-Shyan Chen. 2002. « The broadcast storm problem in a mobile ad hoc network ». In *ACM Wireless Networks*, vol. 8. p.152-167. Kluwer Academic Publishers.

Turgut, Damla, Begumhan Turgut, Ramez Elmasri et Than V. Le. 2003. « Optimizing clustering algorithm in mobile ad hoc networks using simulated annealing ». In *Wireless Communications and Networking Conference (WCNC)*. (New Orleans, 2003), vol. 3, p. 16-20. USA : IEEE.

Vaidya, Nitin, Anurag Dugar, Seema Gupta et Paramvir Bahl. 2005. « Distributed Fair Scheduling in a Wireless LAN ». In *IEEE TRANSACTIONS ON MOBILE COMPUTING*. vol.4, n° 6, p. 616-629. IEEE.

Venkataraman, Gayathri, Sabu Emmanuel et Srikanthan Thambipillai. 2007. « Size-restricted cluster formation and cluster maintenance technique for mobile ad hoc networks ». In *INTERNATIONAL JOURNAL OF NETWORK MANAGEMENT*. p. 171-194. John Wiley & Sons, Inc.

Veres, Andras, Andrew T. Campbell, Michael Barry et Li-Hsiang Sun. 2001. « Supporting service differentiation in Wireless packet networks using distributed control ». In *IEEE Journal of Selected Areas in Communications (JSAC)*, vol. 19, n° 10, p. 2081-2093. IEEE

Vishnumurthy, V., T. Sandeep, B. S. Manoj et C. Siva Ram Murthy. 2004. « A novel out-of-band signaling mechanism for enhanced real-time support in tactical ad hoc wireless networks ». In *Proceedings of the 10th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS'04)*. p. 56-63. IEEE.

Vuong, Thai H. P. et Dung T. Huynh. 2000. « Adapting d-hop dominating sets to topology changes in *Ad hoc* networks ». In *Ninth International Conference on Computer Communications and Networks*. p. 348-353. IEEE.

Wan, Peng-Jun, Khaled M. Alzoubi et Ophir Frieder. 2004. « Distributed construction of connected dominating set in wireless ad hoc networks ». In *Mobile Networks and Applications*. vol. 9, n° 2, p. 141-149. Kluwer Academic Publishers.

Wang, Lan et Stephan Olariu. 2004. « A two-zone hybrid routing protocol for mobile *Ad hoc* networks ». In *IEEE Transactions on Parallel and Distributed Systems*. vol.15, n° 12, p. 1105-1116. IEEE.

Wang, Weizhao et Xiang-Yang Li. 2006. « Low-cost routing in selfish and rational wireless ad hoc networks ». In *IEEE TRANSACTIONS ON MOBILE COMPUTING*, vol. 5, n° 5, p. 596-607. USA : IEEE.

Wei, Dali; H. Anthony Chan. 2006. « Clustering *Ad hoc* Networks: Schemes and Classifications». In *3rd Annual IEEE Communications Society on Sensor and Ad hoc Communications and Networks SECON '06,* vol. 3, p. 920-926. USA : IEEE.

Weniger, Kilian. 2005. « Pacman : Passive autoconfiguration for mobile *Ad hoc* networks ». In *IEEE Journal on Selected Areas in Communications*. vol. 23, n° 3, p. 507-519. IEEE.

Wikipédia. 2008. 4G. In Le site de l'encyclopédie libre. En ligne. <http://fr.wikipedia.org/wiki/4G>. Consulté le 13 mai 2008.

Wu, Jie. 2003. « An enhanced approach to determine a small forward node set based on multipoint relays ». In *IEEE 58th Vehicular Technology Conference (VTC 2003)*. (Orlando, 2003), p. 2774-2777. IEEE.

Wu, Jie et H. Li. 2001. « A Dominating-Set-Based Routing Scheme in *Ad hoc* Wireless Networks ». In *Special Issue on Wireless Networks in the Telecommunication Systems Journal*. vol. 3, p. 63-84.

Wu, Jie, Ming Gao et Ivan Stojmenovic. 2002. « On Calculating Power-Aware Connected Dominating Sets for Efficient Routing in *Ad hoc* Wireless Networks ». In *Journal of Communications and Networks*. vol. 4, n° 1, Mar. 2002, p. 59-70.

Wu, Mingyu. 2006. « A Survey of MAC Protocols in *Ad hoc* Networks ». *The University of Texas at Dallas*. En ligne. <http://www.utdallas.edu/~mxw013200/MAC_ADHOC.html>. Consulté le 8 mai 2008.

Xiao, Hannan, Winston K. G. Seah, Anthony Lo et Kee Chaing Chua. 2000. « A Flexible Quality of Service Model for Mobile ad-hoc networks ». In *Vehicular Technology Conference* (VTC 2000). (Tokyo, 15-18 May 2000), vol. 1, p. 445-449. USA : IEEE.

Yang, Hao, Haiyun Luo, Fan Ye, Songwu Lu et Lixia Zhang. 2004. « Security in mobile *Ad hoc* networks: Challenges and solutions ». In *IEEE Wireless Communications*. vol.11, n° 1, p. 38-47. USA : IEEE.

Yu, J. Y. et P. H. J. Chong. 2003. « 3hBAC (3-hop between Adjacent Clusterheads): a Novel Non-overlapping Clustering Algorithm for Mobile *Ad hoc* Networks ». In *Proceedings IEEE Pacrim'03*. (Victoria, 2003), vol. 1, p. 318-321. USA : IEEE.

Yu, J.Y. et P.H.J. Chong. 2005. « A survey of clustering schemes for mobile ad hoc networks ». In *Communications Surveys & Tutorials, IEEE First Quarter*. vol. 7, n° 1, p. 32- 48. USA : IEEE.

Zheng, Qunwei, Xiaoyan Hong et Sibabrata Ray. 2004. « Recent advances in mobility modeling for mobile ad hoc network research ». In *ACM Southeast Regional Conference : Proceedings of the 42nd annual Southeast regional conference*. (Huntsville Alabama, 2004), p. 70-75 , USA : ACM.

Zhou, Hongbo, Lionel M. Ni et Matt W. Mutka. 2003. « Prophet Address Allocation for Large Scale Manets ». In *The Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM 03*. (San Francisco, 30 March-3 April 2003), vol. 2, p. 1304-1311. USA : IEEE.