



**HAL**  
open science

# Utilisation d'une autostructuration pour un routage hiérarchique géographique dans les réseaux sans fil ad hoc

Mandimby Nirina Ranaivo Rakotondravelona

► **To cite this version:**

Mandimby Nirina Ranaivo Rakotondravelona. Utilisation d'une autostructuration pour un routage hiérarchique géographique dans les réseaux sans fil ad hoc. Réseaux et télécommunications [cs.NI]. Université de la Réunion, 2017. Français. NNT : 2017LARE0032 . tel-01879645

**HAL Id: tel-01879645**

**<https://theses.hal.science/tel-01879645>**

Submitted on 24 Sep 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**Thèse de Doctorat de l'Université de la Réunion**

Spécialité

**INFORMATIQUE**

présentée par

**M. Mandimby Nirina RANAIVO RAKOTONDRAVELONA**

pour obtenir le grade de

**DOCTEUR de l'Université de la Réunion**

---

**Utilisation d'une autostructuration pour  
un routage hiérarchique géographique  
dans les réseaux sans fil ad hoc**

---

Soutenue publiquement le 3 juillet 2017 devant le jury composé de

**Pr. Frédéric MESNARD**

**Pr. Véronique VEQUE**

**Pr. Anthony BUSSON**

**Pr. André-Luc BEYLOT**

**Dr. HDR. Tahiry RAZAFINDRALAMBO**

**Dr. HDR. Pascal ANELLI**

**Président**

**Rapporteur**

**Rapporteur**

**Examineur**

**Examineur**

**Directeur de thèse**

Cette thèse a reçu le soutien financier de la Région Réunion et de l'Union Européenne  
(Fonds Européen de Développement Régional - FEDER) dans le cadre du  
Programme Opérationnel de Coopération Territoriale.





*A mes parents*



## Remerciements

Mes premiers remerciements vont à l'endroit de Monsieur Pascal Anelli, Maître de conférences HDR à l'Université de la Réunion, qui a accepté d'encadrer cette thèse. Je le remercie d'avoir mis toute son énergie pour que je puisse mener ces travaux dans les meilleures conditions. Je le remercie également pour la confiance et la patience sans faille qu'il m'a témoignées durant ces années.

Je remercie Monsieur Fanilo Harivelo qui a proposé ce sujet de thèse et sans qui je n'aurais pu réaliser ces travaux. Je lui suis extrêmement reconnaissant pour sa grande générosité en m'ayant accordé tant de son temps libre.

Je tiens à remercier tout particulièrement Madame Véronique Vèque, Professeur à l'Université Paris Sud, et Monsieur Anthony Busson, Professeur à l'Université de Lyon 1 qui m'ont fait l'honneur d'accepter de juger ces travaux et d'en être rapporteurs.

Je remercie également Monsieur Frédéric Mesnard, Professeur à l'Université de la Réunion, d'avoir accepté de présider le jury de thèse. Je tiens également à remercier Monsieur André-Luc Beylot, Professeur à l'ENSEEIH Toulouse ainsi que Monsieur Tahiry Razafindralambo, Maître de conférences HDR à l'Université de la Réunion, d'avoir accepté de faire partie de ce jury.

Je remercie également Monsieur Bruno Baynat, Maître de conférences à l'Université de Paris 6, qui a su donner un nouveau souffle à ces travaux à travers ses conseils lors de son passage.

Je remercie également Madame Isabelle Guérin-Lassous, Professeur à l'Université de Lyon 1, et Monsieur Pierre Ugo Tournoux, Maître de conférences à l'Université de la Réunion, d'avoir accepté de faire partie de mon comité de suivi de thèse.

Je remercie également le Laboratoire d'Informatique et de Mathématiques, dirigé par Monsieur Jean Diatta, Professeur à l'Université de la Réunion, ainsi que l'ensemble des membres, pour l'accueil et les moyens qui m'ont été mis à disposition pour le bon déroulement de cette thèse.

Je remercie également la Région Réunion qui a soutenu financièrement ces travaux de thèse.

Je remercie également mes collègues, mes amis et mes proches pour leurs contributions et leurs encouragements. Merci à la personne qui a fait un travail remarquable de relecture mais qui n'a pas souhaité être citée nommément.

Je tiens à remercier chaleureusement Dada, Neny et Tiana, qui ont toujours été d'un soutien infaillible malgré la distance.

Je remercie toutes les personnes, qui ont contribué de près ou de loin à la réalisation de ces travaux.

Enfin, je remercie mon Dieu. A Lui seul soit la gloire.



## Résumé

La multiplication de l'usage des équipements connectables à un réseau sans fil tels que les smartphones ou les objets connectés peut être mise à profit pour construire des réseaux reposant sur des communications de proximité : les appareils communiquent directement entre eux, sans le support d'une infrastructure centrale. On parle de réseau ad hoc. Ce type de réseau sans infrastructure est exploité dans de nombreux domaines comme le militaire ou la gestion de catastrophes naturelles. Pour certains cas d'utilisation actuels et futurs, le passage à l'échelle, c'est-à-dire, la capacité à fonctionner malgré l'accroissement de la taille du réseau, est une propriété indispensable. Dans cette thèse, nous nous intéressons à la problématique du routage sous l'aspect du passage à l'échelle. Le routage est la fonction responsable de l'acheminement des données au sein du réseau. Pour cela, des informations de contrôle sont échangées entre les équipements connectés au réseau appelés nœuds. Mais le trafic associé à ces échanges peut être un frein pour passer à l'échelle. Afin de le réduire, nous proposons d'agréger les informations concernant des nœuds géographiquement proches. Nous présentons donc un modèle de protocole de routage combinant les approches topologique et géographique : il s'agit, pour un nœud donné, de maintenir des routes menant, d'une part, vers d'autres nœuds dans son voisinage proche et, d'autre part, vers des zones géographiques de plus en plus larges au fur et à mesure de l'éloignement. Dans le but de mettre en œuvre ce modèle, nous proposons une structuration de la zone de déploiement du réseau en régions hiérarchiques. Ces régions servent d'agrégateur de routes à travers un adressage des nœuds selon les régions auxquelles ils appartiennent. Enfin, la mise en œuvre du modèle a donné naissance à un nouveau protocole de routage que nous avons baptisé Hierarchical Geographical Distance Vector protocol (HGDV). Les évaluations montrent une meilleure faculté à monter à grande échelle par rapport aux protocoles de routage ad hoc classique et aussi une capacité à adresser les problèmes propres aux protocoles de routage purement géographique.

### Mots-clés :

Réseaux sans fil, réseaux ad hoc, adressage, routage hiérarchique, routage géographique, agrégation.



# Abstract

The widespread use of wireless devices such as smartphones and connected objects leads to an increasing emergence of infrastructure-less networks relying on device-to-device communication: ad hoc networks. Applications of this type of network vary widely from military operations to emergency situations. Scalability is a key property for some current or future use cases involving a large number of devices, also called nodes. We mean by scalability, the ability of the network to increase in size without drastic loss of performance and with a cost-effective network maintenance. In this thesis, we focus on the scalability from a routing perspective. Routing handles the delivery of data packets from a source to a destination across the network. This is achieved by the exchange of information between nodes. The corresponding traffic may represent a break for scalability in presence of numerous nodes. In this work, we propose an aggregation of routing information. First, we introduce a mixed topological and geographical-based routing protocol model: a node maintains precise routing information to other nodes in close proximity and geographically aggregate information for nodes at greater distances. In order to implement the model, we propose a partition of the deployment area into smaller hierarchical regions based on geographical coordinates. Nodes are assigned hierarchical address depending on the regions they are located in. Regions are the units of nodes aggregation. The further the nodes are, the larger the regions referring to them are. Finally, the implementation of the model led to a new routing protocol called Hierarchical Geographical Distance Vector protocol (HGDV). The results show that our proposition outperforms regular ad hoc routing in terms of scalability and addresses more efficiently geographical routing-related issues compared to classical geographical routing.

**Keywords:**

Wireless networks, ad hoc networks, addressing, hierarchical routing, geographical routing, aggregation.



# Table des matières

<b>Remerciements</b>	<b>v</b>
<b>Résumé</b>	<b>vii</b>
<b>Abstract</b>	<b>ix</b>
<b>Table des matières</b>	<b>xi</b>
<b>1 Introduction générale</b>	<b>1</b>
1.1 Contexte . . . . .	1
1.2 Problématique . . . . .	3
1.3 Contributions et plan . . . . .	4
<b>2 Routage dans les réseaux sans fil ad hoc</b>	<b>7</b>
2.1 Introduction . . . . .	7
2.2 Routage par approche topologique . . . . .	8
2.2.1 Protocole proactif . . . . .	8
2.2.2 Protocole réactif . . . . .	9
2.2.3 Protocole hybride . . . . .	11
2.2.4 Limites . . . . .	11
2.3 Routage hiérarchique . . . . .	12
2.3.1 Principes . . . . .	12
2.3.2 Limites . . . . .	13
2.4 Routage géographique . . . . .	14
2.4.1 Principes . . . . .	14
2.4.2 Limites . . . . .	18
2.5 Autres types de routage . . . . .	18
2.6 Propositions les plus proches de notre protocole . . . . .	19
2.7 Adressage des nœuds . . . . .	19
2.7.1 Autoconfiguration d'adresse avec état . . . . .	20
2.7.2 Autoconfiguration d'adresse sans état . . . . .	21
2.7.3 Hiérarchisation . . . . .	22
2.7.4 Séparation identification/adresse . . . . .	24
2.8 Conclusion . . . . .	24

<b>3</b>	<b>Vers un routage à vecteurs de distance géographique</b>	<b>25</b>
3.1	Introduction	25
3.2	Caractéristiques des réseaux sans fil ad hoc à prendre en compte pour la conception d'un protocole de routage	26
3.3	Cahier des charges du routage dans les réseaux sans fil ad hoc	26
3.4	Modèle de routage à vecteurs de distance géographique	28
3.4.1	Motivations	28
3.4.2	Idée générale	30
3.4.3	Les deux types de zones pour le routage	31
	Zone locale	31
	Zone d'agrégation	32
3.4.4	Hypothèses	33
3.4.5	Structure d'une table de routage	33
3.4.6	Maintenance des tables de routage	35
	Décision d'agrégation	35
	Conventions	36
	Initialisation	36
	Heuristique	36
	Exemple	37
3.4.7	Procédure de routage	38
	Heuristique	38
	Exemple	39
3.4.8	Discussions	39
	Temps de convergence	39
	Taille des tables de routage	40
	Taux d'acheminement de paquets	40
	Mise en œuvre	41
3.5	Conclusion	42
<b>4</b>	<b>Autostructuration par adressage hiérarchique géographique</b>	<b>43</b>
4.1	Introduction	43
4.2	Problématiques et motivations	44
4.3	Partitionnement du plan	46
4.3.1	Partitionnement en régions hiérarchiques	46
4.3.2	Numérotation des régions	47
4.3.3	Algorithme	47
4.3.4	Propriétés	50
4.3.5	Adressage	51
4.4	Adressage hiérarchique géographique avec information de positionnement absolu	51
4.4.1	Hypothèses	51
4.4.2	Acquisition d'adresse	52

4.4.3	Changements topologiques	52
	Mobilité	52
	Partitionnement et fusion	53
4.4.4	Discussions	53
	Non-unicité des adresses	53
	Intégration avec un protocole de routage	53
	Compatibilité avec les applications sur IP	53
4.5	Adressage hiérarchique géographique par information de positionnement relatif	54
4.5.1	Hypothèses	54
4.5.2	Procédure générale d'autoconfiguration	54
4.5.3	Acquisition d'adresse	55
	Nouveau nœud	55
	Devenir agent d'adresse (AA)	55
	Exemple	56
4.5.4	Changements topologiques	57
	Mobilité d'un nœud	57
	Partitionnement	57
	fusion	57
4.5.5	Discussions	58
	Consommation de bande passante	58
	Latence	58
	Faiblesses	59
4.5.6	Simulation	59
	Scénario et métriques	59
	Paramètres	60
	Résultats	60
4.6	Conclusion	62
<b>5</b>	<b>Routage à vecteurs de distance hiérarchique géographique avec zones d'agrégation</b>	<b>63</b>
5.1	Introduction	63
5.2	Problématiques et motivations	63
5.3	Mise en œuvre générale	64
5.4	Hypothèses faites par la suite	65
5.5	Rappel sur le partitionnement et l'adressage des nœuds et des zones d'agrégation	66
	5.5.1 Choix de la taille d'une région élémentaire	66
5.6	Définition des zones à partir du partitionnement du plan	67
	5.6.1 Zone locale et paramètre $k$	67
	5.6.2 Zones d'agrégation et paramètre $m$	68
5.7	Structure des tables	69

5.8	Maintenance des tables . . . . .	71
5.8.1	Principe . . . . .	71
	Gestion de la mobilité . . . . .	71
5.8.2	Contenu des informations de route . . . . .	72
5.8.3	Initialisation . . . . .	73
5.8.4	Algorithme . . . . .	73
5.8.5	Exemple . . . . .	77
5.9	Procédure de routage . . . . .	78
5.9.1	Principe . . . . .	78
5.9.2	Algorithme . . . . .	79
5.9.3	Exemple . . . . .	80
5.10	Simulation . . . . .	81
5.10.1	Objectifs . . . . .	81
5.10.2	Métriques . . . . .	82
5.10.3	Paramètres communs . . . . .	82
5.10.4	Influence de la taille de région élémentaire . . . . .	83
	Taille des tables de routage . . . . .	83
	Trafic de contrôle . . . . .	84
	Taux de succès d'acheminement et optimalité . . . . .	84
	Synthèse . . . . .	85
5.10.5	Influence des paramètres $k$ et $m$ et passage à l'échelle . . . . .	85
	Temps de convergence . . . . .	85
	Taille des tables de routage . . . . .	86
	Trafic de contrôle . . . . .	88
	Taux de succès d'acheminement . . . . .	89
	Optimalité des routes . . . . .	90
	Synthèse . . . . .	91
5.10.6	Comparaison avec d'autres protocoles de routage et mobilité . . . . .	92
	Taille de table de routage . . . . .	93
	Trafic de contrôle . . . . .	93
	Taux de succès d'acheminement . . . . .	95
	Optimalité . . . . .	97
5.11	Conclusion . . . . .	97
<b>6</b>	<b>Conclusion générale</b>	<b>101</b>
	<b>Bibliographie</b>	<b>107</b>
	<b>Table des figures</b>	<b>117</b>

## Chapitre 1

# Introduction générale

### 1.1 Contexte

Ordinateurs, smartphones, capteurs, objets connectés ... la multiplication des appareils connectables à un réseau sans fil nécessite toujours plus d'infrastructures pour les accueillir. Mais ces équipements ont la capacité de communiquer directement : il suffit que deux entités soient à portée l'une de l'autre pour que les échanges puissent se dérouler. L'exploitation de ces communications de proximité avec la possibilité d'utiliser les autres appareils comme relais pour les échanges à grande distance permet de se passer des infrastructures. Dans cette thèse, nous nous intéressons à un type de réseau qui exploite presque exclusivement ce paradigme de communication : les réseaux sans fil *ad hoc*.

Un réseau sans fil *ad hoc* est constitué d'appareils appelés *nœuds*, communiquant directement entre eux ou en utilisant les autres comme relais. Un nœud est à la fois hôte et routeur. Les réseaux sans fil *ad hoc* possèdent des caractéristiques particulières qui les différencient des autres types de réseaux :

- Sans infrastructure : contrairement aux réseaux sans fil classiques, un réseau *ad hoc* ne nécessite aucune infrastructure préexistante, comme par exemple la présence d'un ou plusieurs points d'accès, pour se former. Le réseau se constitue au fur et à mesure que les nœuds s'associent entre eux ;
- Fonctionnement distribué : il n'existe pas d'entité centrale, intérieure ou extérieure, responsable de la mise en place, ou de la maintenance du réseau. Tout repose sur les échanges d'égal à égal entre les nœuds. En d'autres termes, les nœuds sont supposés égaux du point de vue de la responsabilité ;
- Mobilité : les nœuds d'un réseau sans fil *ad hoc* ne sont pas tenus d'occuper un emplacement fixe. La mobilité des nœuds associée au fonctionnement distribué rend ainsi plus complexe la maintenance du réseau ;
- Canal de communication limité : les réseaux sans fil *ad hoc* sont particulièrement sensibles aux problèmes communs des technologies sans fil comme les interférences ou le problème de la station cachée dans lequel, selon une topologie particulière, deux stations ne peuvent pas s'entendre et, pensant que le canal de communication est libre, envoient simultanément des paquets

vers une troisième station, provoquant des collisions. La mobilité des nœuds augmente la probabilité d'apparition d'une telle topologie et donc du phénomène ;

- Énergie limitée : pour être mobiles, les nœuds doivent embarquer des sources d'énergie portables comme les batteries. Ces dernières ont des capacités limitées. Aussi, l'une des contraintes de fonctionnement d'un réseau ad-hoc consiste à minimiser la consommation énergétique.

Selon les importances relatives de ces différentes caractéristiques, nous pouvons distinguer, de manière non-exhaustive, plusieurs types de réseau sans fil ad hoc. Un *Mobile Ad hoc Networks* (MANETs) [1] est, comme son nom l'indique, constitué de nœuds mobiles. Il s'agit d'un type générique qui n'insiste pas sur une caractéristique en particulier. Les réseaux de capteurs ou *Wireless Sensor Networks* (WSNs) [2] sont constitués de nœuds ayant des ressources énergétiques limitées et dont l'objectif est de remonter les informations récoltées par ces nœuds, en les relayant si nécessaire, vers une entité centrale. La gestion de la consommation d'énergie est une priorité dans ce type de réseau. Les *Vehicular Ad Hoc Networks* (VANETs) [3] reposent sur l'utilisation des véhicules routiers en tant que nœuds. La gestion de la mobilité est donc un point essentiel pour l'exploitation de ces réseaux. Poussant la mobilité des nœuds à des cas extrêmes, les *Flying Ad hoc Networks* (FANETs) [4] sont les équivalents aériens des VANETs. Enfin les réseaux mesh [5] sont des réseaux hétérogènes dans le sens où une architecture à deux niveaux est présente : une infrastructure principale multi saut communément appelée *backbone* et des points d'accès connectés à celle-ci, assurant la connectivité des clients finaux du réseau [6].

Les réseaux sans fil ad hoc trouvent des applications dans de nombreux domaines. Nous allons expliciter quelques cas d'utilisation. La non nécessité d'infrastructure et le fonctionnement distribué de ces réseaux facilitent et accélèrent leurs déploiements. Ils constituent ainsi l'une des alternatives les plus intéressantes pour étendre Internet vers une zone dépourvue d'infrastructure. Il s'agit d'ailleurs d'une des principales motivations pour le développement des réseaux mesh [6]. Les réseaux ad hoc sont aussi exploités dans les situations où l'intervention humaine est évitée afin d'améliorer le rendement et la sécurité : les machines communiquent directement entre elles. On parle alors de communication machine à machine ou *machine-to-machine* (M2M). Dans ce domaine, les auteurs dans [6] montrent les bénéfices du multi saut. Parmi les autres applications des réseaux ad hoc, peut citer celles aussi bien militaires [7] que civiles comme par exemple, en tant qu'outil d'aide à la gestion de catastrophes [8]. On retrouve aussi ces réseaux dans des domaines très spécifiques comme l'agriculture [9] ou le spatial [10] par exemple. Enfin, dans [6], les auteurs décrivent les futures domaines d'application où les réseaux ad hoc devraient s'imposer comme la santé, les *smart cities*, les *smart grid* ou l'Internet industriel.

## 1.2 Problématique

Comme nous l'avons vu dans la section précédente, les réseaux sans fil ad hoc présentent de nombreuses caractéristiques spécifiques qui peuvent, les unes comme les autres, faire l'objet d'études approfondies. Mais, à l'heure de l'informatique ubiquitaire et de l'Internet des objets qui impliquent l'interconnexion entre un grand nombre d'éléments, il est important de s'intéresser à l'impact de ce nombre sur le bon fonctionnement des réseaux. Par ailleurs, certaines applications nécessitent l'exploitation de réseaux de grande taille c'est-à-dire, constitués d'un nombre important de nœuds comme dans le cas d'un déploiement massif de capteurs dans la nature ou la constitution d'un réseau à l'échelle d'une ville. On parle de réseaux passant à l'échelle. De manière générale, on peut définir le passage à l'échelle d'un système par sa capacité à supporter l'accroissement de ses paramètres limitants [11]. Dans le domaine des réseaux ad hoc, les paramètres limitants sont ceux dont l'augmentation est susceptible de dégrader les performances du réseau comme sa taille, la mobilité des nœuds ou le trafic de données ... [11]. Ainsi, à titre d'exemple, un réseau qui passe à l'échelle continue de fonctionner sans grande dégradation malgré une croissance en taille, et ce sans augmentation déraisonnable des coûts liés à l'exploitation de celui-ci. Nous ajoutons que la considération du passage à l'échelle est un moyen d'assurer une pérennité des technologies développées pour les réseaux ad hoc. En effet, même s'il n'existe pas encore d'applications grand public qui exploitent à très grande échelle ce type de réseau, il n'en serait pas de même dans le futur [6]. Il est important de développer des technologies qui anticipent les cas d'utilisation de demain. L'idée est analogue à celui d'Internet dont les premiers concepteurs ont développés une technologie passant à l'échelle sans que les usages de l'époque ne le requièrent, cette propriété ayant été indispensable que bien plus tard. Considérant ces éléments, dans cette thèse, nous nous intéressons au passage à l'échelle des réseaux sans fil ad hoc mais du point de vue d'une fonction essentielle : le *routage*.

Le routage unicast est la fonction responsable de l'acheminement des paquets de données entre un nœud source et un nœud destination. Plus précisément, l'*acheminement* consiste en l'aiguillage des données de la source vers la destination à partir de prises de décision effectuées par chaque nœud où les données transitent. Ces décisions se font sur la base d'informations que chaque nœud stocke dans une table qui lui est propre appelée *table de routage*. Les informations dans les tables sont constituées à partir des échanges d'informations de topologie entre les nœuds. Ces échanges suivent des règles précises, définies par le *protocole de routage*. La conception d'un protocole de routage doit tenir compte des propriétés précédentes des réseaux sans fil ad hoc comme l'absence d'infrastructure, la mobilité, etc. mais aussi, dans notre cas, du passage à l'échelle. Pour prendre les décisions relatives à l'acheminement des paquets, un protocole peut maintenir les informations concernant toute la topologie du réseau comme cela est fait de manière classique dans les réseaux filaires. Les

routes obtenues sont alors presque optimales en termes de longueur, c'est-à-dire, se rapprochant du plus court chemin. Mais plus le réseau est grand, plus la quantité d'information à enregistrer et à échanger dans le cadre du protocole est importante n'autorisant pas une montée à très grande échelle [12]. Afin de remédier à cela, une approche reposant sur des informations de nature géographique réduit au maximum les informations retenues et échangées entre les nœuds [13]. Mais les routes peuvent être de moins bonne qualité. Un compromis peut donc potentiellement être trouvé en cherchant un moyen de réduire le volume d'informations échangées dans l'approche classique et/ou en exploitant mieux les informations géographiques à disposition. Par ailleurs, dans [14] il est soutenu que l'auto-organisation par structuration des réseaux ad hoc leur offre de meilleures capacités pour passer à l'échelle. En effet, elle résulte des interactions locales entre les nœuds. Ainsi un changement local a peu d'impact au niveau global, autorisant une meilleure montée à l'échelle.

Notre but est de concevoir un protocole de routage qui offre le compromis évoqué plus haut en s'appuyant à la fois sur des informations de topologie et sur des informations de nature géographique. Pour permettre le passage à l'échelle dans ces conditions, notre proposition se base sur une auto-organisation par structuration des nœuds.

### 1.3 Contributions et plan

La proposition principale de cette thèse repose sur la combinaison d'un routage classique avec un routage géographique. Dans un périmètre local, les mécanismes de base de notre proposition, à savoir les échanges de messages de contrôle, le calcul des routes et l'acheminement des paquets relèvent du routage ad hoc classique. Au-delà de ce périmètre local, l'approche géographique prend le relais et les routes sont choisies selon des critères géographiques. Dans ce dernier cas, les destinations des routes ne sont plus des nœuds mais des zones géographiques correspondant à un nœud ou une agrégation de nœuds. Cette agrégation géographique constitue donc une extension de l'approche classique et un point de convergence avec l'approche géographique. Les destinations des routes sont agrégées sur la base d'informations de nature géographique. Nos travaux montrent que l'agrégation de nature géographique offre un compromis entre des routes de bonne qualité et un volume de trafic de contrôle limité en faisant le pari qu'agréger les informations lointaines a peu d'influence sur la qualité des routes. Ce compromis est la clé d'une solution passant à l'échelle.

Le chapitre 2 dresse un état de l'art du routage dans les réseaux sans fil ad hoc avec des discussions sur les limites des protocoles existants notamment en termes de passage à l'échelle.

Le chapitre 3 introduit notre proposition de combinaison des approches classique et géographique pour le routage dans les réseaux sans fil ad hoc. Il s'agit d'un modèle général de protocole de routage qui fera l'objet d'une mise en œuvre dans la suite. Plus précisément, nous développons des heuristiques pour la réalisation des mécanismes de routage de base à travers l'agrégation géographique des nœuds : traitement et mise à jour des informations de routage, choix des routes pour l'acheminement des paquets de données.

Au chapitre 4, nous proposons un mode d'agrégation basé sur deux principes :

- Un partitionnement hiérarchique de la zone de déploiement du réseau ;
- Un système d'adressage des nœuds qui met en place une structuration hiérarchique géographique de ces derniers.

Fort de ces principes, nous proposons au chapitre 5 une mise en œuvre de notre modèle général de routage combinant les approches classique et géographique à travers l'agrégation des nœuds. Pour cela nous avons développé un protocole de routage appelé HGDV pour *Hierarchical Geographical Distance Vector protocol*. Nous présenterons ensuite les résultats des simulations mesurant les performances obtenues notamment sur le plan du passage à l'échelle et de la mobilité, comparées aux approches exclusivement classiques et géographiques.



## Chapitre 2

# Routage dans les réseaux sans fil ad hoc

### 2.1 Introduction

Les réseaux sans fil ad hoc trouvent des applications dans un large spectre de domaines. Mais pour qu'un réseau soit exploitable, il est nécessaire que certaines fonctionnalités soient préalablement mises en place. Par exemple, un mécanisme d'adressage des nœuds assure l'identification de ceux-ci. Les fonctionnalités développées pour les réseaux locaux sans fil avec point d'accès ne peuvent pas être transposées directement aux réseaux ad hoc. Il est nécessaire de les adapter pour prendre en compte les particularités de ces réseaux, en particulier, leur caractère distribué, la mobilité des entités membres et les communications multi-sauts. Dans ce chapitre, nous allons nous intéresser à une de ces fonctionnalités : le routage. Le routage assure la connectivité du réseau et met en place un système d'aiguillage qui permet à un paquet de trouver son chemin vers sa destination [15]. Ce chemin est constitué de un ou plusieurs intermédiaires. Les chemins sont déduits des informations de routage échangées entre les nœuds. Ces échanges suivent des règles définies par un protocole de routage. Selon la nature de ce dernier, les chemins sont déterminés à partir d'informations topologiques ou géographiques par exemple.

Nous présentons dans ce chapitre un état de l'art des protocoles de routage conçus pour les réseaux sans fil ad hoc. Nous avons choisi de mettre en avant trois grandes familles pour une meilleure mise en perspective avec notre propre proposition : l'approche topologique, l'approche hiérarchique et l'approche géographique. Mais nous évoquerons aussi les propositions reposant sur d'autres approches notamment hybrides. La dernière section de ce chapitre s'intéresse à l'adressage qui constitue un préalable à tout protocole de routage et peut influencer ses performances.

## 2.2 Routage par approche topologique

Les protocoles de routage par approche topologique reposent sur la connaissance de la topologie du réseau [16]. Chaque nœud a une connaissance plus ou moins complète de tous les autres nœuds existants dans le réseau à travers, entre autres, des informations sur l'existence ou non d'un lien vers ces nœuds et les coûts pour y arriver. Typiquement chaque nœud peut savoir comment joindre un autre nœud et en combien de sauts, par exemple. Ces informations peuvent être obtenues sur demande ou être consignées dans une table pour être disponibles à chaque instant. On peut distinguer trois familles de routage par approche topologique : proactif, réactif et hybride. Une étude comparative entre ces différents groupes est proposée dans [12]. Dans la suite, nous allons nous intéresser uniquement à une sélection de protocoles appartenant à chaque famille. Il s'agit des protocoles ayant été à l'origine des principes de base de chaque famille mais aussi ceux qui sont les plus cités dans la littérature.

### 2.2.1 Protocole proactif

Un protocole proactif fait en sorte que les informations de routage soient disponibles à tout moment. Pour cela, les nœuds s'échangent périodiquement des paquets de signalisation dans le but de maintenir les routes à jour. Ces routes sont stockées dans une table de routage. Les paquets de signalisation véhiculent les informations contenues dans cette table. On voit ici un premier inconvénient : les échanges se font même en l'absence de trafic de données. Une partie de la bande passante est donc gâchée sans parler de la consommation en énergie qui en résulte. Par contre, l'avantage est la disponibilité immédiate des routes ce qui réduit les délais. Selon le type d'information stockée dans les tables de routages, on peut distinguer deux types de protocoles proactifs : à vecteurs de distance et à états de liens.

*Destination-Sequenced Distance-Vector Routing (DSDV)* [17] fait partie des premiers protocoles de routages à vecteurs de distance conçus spécifiquement pour les MANETs. Il s'agit d'une adaptation de l'algorithme de Bellman-Ford [17] afin de prendre en compte la mobilité. En effet, cet algorithme a tendance à générer des boucles dans un environnement dynamique. Pour y remédier des numéros de séquences sont associés aux routes. A terme, chaque nœud génère une entrée de type {destination, prochain saut, coût} pour chacun des autres nœuds du réseau dans sa table. Ainsi, dans un réseau connexe, la taille des tables est linéaire avec le nombre de nœuds dans le réseau. Chaque nœud envoie périodiquement des paquets de signalisation véhiculant l'ensemble des entrées dans sa table. Ramené à l'échelle du réseau, le trafic correspondant aux paquets de signalisation est fonction du nombre total de nœuds multiplié par le nombre d'entrées dans les tables qui est lui-même également fonction du nombre total de nœuds dans le réseau. En d'autres termes, le trafic de contrôle augmente avec le carré du nombre total de nœuds dans le réseau. A cause

de cette augmentation exponentielle, la solution ne satisfait pas les critères de passage à l'échelle.

D'autres chercheurs ont choisi d'adapter les protocoles à états de liens aux MANETS. C'est le cas de Chen et al. qui proposent *Global State Routing* (GSR) [18]. Dans les protocoles de routage à états de liens, dès qu'une modification de lien survient, cette information est diffusée dans tout le réseau ce qui ne passe évidemment pas à l'échelle. Dans GSR, les informations de mise à jour de lien sont communiquées uniquement aux voisins à un saut grâce aux envois périodiques de paquets de signalisation. Ainsi, ces informations se propagent de la même manière que dans DSDV, c'est-à-dire, de proche en proche jusqu'à atteindre tous les nœuds du réseau. Étant donné que le nombre de liens dans le réseau augmente avec la taille de celui-ci, les tables de routage augmentent aussi rapidement, rendant difficile le passage à grande échelle. *Fisheye State Routing* (FSR) [19] est une amélioration de GSR faisant en sorte que les envois des paquets de mises à jour des liens proches en termes de nombre de sauts soient plus fréquents que celui des liens plus éloignés. L'objectif est de réduire le trafic de contrôle généré. Malgré tout, la taille des tables de routage reste du même ordre que le nombre de nœuds dans le réseau. *Optimized Link State Routing* (OLSR) réduit encore plus le volume de trafic de contrôle par l'intermédiaire d'une sélection de nœuds spéciaux appelés *relais multipoints* qui deviennent responsables de la dissémination des informations de lien. Ces dernières concernent toujours l'ensemble des liens dans le réseau. Le trafic associé reste donc fortement sensible à l'augmentation du nombre de nœuds dans le réseau. L'objectif principal de ces deux derniers protocoles est de réduire le volume de signalisation par des considérations de distance en termes de nombre de sauts ou en misant sur une organisation hiérarchique des nœuds. Il s'agit d'une des approches utilisées pour permettre le passage à l'échelle. Nous verrons que réduire la table de routage en plus de la signalisation renforce encore plus la capacité de montée à grande échelle.

### 2.2.2 Protocole réactif

Un protocole de routage réactif ne maintient pas les informations de routages à tout instant. Elles sont disponibles sur demande. Ainsi, un nœud souhaitant envoyer un paquet doit formuler une requête pour découvrir la route vers la destination. Cette requête est diffusée dans tout le réseau. Les différences entre les protocoles concernent la façon dont les informations de routes sont exploitées.

Dans *Dynamic Source Routing* (DSR) [20], un nœud cherchant à joindre une destination émet une requête de découverte de route à tous ses voisins à un saut. Cette requête contient l'adresse du nœud cible. Les voisins retransmettent les paquets à leurs propres voisins en ajoutant leurs propres adresses et ainsi de suite jusqu'à ce que le destinataire reçoive la requête. Notons qu'à chaque réception d'une requête,

un nœud enregistre la liste des adresses dans l'entête du paquet. Quand le nœud destinataire reçoit la requête, il émet en unicast une réponse au nœud précédent dans la liste des nœuds intermédiaires qu'il a auparavant enregistré. Chaque nœud intermédiaire fait de même jusqu'à ce que la réponse atteigne la source. Notons que chaque nœud intermédiaire ajoute sa propre adresse avant d'émettre à nouveau la réponse. A la réception d'une réponse à sa requête de découverte de route, le nœud source connaît désormais la route vers la destination par l'intermédiaire de la liste des adresses contenue dans celle-ci. Il peut maintenant envoyer les paquets de données en inscrivant dans l'entête la liste des nœuds intermédiaires par lesquels les paquets doivent transiter. Ainsi plus le réseau est grand, plus l'entête des paquets de données est volumineux réduisant le débit utile du protocole.

*Ad hoc on Demand Distance Vector (AODV)* [21], au contraire de DSR, n'inscrit pas les routes dans les entêtes des paquets de données. AODV s'inspire, d'une part de DSDV en ce qui concerne l'utilisation de tables de routage contenant des vecteurs de distance, et d'autre part, de DSR par l'utilisation des requêtes de découverte de route. Une source  $S$  souhaitant découvrir la route vers une destination  $D$  commence par envoyer une requête contenant l'adresse de cette dernière à tous ses voisins qui les retransmettent à leurs tours et ainsi de suite. A la réception d'une requête, un nœud intermédiaire inscrit dans sa table une route inversée, c'est-à-dire, ayant comme destination  $S$  et comme prochain saut, le nœud par lequel la requête a été reçue. A la réception d'une requête,  $D$  envoie une réponse en unicast à destination de  $S$ . Cette réponse sera routée jusqu'à  $S$  grâce aux routes inversées enregistrées par tous les nœuds concernés par les requêtes. A la réception d'une réponse, un nœud intermédiaire enregistre dans sa table la route ayant comme destination  $D$  et comme prochain saut, le nœud par lequel la réponse a été reçue. A la fin,  $S$  aura une entrée vers  $D$  dans sa table et pourra commencer à envoyer les paquets de données. Le principal inconvénient de cette approche est la diffusion globale des requêtes de découverte de route qui devient trop coûteuse dans un réseau à grande échelle. En outre, dans un contexte de forte mobilité, les routes deviennent invalides rapidement nécessitant l'envoi fréquent de requêtes.

*AODV Backup Routing (AODV-BR)* [22] améliore les performances d'AODV en construisant des routes alternatives en plus des routes principales construites classiquement par AODV. Pour cela, les nœuds profitent du mode « promiscuité » dans les communications sans fil pour enregistrer, dans leurs tables, les routes correspondant aux réponses reçues par leurs voisins mêmes si elles ne les concernent pas. Mais l'utilisation de l'inondation pour les requêtes de découverte de routes limite toujours les possibilités de monter à grande échelle. AODVv2 [23] réduit le nombre d'envois de requêtes de découverte de routes grâce à une fonction d'accumulation de route : une requête génère de multiples routes vers différentes destinations en plus de celle

pour laquelle la requête a été envoyée. Mais recourir à l'envoi par inondation de paquets de requête est toujours nécessaire. Nous précisons que l'envoi par inondation n'est pas désirable dans un réseau de grande taille car elle peut conduire à une saturation du canal de communication. En effet, quand un nœud reçoit un paquet à diffuser globalement, il le transmet directement à ses voisins et ainsi de suite. Plus les nœuds sont nombreux, plus il y a d'envois de paquets simultanément gaspillant beaucoup de bande passante. Les nœuds peuvent recevoir plusieurs duplicatas du même paquet.

### 2.2.3 Protocole hybride

Un protocole hybride combine les aspects proactif et réactif afin de permettre la montée à l'échelle. En effet un protocole proactif génère trop de trafic à grande échelle. L'idée est donc de limiter le mode proactif et de recourir au réactif quand cela est nécessaire. Par exemple, *Zone Routing Protocol (ZRP)* [24] définit deux zones pour chaque nœud :

- Une zone intérieure : les routes vers les nœuds situés dans cette zone sont maintenues proactivement ;
- Une zone extérieure : les routes vers les nœuds situés dans cette zone sont découvertes à la demande.

La frontière de la zone intérieure est définie en termes de nombre de sauts. Patel et al. [24] ont montré que la taille optimale de la zone intérieure correspond à celle faisant en sorte que les trafics de contrôle générés par les modes proactif et réactif soient équivalents. Cette frontière varie en fonction de la mobilité dans le réseau et du volume de données qui y transite. En effet, le trafic généré par le mode proactif a tendance à augmenter avec la mobilité et celui généré par le mode réactif avec le volume de données routées dans le réseau.

### 2.2.4 Limites

En prenant le parti de consigner toutes les routes vers toutes les destinations-nœuds, les protocoles de routage par approche topologique apportent de la précision et, en particulier les protocoles à états de liens, trouvent les chemins optimaux, par exemple correspondant au plus court chemin, mais au prix de plus en plus de trafic de contrôle à mesure que le réseau grandit. Pour les proactifs, les messages d'échanges de routes consomment trop de bande passante quand il y a beaucoup de nœuds. Une piste d'amélioration serait d'adapter la fréquence d'envoi des routes en fonction de la mobilité : moins les nœuds sont mobiles, moins il est nécessaire de mettre à jour les tables. Concernant les réactifs, le recours à l'inondation des requêtes de découverte de route est problématique dans un réseau étendu et surtout mobile. Les protocoles hybrides ont été conçus pour adresser les problèmes de passage à grande échelle. Mais en utilisant les mêmes principes que les proactifs et les réactifs, ils héritent de leurs inconvénients même dans une moindre mesure. Une

amélioration possible consiste à changer de mode de fonctionnement, du proactif au réactif ou inversement, non pas en fonction de zones mais en fonction d'autres critères comme la taille estimée du réseau [25]. Par ailleurs, le seuil optimal de transition vers le mode réactif dépend des caractéristiques globales du réseau comme la taille et locales comme le trafic de données, la mobilité ... [26]. Ainsi il peut changer fréquemment, or le changement de mode peut nécessiter un délai non négligeable avec l'attente de la convergence des tables de routage, par exemple.

La recherche de la propriété de passage à l'échelle a donné lieu à l'émergence d'autres types de routage comme nous le verrons dans la sous-section suivante.

## 2.3 Routage hiérarchique

### 2.3.1 Principes

Nous avons vu que le principal inconvénient en terme de montée à l'échelle d'un protocole par approche topologique vient du fait que l'ensemble du réseau peut potentiellement être concerné par des changements localisés. Par exemple, dans les protocoles proactifs, l'arrivée d'un nœud provoque la propagation d'une nouvelle information de route à travers le réseau et dans les protocoles réactifs, une inondation de requête est nécessaire avant de trouver la route associée. Les protocoles comme ZRP proposent déjà de limiter la zone de propagation d'une information de route à l'intérieur d'une zone définie autour de chaque nœud. Les protocoles hiérarchiques vont encore plus loin. En effet, dans ces protocoles, les nœuds sont organisés en groupes et les échanges entre nœuds d'un même groupe sont cantonnés à l'intérieur du groupe sous la coordination d'un nœud spécial. Ce schéma permet de limiter le volume de trafic de contrôle associé. [27] propose une analyse du volume de trafic de contrôle généré par les routages de type hiérarchique en montrant les avantages de ceux-ci par rapport aux autres protocoles et en indiquant sous quelles conditions ces avantages sont les plus significatifs. Plus précisément, les auteurs montrent, d'une part, que le volume de trafic de contrôle a seulement une croissance polylogarithmique avec le nombre total de nœuds, et d'autre part, que le nombre de niveaux dans la hiérarchie doit augmenter comme le logarithme de la taille du réseau pour en tirer le maximum d'avantages.

Une hiérarchie peut être construite en groupant les nœuds qui sont géographiquement proches entre eux en grappes de nœuds (*clusters*). Les membres d'une grappe élisent un représentant, la tête de grappe, qui coordonne toutes les communications au sein de la grappe. Les nœuds qui ne sont pas têtes de grappe constituent le premier niveau de la hiérarchie et les têtes de grappe forment un niveau supérieur. Les têtes de grappe peuvent se regrouper à leur tour afin de former des super-grappes. Des têtes de super-grappe sont élues et forment un niveau supérieur

et ainsi de suite. *Hierarchical State Routing* (HSR) [28] est un exemple de protocole de routage hiérarchique avec grappes de noeuds à états de liens. Les noeuds normaux n'échangent leurs informations de liens qu'avec les noeuds appartenant aux mêmes grappes. Les têtes de grappe résumant les informations de liens de leurs propres grappes avant de les échanger avec leurs homologues. Ainsi les informations de lien se propagent dans le réseau sans concerner l'ensemble des noeuds réduisant le volume du trafic de contrôle, offrant de meilleures possibilités de passage à l'échelle. [29] passe en revue plusieurs algorithmes permettant la constitution de grappes de noeuds au sein d'un MANET. En général, la proximité entre les noeuds est évaluée en termes de nombre de sauts. Il est aussi possible de retenir d'autres critères par exemple, géographiques ou énergétiques, pour la construction des grappes. Dans [30], le réseau est partitionné globalement en carrés. Chaque carré représente une grappe et l'appartenance à une grappe de noeuds dépend uniquement de la position géographique à travers l'appartenance à un carré. Par ailleurs, il est intéressant de construire et de maintenir les grappes en fonction de l'état d'énergie des noeuds ou de tout le système [16]. Dans [31] et [32], les noeuds deviennent têtes de grappe en fonction de leurs énergies résiduelles et d'autres paramètres. Un noeud rejoint la grappe dont la tête est la plus proche.

D'autres protocoles hiérarchiques organisent les noeuds en arbre. C'est le cas de [33] qui repose sur un adressage hiérarchique des noeuds pour effectuer le routage. Ce type de routage se rapproche de ce qui est utilisé dans l'Internet.

### 2.3.2 Limites

Les protocoles hiérarchiques sont réputés passer à l'échelle. En effet, comme nous l'avons vu, le trafic de contrôle est significativement réduit par rapport au routage à plat. Mais leur principal inconvénient est lié au coût de la construction et de la maintenance de la hiérarchie. D'autant plus que dans un environnement dynamique comme les MANETs, garder la structure hiérarchique est difficile. En effet, de fréquentes reconstructions sont nécessaires pour que la hiérarchisation mise en place reflète la topologie du réseau. Par ailleurs, pour les protocoles orientés grappes de noeuds, à part les coûts liés aux élections des têtes de grappe, un autre inconvénient vient du fait que ces dernières doivent supporter plus de trafic que les autres noeuds et peuvent devenir des goulots d'étranglement. De plus quand elles se déplacent ou même quittent le réseau (scénarios assez fréquents dans les MANETs), de nouvelles élections doivent se faire.

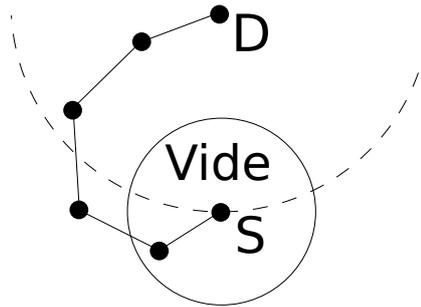


FIGURE 2.1 – Maximum local.

## 2.4 Routage géographique

### 2.4.1 Principes

Les protocoles de routage géographique reposent sur les informations de localisation des nœuds pour prendre les décisions d’acheminement des paquets. Le routage glouton est le plus simple de ces protocoles : une source souhaitant envoyer un paquet choisit comme prochain saut son voisin qui est plus proche géographiquement de la destination qu’elle ne l’est. Chaque nœud exécute cet algorithme jusqu’à ce que le paquet arrive à la destination finale. On voit que pour router un paquet, à condition de connaître la position géographique de la destination, un nœud n’a besoin de connaître que les positions géographiques de ses voisins. De ce fait, l’approche passe à l’échelle car ces informations ne requièrent que des échanges à un saut pour être maintenues. Par contre, que se passe-t-il si aucun voisin n’est plus proche de la destination que la source alors qu’un chemin existe entre la source et la destination ? Un nœud se trouvant dans cette situation est appelé *maximum local* et l’espace entre celui-ci et la destination *vide*. Sur la figure 2.1, le réseau est représenté par un graphe dont les sommets correspondent aux nœuds. Une arête existe entre deux sommets si les nœuds correspondants sont à portée l’un de l’autre. La portée de S est représentée par le cercle dont il est le centre et la distance entre S et D est matérialisée par le demi-cercle hachuré centré en D et passant par S. La source S est donc un maximum local pour un acheminement géographique vers la destination D. Le routage glouton y échoue. Pour trouver tous les chemins existants dans le réseau, les protocoles de routage géographique doivent donc contourner ces vides. De nombreuses techniques ont été proposées dans la littérature. Nous allons en voir quelques unes.

Dans le protocole *Location Aided Routing* (LAR) [34], pour éviter les problèmes de maximum local, le nœud source définit une zone de requête rectangulaire contenant à la fois la source et la destination. Ensuite des requêtes équivalentes à celles utilisées en routage réactif sont envoyées à tous les voisins. Seuls les nœuds appartenant à la zone de requête transmettent les requêtes et ainsi de suite jusqu’à la

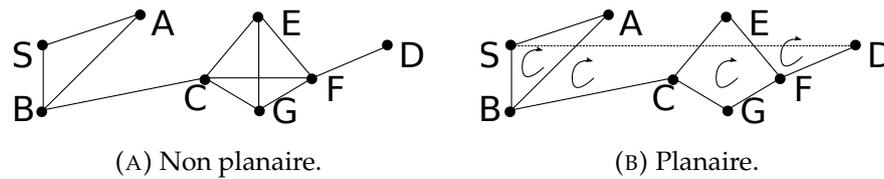


FIGURE 2.2 – Exemple de planarisation et de traversée de face.

destination. Le recours à l'inondation contrôlée réduit le trafic de contrôle comparé aux approches totalement réactives. Mais cela n'est pas suffisant, d'autant plus que la zone de requête peut ne pas être négligeable dans le cas d'un réseau étendu avec sources et destinations à de grandes distances les unes des autres. La capacité à monter à grande échelle est donc limitée.

Sur la figure 2.1 le vide peut être contourné en suivant les arêtes entourant celui-ci jusqu'à arriver à un nœud où le routage glouton fonctionne de nouveau. Les arêtes sont parcourues en choisissant arbitrairement la direction gauche ou droite, selon la règle de la main gauche ou droite, à l'intérieur d'une face. Une face est une surface fermée ou non délimitée par un ensemble d'arêtes. Cette technique de parcours est appelée traversée de face [35]. Mais elle ne fonctionne pas si des arêtes se coupent. Il faut donc rendre le graphe planaire. La planarisation peut se faire de manière distribuée. Sur la figure 2.2a nous avons un graphe de réseau non planaire qui a été planarisé sur la figure 2.2b, les arêtes CF et EG ayant été supprimées. L'application de la technique de traversée de face conduit au parcours des arêtes SABCEFD pour un acheminement de S vers D.

*Greedy Perimeter Stateless Routing* (GPSR) [36] propose de planariser le graphe du réseau, de faire du routage glouton tant que celui-ci fonctionne puis d'exécuter la traversée de face quand un paquet arrive à un maximum local. La limite de GPSR vient du fait que le graphe du réseau est supposé correspondre à un graphe de disque unitaire, c'est-à-dire, que les nœuds ont tous la même portée et que tous les liens sont bidirectionnels. Si ces conditions ne sont pas respectées, la planarisation peut conduire à la perte de la connexité du graphe donc à la non découverte de routes valides.

*Cross-Link Detection Protocol* (CLDP) [37] sonde périodiquement les liens existants pour supprimer les chevauchements de manière distribuée. Dans [38], la proposition améliore CLDP en réduisant le trafic lié aux sondages, en supprimant un sous-ensemble des liens qui se chevauchent. En effet, il y est montré que l'existence de certains croisements de liens n'impacte pas les algorithmes de traversée de face. Par ailleurs, les routes trouvées peuvent ne pas être optimales (en contournant dans la mauvaise direction, par exemple). *Greedy Other Adaptive Face Routing* (GOAFR) [39] propose un algorithme de traversée de face qui résulte en de meilleurs routes

dans le pire des cas par rapport à GPSR.

Dans les MANETs, à cause de la mobilité des nœuds, le graphe du réseau évolue à chaque instant. Aussi la planarisation doit être refaite d'autant plus fréquemment que les nœuds sont mobiles. Et si l'algorithme de planarisation ne converge pas rapidement par rapport à la dynamique du réseau, la connexité du graphe résultant ne sera plus assurée rendant certaines routes invisibles par le routage. Certains travaux ont donc pour objet des techniques de contournement de vide ne recourant pas à la planarisation. C'est le cas de *Geographic Landmark Routing* (GLR) [40], qui propose de découvrir certains nœuds qui se situent à proximité d'un vide et de les enregistrer en tant que borne (*landmark*). La découverte des bornes est effectuée avant l'envoi d'un paquet vers une destination pour laquelle aucune borne n'a été enregistrée. Par la suite, le routage consiste à utiliser les bornes comme destinations intermédiaires par lesquelles il faut passer pour atteindre la destination finale. D'après les auteurs, ce protocole présente comme avantage de meilleures routes par rapport aux solutions avec planarisation. Mais la nécessité d'une découverte préalable des bornes induit des délais dans l'acheminement des données sans parler du trafic de contrôle généré que ce soit de manière proactive ou réactive.

*Greedy Distributed Spanning Tree Routing* (GDSTR) [41] repose sur l'utilisation d'un type particulier d'arbre recouvrant : un arbre d'enveloppe convexe. Chaque nœud parent enregistre les informations géographiques de ses nœuds fils qui se trouvent tous dans une même enveloppe convexe. L'idée est donc, en cas de maximum local, de parcourir l'arbre afin de rejoindre le nœud responsable de l'enveloppe convexe contenant la destination. L'inconvénient de cette approche reste la construction et la maintenance de l'arbre d'enveloppe convexe surtout avec la mobilité des nœuds.

Une autre classe de protocole repose sur un partitionnement du réseau sous forme de grille. Dans [42], le protocole GRID proposé considère chaque carré de la grille comme un *cluster* ou groupe de nœuds. Les nœuds d'un même carré élisent un représentant qui construit la route vers le carré et une passerelle qui est responsable de la communication avec les autres carrés. Le routage s'effectue ensuite de carrés en carrés à travers les représentants. GRID est un protocole réactif adapté de AODV [21]. Contrairement à ce dernier, seuls les représentants de carrés et les passerelles transmettent les requêtes, réduisant ainsi le trafic lié à la recherche des routes. Par contre, la mobilité des représentants peut être problématique. En effet, si un représentant quitte son carré, les nœuds restants doivent en élire un nouveau et les informations de routes enregistrées par le précédent représentant doivent être reconstituées. Le choix de la taille des carrés influence les performances du protocole. Plus les carrés sont petits, plus les nœuds changent de carré dans leur mobilité rendant les informations de routage rapidement obsolètes. Au contraire, plus les carrés

sont larges, moins les informations de routage sont précises allongeant potentiellement les routes. Pour arriver à un compromis, les auteurs de [43] proposent une approche avec plusieurs grilles de différents maillages plus ou moins serrés superposées. Une image de la connectivité de chaque grille est enregistrée dans une table au niveau de chaque nœud. Un nœud souhaitant envoyer un paquet consulte les informations concernant la grille de maillage le plus serré. Si un chemin passant de carré en carré jusqu'à la destination existe, le paquet est envoyé suivant cette route, sinon la grille supérieure de maillage moins dense est consultée et ainsi de suite.

Tous les protocoles de routage géographique que nous avons vus jusqu'ici s'appuient sur les coordonnées géographiques réelles des nœuds. Il est aussi possible de recourir à des coordonnées virtuelles. Typiquement, des nœuds *ancres* sont choisis au sein du réseau manuellement ou aléatoirement. Ces ancres signalent périodiquement leur présence à l'ensemble des nœuds du réseau. Chaque nœud calcule ses coordonnées par rapport aux ancres. Le système de coordonnées a autant de dimensions que de nombres d'ancres et chaque composante correspond à la distance en termes de nombre de sauts à une ancre. Par la suite, le routage glouton est utilisé en utilisant comme fonction de distance celle euclidienne basée sur les coordonnées virtuelles. Notons qu'il n'existe aucun lien entre les coordonnées géographiques réelles et les coordonnées virtuelles. [44] montre que le routage glouton rencontre moins de vide avec ce système. *Beacon Vector Routing* (BVR) [45] et *Hop ID* [46] sont des exemples de protocoles de routage géographique à coordonnées virtuelles. Ces protocoles n'ont pas été conçus pour la mobilité à cause de la difficulté de maintenir des coordonnées justes, en sachant qu'il est nécessaire de compiler les informations provenant de l'ensemble des ancres répartis dans tout le réseau. C'est pourquoi ces protocoles sont plutôt en vogue dans les réseaux de capteurs. *Routing Protocol for Low Power and Lossy Networks* (RPL) [47] est l'un des protocoles les plus aboutis dans ce domaine. Il s'agit d'un type particulier de protocole à coordonnées virtuelles appelé à *gradients*. Les puits du réseau de capteur correspondent aux ancres. Chaque nœud du réseau construit des coordonnées virtuelles basées non seulement sur les distances en termes de nombre de sauts mais aussi en prenant en compte d'autres métriques comme la bande passante offerte par les liens ou le taux de succès d'acheminement au sein du lien. L'idée consiste à transférer les paquets en suivant les gradients de métriques.

Pour conclure sur le routage géographique, nous évoquons brièvement les protocoles prenant en compte la troisième dimension. Ces protocoles sont nécessaires dans le cas d'un déploiement dans un immeuble ou pour les réseaux sous-marines [48]. Les protocoles de routage géographique 3D sont aussi confrontés au problème de maximum local. Les techniques de planarisation ne peuvent être utilisées car la notion de sous-graphe planaire n'a pas de sens en trois dimensions. De plus, [49] montre que même en faisant en sorte que chaque nœud ait une connaissance du voisinage à  $k$  sauts pour  $k$  fixé, cela n'est pas suffisant pour garantir la livraison des

paquets. Aussi, pour garantir la livraison avec un routage glouton, [50] propose une limite inférieure au rayon de transmission des nœuds. Pour plus de détails sur le routage géographique 3D, il convient de consulter [51].

### 2.4.2 Limites

Nous avons vu que les protocoles de routage géographique s'appuient sur la connaissance des positions géographiques du voisinage pour acheminer les paquets. Ces informations sont maintenues à l'aide d'échanges périodiques entre nœuds voisins. A ce propos, dans [52] il est montré que la fréquence d'envoi de ces messages de signalisation doit être adaptée à la mobilité du nœud concerné pour plus d'efficacité en termes de coûts et en termes de pertinences des informations enregistrées. D'autres protocoles ont même fait le choix de ne plus recourir à ces échanges périodiques. Dans [53], ces protocoles sont évalués. Les protocoles de routage géographique fonctionnent en général en mode glouton jusqu'à ce qu'ils soient confrontés aux maximums locaux. Pour contourner les vides, le recours à l'inondation est une solution efficace mais ne passant pas à l'échelle. Les autres techniques comme la planarisation du graphe du réseau et la structuration en arbre sont difficiles dans un contexte de mobilité, le graphe planaire ou la structure d'arbre devenant rapidement obsolète. Notons aussi que la limite de la majorité des protocoles de routage géographique provient du fait qu'ils ne peuvent pas anticiper une voie sans issue causant des allongements dans la longueur des routes et des délais. Pour plus de détails sur le routage géographique, nous renvoyons à [54].

## 2.5 Autres types de routage

Nous avons choisi de mettre en évidence les trois familles de protocoles de routage précédentes afin de mettre notre proposition en perspective par la suite. Il existe évidemment d'autres approches qui ne peuvent être classées dans ces groupes. Certains protocoles s'inspirent du comportement des systèmes biologiques. Dans [55] qui est un protocole réactif, le mode de découverte des routes copie le principe utilisé par les fourmis pour trouver les sources de nourriture et échanger les informations grâce notamment aux phéromones. L'objectif ici n'est pas tant le passage à l'échelle mais plutôt une meilleure gestion de l'énergie. D'autres protocoles reprennent les principes des tables de hachage distribuées comme Tribe [56]. Ce dernier est conçu pour passer à grande échelle mais ne fonctionne que dans les réseaux où les nœuds sont stationnaires ou très peu mobiles. Enfin, certains protocoles interviennent sous la couche IP comme *Lightweight Underlay Network ad hoc Routing* (LUNAR) [57] dans le but d'accélérer la mise en œuvre, le passage à l'échelle n'étant pas un objectif.

## 2.6 Propositions les plus proches de notre protocole

Dans cette section, nous nous intéressons aux protocoles de routage par rapport auxquels notre proposition est la plus proche aussi bien dans l'esprit que dans la mise en œuvre. Dans [58], il est proposé *Hierarchical Distance-Vector Georouting* (HDVG), un protocole s'appuyant sur une construction récursive de grappes dans le réseau avec élection de passerelles entre grappes. Dans cette proposition, les vecteurs de distance sont construits sur la base de considérations géographiques. Mais les tables de routage enregistrent plus d'informations pour suivre avec précision le déplacement des nœuds. Ceci nécessite aussi des vecteurs de mouvement. De plus, ces vecteurs de mouvement introduisent un coût supplémentaire. Dans [59], les auteurs proposent un protocole de routage qui s'appuie sur un partitionnement en *quadtree* du réseau. Mais le protocole est conçu pour faire du multicast géographique ou geocast. Une région issue du partitionnement correspond à un groupe de multicast et joue le rôle de résumé des nœuds qui y sont situés. Enfin, dans [60] (C-Map), le réseau est préalablement partitionné de manière hiérarchique selon un modèle de *quadtree*. Il s'agit d'un partitionnement fixe. Notre proposition propose un mode de partitionnement différent dans le sens où celui-ci n'est pas fixé d'avance. Il est alors à tout moment possible d'étendre la zone partitionnée ou de rendre le découpage plus fin. Toujours dans [60], les informations de routage vers les zones issues du partitionnement sont stockées à l'intérieur de « cartes de connectivité ». Ces dernières renseignent si une zone est accessible et si oui quel est le coût pour la rejoindre. Ce sont donc des vecteurs de distance. Un nœud considère des zones de plus en plus étendues avec l'éloignement, comme dans notre approche. Par contre, C-Map enregistre plus d'informations dans ses tables en considérant plus de zones. En effet, une zone considérée doit toujours être complètement entourée par des zones de même niveau ce qui augmente le trafic de contrôle car, pour maintenir les tables à jour, les paquets échangés contiennent plus d'informations.

## 2.7 Adressage des nœuds

De manière générale, les protocoles de routage requièrent un adressage préalable des nœuds. Le type d'adressage choisi peut avoir un impact sur les performances du routage. Certains protocoles ont même besoin d'une forme d'adressage compatible pour pouvoir fonctionner. Dans cette section, nous nous intéressons aux protocoles existants pour assigner des adresses aux nœuds.

Il existe différentes manières de classer les protocoles d'autoconfiguration d'adresse dans les réseaux ad hoc [61]. Nous avons choisi de présenter deux grandes catégories dans un premier temps : *avec état* et *sans état*. Dans un second temps, nous nous intéresserons aux protocoles hiérarchiques et ceux prônant la séparation entre identifiant et adresse qui ont une philosophie proche de celle de notre proposition.

### 2.7.1 Autoconfiguration d'adresse avec état

Un protocole d'autoconfiguration d'adresse est dit *avec état* quand le réseau, pour rester général, enregistre sous une certaine forme les adresses utilisées par les nœuds à un instant donné. Ces informations peuvent être maintenues de manière distribuée : chaque nœud possède une table enregistrant une partie des adresses déjà utilisées. Dans ce cas, pour s'assurer de l'unicité d'une nouvelle adresse, il est nécessaire de vérifier auprès de tous les nœuds du réseau si celle-ci a déjà été attribuée. Ce mécanisme de vérification s'appelle *Duplicate Address Detection* (DAD). MANET-Conf [62] est un protocole d'autoconfiguration d'adresse avec état et avec DAD. Un nouveau nœud envoie en diffusion un message pour se signaler à ses voisins. A la réception de ce message, un nœud qui est proche et qui possède déjà une adresse valide devient responsable de l'attribution d'une adresse à ce nouvel arrivant, il est désigné comme *initiateur*. Pour cela, il choisit une adresse qui n'est pas présente dans sa table. Il enclenche ensuite un DAD à travers l'inondation d'une requête contenant cette adresse choisie. Si un nœud possède déjà l'adresse dans sa table, il le signale. L'initiateur choisit alors une autre adresse et un DAD est de nouveau effectué. Si aucun nœud ne se manifeste, l'adresse choisie est attribuée au nouveau nœud. Nous constatons tout de suite la difficulté de cette solution à monter à grande échelle à cause de l'inondation de requêtes dans tout le réseau à chaque arrivée d'un nœud.

Pour s'assurer de l'unicité des adresses attribuées, au lieu d'enregistrer l'ensemble des adresses utilisées dans le réseau, il est possible d'allouer à chaque nœud une plage d'adresses. Un nœud n'est autorisé à attribuer que les adresses contenues dans cette plage. Les plages sont distribuées de telle sorte qu'elles sont disjointes entre les nœuds. Dans [63], les auteurs proposent *Distributed Dynamic Host Configuration Protocol* (D2HCP). Un nouveau nœud, en plus de se voir attribuer une adresse, reçoit aussi une plage d'adresses correspondant à la moitié de celle de son initiateur. L'avantage de cette approche est la suppression du DAD qui réduit la consommation en bande passante du protocole et accélère la procédure d'affectation des adresses. Mais à cause de la mobilité des nœuds provoquant les phénomènes de fusion (*merging*), les plages peuvent ne plus être disjointes. La fusion permet à deux groupes initialement isolés de nœuds forment, par la suite, un seul groupe. *Enhanced Distributed Dynamic Host Configuration Protocol* (E-D2HCP) [64] est une extension de D2HCP qui utilise des identifiants de réseau et permet de détecter une fusion grâce à des échanges périodiques de paquets de signalisation contenant ces identifiants.

Une autre approche consiste à utiliser les propriétés de certains objets mathématiques pour générer et maintenir les adresses. Dans Prophet [65], l'état des adresses est représenté par une fonction notons  $f(n)$ ,  $n$  étant une valeur d'état.  $f(n)$  génère un nombre à partir d'une valeur d'état. Le nombre généré est utilisé comme adresse de nœud.  $f(n)$  est conçue de telle manière que la probabilité de générer deux nombres

identiques à partir de deux valeurs d'état différentes est très faible.

Chaque nouveau nœud reçoit en plus d'une adresse une nouvelle valeur d'état qu'il utilisera pour configurer les prochains nœuds. Cette approche ne garantit pas l'unicité des adresses obtenues à cause de la probabilité non nulle de générer deux adresses identiques. Dans le même esprit, Prime DHCP [66] exploite une propriété des entiers positifs qui leur permet d'être factorisés en une suite unique de nombres premiers pour affecter des adresses uniques. En résumé, un nœud assigne une adresse à un nouveau nœud en multipliant sa propre adresse par une suite de nombres premiers. La dynamique d'un réseau ad hoc peut être problématique pour ce protocole. En effet, quand un nœud quitte le réseau, son adresse et toutes celles qu'il aurait dû assigner ne peuvent plus être utilisées et sont donc gaspillées.

Dans [67], il est proposé un protocole d'autoconfiguration d'adresse IPv6 avec état. Le concept d'initiateur est repris. La différence par rapport aux autres protocoles se situe au niveau de la maintenance des adresses utilisées. Ici, seul un nœud appelé *gateway* est responsable de l'attribution des adresses. Si un nouveau nœud se trouve trop loin de la gateway pour communiquer directement avec elle, les autres nœuds déjà configurés servent d'intermédiaires. Ce protocole a été conçu spécialement pour 6LowPan [68] d'où l'utilisation du gateway, au sens de 6LowPan, en tant qu'initiateur principal. L'inconvénient de ce protocole est son incompatibilité avec d'autres types de réseau.

### 2.7.2 Autoconfiguration d'adresse sans état

Dans l'autoconfiguration d'adresse *sans état* aucune information n'est conservée concernant les adresses déjà attribuées. Généralement, un nouveau nœud sélectionne une adresse aléatoire et vérifie son unicité en effectuant un DAD à l'échelle du réseau. L'un des premiers protocoles reposant sur ce principe a été proposé par Perkins et al. [69]. L'obligation de faire un DAD à l'échelle du réseau rend impossible la montée à grande échelle.

*Automatic IP Address Configuration* (AIPAC) [70] utilise le concept d'initiateur. Un initiateur est un nœud déjà présent dans le réseau dont le rôle est de configurer un nouveau nœud. Ainsi, c'est l'initiateur qui sélectionne une adresse aléatoire et se charge de sa vérification. Mais surtout, AIPAC propose des mécanismes pour la détection d'adresses dupliquées et la gestion des fusions. D'abord, AIPAC propose de corriger les conflits d'adresse de manière réactive, c'est-à-dire, uniquement quand les nœuds concernés sont sollicités par le protocole de routage, le but étant de réduire le trafic lié à la maintenance des adresses. Ensuite, pour détecter une fusion, une partie d'une adresse est réservée pour l'identification du réseau correspondant à un groupe connexe de nœuds. Les incohérences au niveau des adresses

sont alors traitées de manière graduelle : un nœud décide de changer de réseau et donc d'adresse selon un seuil lié au rapport entre le nombre de voisins appartenant à l'autre réseau et le nombre de ceux appartenant à son réseau originel. L'objectif est d'éviter une explosion de trafic. Mais l'inondation des requêtes pour l'acquisition de nouvelles adresses reste toujours problématique.

L'auteur de [71] propose la même approche mais adaptée à la configuration d'adresses IPv6. Dérivée de la fonction d'autoconfiguration d'adresse IPv6 standardisée, cette proposition l'étend aux MANETs. Pour ce faire, il est proposé une organisation du réseau en étendues (*scopes*). Une étendue est un groupe de nœuds à un nombre limité de sauts les uns des autres. Pour générer une adresse de lien local, chaque nœud s'appuie sur son adresse MAC et vérifie l'unicité à l'échelle de son étendue locale uniquement. Cette adresse est valide uniquement au sein de l'étendue locale. Par la suite, pour chaque étendue, un nœud représentant est élu. Chaque nœud représentant choisit une adresse de site aléatoire et effectue un DAD à l'échelle du réseau en échangeant uniquement avec ses homologues. Une fois le DAD effectué avec succès, le représentant communique l'adresse de site validée à tous les nœuds de son étendue.

Dans [72], les auteurs proposent AROD pour *Address Reservation and Optimistic Duplicated address detection*. Il s'agit d'une optimisation des approches précédentes dans le but de réduire à la fois les délais de configuration et le trafic de contrôle. Pour cela, un nœud est autorisé à effectuer un DAD pour plusieurs adresses afin d'avoir des adresses de réserve. Quand un nouveau nœud arrive, il peut être configuré avec une adresse de réserve sans DAD réduisant donc le temps de configuration. La fréquence des DAD est donc réduite par rapport aux approches classiques permettant des économies de bande passante. Mais les DAD requièrent toujours une inondation du réseau.

### 2.7.3 Hiérarchisation

La hiérarchisation constitue une des approches traditionnelles dans les réseaux pour faire face à la problématique du passage à l'échelle. Sans surprise, cette approche a été explorée pour la conception des protocoles d'autoconfiguration d'adresse. Nous distinguons deux manières d'introduire la hiérarchie : à travers le mécanisme d'acquisition d'adresse ou à travers des adresses de nature hiérarchique.

Dans le premier cas, une organisation hiérarchique des nœuds est mise en place sans que les adresses aient par la suite une structure hiérarchique. *Logical Hierarchical Addressing* (LHA) [73] définit une plage d'adresses configurables à l'échelle du réseau. Cette plage est ensuite divisée en plusieurs blocs disjoints. Chaque nœud configuré du réseau est responsable d'un bloc. Chaque nœud possède une table de

hiérarchie contenant, entre autres, l'adresse de son *prédécesseur* dans la hiérarchie, c'est-à-dire, le nœud par lequel il a obtenu son adresse et la liste des adresses affectées ou pas encore de ses *successeurs* qui sont les nœuds qu'il a configurés ou configurera. Cette table est constituée en même temps que l'acquisition d'une adresse en se basant sur les informations transmises par le prédécesseur. Elle est mise à jour à chaque fois qu'un nœud configure un nouveau successeur. Ce protocole permet la distribution à grande échelle des blocs d'adresses en minimisant les risques de collisions d'adresses. Un autre avantage est le délai de configuration court. Par contre, la maintenance de la hiérarchie en cas de fusion nous semble peu pratique. En effet, cela implique la reconstruction de la hiérarchie avec redéfinition des blocs affectés. Dans [74], avant toute configuration d'adresse, une structure en arbre est d'abord mise en place à partir d'adresses temporaires reposant sur les adresses MAC, par exemple. Par la suite, la racine configure son adresse puis celles de ses fils. Ces derniers, à leur tour, attribuent des adresses à leurs propres fils et ainsi de suite. A part son adresse, un nœud retient d'autres informations utilisées pour définir à partir de sa propre adresse, l'adresse d'un nouveau nœud qu'il prend en charge. La hiérarchisation par construction est juste une forme de protocole d'adressage à état qui propose un moyen plus efficace de créer des plages d'adresses disjointes. Nous allons voir que la hiérarchisation des adresses offre davantage d'intérêts.

Les protocoles de routage hiérarchique qui sont, rappelons-le, plébiscités pour les réseaux de grande taille, bénéficient de l'utilisation d'adresses hiérarchiques. Ainsi certains protocoles d'adressage proposent un plan d'adressage hiérarchique conçu spécifiquement pour un protocole de routage particulier. Dans [75], les nœuds s'organisent en grappes (*clusters*) de plusieurs niveaux. Une tête de grappe est élue pour chaque grappe. L'adresse d'un nœud est la concaténation de l'adresse des têtes de grappe de toutes les grappes auxquelles celui-ci appartient. La taille des adresses peut donc être problématique selon la profondeur de la hiérarchie de grappes. Dans [33], un adressage hiérarchique sans grappe est proposé. Selon cette proposition, une adresse est constituée de préfixes. Chaque préfixe représente un niveau dans la hiérarchie. Ainsi deux adresses partageant un même préfixe appartiennent à un même niveau. Un nouveau nœud obtient une adresse liée à celle d'un voisin déjà configuré. Plus précisément, il s'agit d'une adresse partageant un certain nombre de préfixes à partir du plus haut niveau avec celle du voisin. L'objectif est de faire en sorte que les adresses des nœuds proches géographiquement soient aussi proches dans la hiérarchie. L'adressage dans cette proposition s'effectue de haut en bas dans la hiérarchie. Ainsi, les adresses sont de plus en plus longues en descendant dans la hiérarchie. Or la longueur des adresses est limitée. Il existe donc une granularité limite correspondant à la longueur maximum que peut avoir les adresses. Dans le cas d'un réseau dense, certains nœuds peuvent potentiellement ne pas trouver une adresse disponible. Par ailleurs, la maintenance de la hiérarchie en présence de mobilité peut générer d'important trafic de contrôle étant donné que les adresses

doivent être fréquemment mises à jour et que l'acquisition d'une adresse implique des échanges avec les nœuds voisins.

#### 2.7.4 Séparation identification/adresse

Avant de clore cette section sur la configuration d'adresses dans les réseaux ad hoc, nous souhaitons évoquer les travaux d'Eriksson et al. [33]. Selon les auteurs, l'un des freins à l'exploitation des MANETs à grande échelle concerne l'utilisation des identifiants invariables des nœuds en tant qu'adresses. En effet, avec la mobilité, les adresses perdent leurs sens topologiques et le suivi des nœuds qui peut être nécessaire pour le routage est impossible. Ils montrent qu'il est plus intéressant d'attribuer des adresses dynamiques aux nœuds. Une adresse dynamique est amenée à changer au cours de la vie d'un nœud pour mieux refléter sa position courante dans le réseau. Ainsi il est nécessaire de distinguer l'identifiant inchangé d'un nœud et son adresse variable. Notre proposition s'inscrit pleinement dans ce paradigme.

## 2.8 Conclusion

Nous avons présenté dans ce chapitre un état de l'art du routage dans les réseaux sans fil ad hoc ainsi que l'adressage. Nous avons mis en relief trois approches pour le routage à savoir l'approche topologique, hiérarchique et géographique. Nous avons aussi passé en revue des protocoles hybrides qui empruntent à plusieurs approches. Nous avons évoqués leurs limites en particulier en termes de passage à l'échelle. Pour adresser cette problématique, nous proposons dans cette thèse un protocole hybride qui s'appuient sur des principes de l'approche topologique au travers de l'utilisation de vecteurs de distance, de l'approche hiérarchique par l'utilisation d'un partitionnement hiérarchique de la zone de déploiement des nœuds et de l'approche géographique en prenant en compte les positions géographiques des nœuds. Le prochain chapitre s'attèle à la description du modèle général de protocole de routage que nous avons conçu.

## Chapitre 3

# Vers un routage à vecteurs de distance géographique

### 3.1 Introduction

Les deux précédents chapitres de cette thèse ont été l'occasion d'évoquer, entre autres, la problématique du passage à l'échelle dans les réseaux ad hoc. Par ailleurs, un état de l'art sur le routage a été présenté avec les limites des protocoles existants en termes de passage à l'échelle. Nous avons ainsi vu que les protocoles de routage par approche topologique, en s'appuyant sur des informations topologiques plus ou moins complètes du réseau, montent difficilement à grande échelle. Les protocoles hiérarchique et géographique ont été conçus pour introduire cette propriété de passage à l'échelle. Mais d'autres difficultés se présentent comme la maintenance de la hiérarchie pour le premier et le problème des maximum locaux pour le second. Nous avons choisi un protocole hybride, empruntant à la fois à l'approche topologique pour l'utilisation de vecteurs de distance, l'approche hiérarchique au travers d'un partitionnement hiérarchique de la zone de déploiement du réseau et l'approche géographique en ce qui est de l'exploitation des positions géographiques des nœuds. L'idée est de profiter des bénéfices de chaque approche en aboutissant à un compromis. Dans ce chapitre nous proposons le modèle de protocole de routage que nous avons conçu et qui sera mise en œuvre dans un autre chapitre de cette thèse à travers un outil qui sera aussi décrit plus loin.

Ce chapitre est organisé comme suit. Dans un premier temps, nous rappellerons les caractéristiques des réseaux sans fil ad hoc et les défis à relever pour avoir un routage efficace. Par la suite, nous nous intéresserons plus particulièrement à l'aspect du passage à l'échelle ou comment garder des performances correctes dans des réseaux de plus en plus grands, c'est-à-dire, constitués de plus en plus de nœuds, tout en tenant compte de l'espace de déploiement. Enfin, nous expliciterons les briques de base de notre nouvelle approche pour le routage ainsi que les heuristiques pour sa mise en œuvre. Il s'agit de la première contribution de cette thèse.

### 3.2 Caractéristiques des réseaux sans fil ad hoc à prendre en compte pour la conception d'un protocole de routage

Les réseaux sans fil ad hoc sont constitués de plusieurs appareils sans fil qui communiquent entre eux selon des caractéristiques qui en font leur singularité [6] [48]. Dans le contexte de la connectivité et en ce qui concerne l'élaboration d'un protocole de routage en particulier, il faut citer :

- Absence d'infrastructure centrale : il n'existe aucune entité gérant la mise en place et la maintenance du réseau à un niveau global. Un protocole de routage ne peut donc pas s'appuyer sur une telle structure pour son fonctionnement. Cette caractéristique a pour conséquence les propriétés suivantes :
  - Fonctionnement distribué : un protocole de routage pour les réseaux ad hoc doit pouvoir fonctionner de manière totalement distribuée que ce soit sur le plan de l'échange d'information ou de la prise de décision ;
  - Tous les nœuds sont égaux : aucun nœud ne dispose de prérogative particulière au regard de sa participation au fonctionnement du réseau, du moins au début. Nous verrons que certains protocoles proposent d'attribuer des rôles spécifiques à certains groupes de nœuds.
- Communication multi-sauts : les nœuds ont des portées de communication limitées. Pour qu'un nœud puisse contacter un autre qui se trouve en dehors de son rayon de communication, il doit pouvoir s'appuyer sur les autres nœuds qui jouent le rôle de relais. Les nœuds sont à la fois hôte et routeur. Un protocole de routage doit tenir compte de cet aspect en particulier concernant la diffusion des informations qui sont nécessaires à son bon fonctionnement.
- Mobilité : les nœuds sont libres de leur mouvement occasionnant des modifications dans la topologie du réseau. Les protocoles de routage dans les réseaux sans fil ad hoc doivent gérer ces changements plus ou moins fréquents de topologie.

Nous avons vu que la conception d'un protocole de routage dans les réseaux ad hoc nécessite la prise en compte de certaines caractéristiques particulières à ce type de réseau. Nous allons maintenant nous intéresser aux problématiques soulevées par ces caractéristiques.

### 3.3 Cahier des charges du routage dans les réseaux sans fil ad hoc

Certaines futures applications exploitant les réseaux sans fil ad hoc impliquent l'utilisation de réseaux de grande taille, c'est-à-dire, avec un grand nombre d'éléments [48] [6]. On peut citer, par exemple ceux assurant la communication entre les objets connectés en mouvement déployés à l'échelle d'une ville. Les réseaux sont tenus d'offrir des services fonctionnels quelque soit leur échelle. Nous rappelons

ici la définition du passage à l'échelle d'un système qui est sa capacité à supporter l'accroissement de ses paramètres limitants [11]. Dans notre cas, nous considérons comme paramètre limitant la taille du réseau sans fil ad hoc. Ainsi le réseau passe à l'échelle s'il supporte un accroissement de sa taille, en d'autres termes s'il continue à être exploitable. Par extension, une fonction du réseau est considérée comme passant à l'échelle si l'augmentation du nombre de nœuds n'entame pas l'efficacité de celle-ci et s'accompagne d'un accroissement raisonnable des coûts associés à sa maintenance. Ainsi, un protocole de routage passant à l'échelle arrive, d'une part, à proposer des routes de bonne qualité malgré l'augmentation de la taille du réseau. D'autre part, le coût de la maintenance, par exemple, celui correspondant au trafic généré par les échanges des informations de routage, doit augmenter raisonnablement. Dans notre cas, l'objectif étant un compromis entre approche classique et approche géographique, notre proposition doit se rapprocher du second en termes de coût et du premier en termes de fiabilité malgré l'augmentation de la taille du réseau.

Un élément important à prendre en compte lors de la conception d'un protocole de routage passant à l'échelle est le caractère dynamique des réseaux sans fil ad hoc. En effet des mécanismes qui passent à l'échelle dans un environnement statique peuvent ne pas l'être en présence de mobilité. Par exemple, la mise en place d'une hiérarchie est très efficace dans un réseau statique comme l'Internet, mais l'est moins avec la mobilité car il est alors nécessaire de reconstruire souvent la structure hiérarchique. La plupart des protocoles de routages existants adressent la problématique de passage à l'échelle dans un environnement dynamique [16] [76] selon différentes méthodes [75] que nous étudierons dans la section suivante.

Nous pensons que les protocoles de routage doivent mettre les nœuds sur un même pied d'égalité, c'est-à-dire qu'ils ne doivent pas assigner des responsabilités particulières à certains nœuds. Les avantages sont un risque de goulots d'étranglement réduit mais aussi une consommation d'énergie plus uniforme entre les nœuds, de la robustesse parce qu'il n'y a pas d'élection et la maintenance et les délais sont de ce fait réduits. Notons que cette approche n'est pas incompatible avec les principes d'organisation des nœuds notamment de manière hiérarchique. Il est tout à fait possible de mettre en place une structuration hiérarchique sans élection et sans maintenance, en s'appuyant par exemple sur des informations de nature géographiques comme nous le verrons plus loin.

Pour terminer, nous soulignons une propriété qui prend tout son sens dans les réseaux sans fil ad hoc : l'auto-organisation. L'auto-organisation peut être définie par l'émergence d'une propriété globale à partir d'interactions locales [77]. Par exemple, le fait qu'un paquet est acheminé vers une destination grâce aux interactions locales entre les nœuds est le signe d'une auto-organisation. Une définition plus restrictive de l'auto-organisation correspond à la mise en place d'une structuration à partir

d'interactions locales [14]. Dans [77] et [14], les auteurs montrent l'intérêt de la mise en place d'une telle structure qui sera exploitée par les fonctions du réseau comme le routage. Il s'agit de faire correspondre la structure topologique à plat d'un réseau avec une structure logique plus organisée, par exemple hiérarchique, de telle sorte que des changements à des niveaux locaux n'aient pas d'impact à un niveau global. Cela apporte donc de la stabilité malgré une topologie changeante et concourt avec les autres propriétés pour assurer le passage à l'échelle. Ainsi il peut être intéressant de mettre en place une telle structuration de manière totalement décentralisée avant d'effectuer le routage.

### 3.4 Modèle de routage à vecteurs de distance géographique

Dans cette section nous allons poser les principes de notre proposition de routage à vecteurs de distance géographique par zones d'agrégation.

#### 3.4.1 Motivations

Le passage à l'échelle est une propriété indispensable dans les réseaux ad hoc afin de permettre la croissance du réseau et il est nécessaire à certaines applications [48]. Étant donné que le principal rôle du réseau est d'assurer la communication entre n'importe quelle source et destination et que le transfert des messages de communication se fait, entre autres, grâce à la mise en œuvre d'un protocole de routage, ce dernier doit aussi supporter le passage à l'échelle.

La section précédente a dressé un bilan des principales familles de protocoles utilisées dans les réseaux ad hoc. Nous avons vu que le routage par approche topologique repose sur une connaissance plus ou moins complète de l'ensemble du réseau au niveau de chaque nœud. Dans l'approche proactive, chaque nœud stocke ces informations de routage dans des tables. Les nœuds s'échangent les contenus de leurs tables pour garder des informations à jour. Plus le réseau est grand, plus il y a d'entrées dans les tables, plus les échanges correspondantes consomment de la bande passante d'où l'impossibilité de monter à grande échelle. En effet, une trop grande part de la bande passante du réseau serait allouée à la maintenance des informations de routage. Pour réduire le volume de ce trafic de contrôle, une possibilité consiste à réduire la taille des tables. Comment ? En trouvant un moyen d'agréger les informations qui s'y trouvent. L'agrégation s'accompagne évidemment d'une perte d'information ayant potentiellement des conséquences sur les performances du routage. Un compromis est donc à trouver entre l'économie en termes de trafic de contrôle et les pertes en termes de performances. La question est maintenant de trouver la nature de l'agrégation à mettre en place. Dans la littérature, le concept de grappe (*cluster*) [29] est une réponse à la problématique de regroupement des nœuds. L'idée est d'agréger les informations concernant les nœuds issus d'une

même grappe. Ainsi d'un point de vue extérieur, les nœuds appartenant à une même grappe forment une seule entité et forment une seule destination au sens du routage. En d'autres termes, les informations de routage correspondant aux nœuds sont agrégées en celles correspondant à la grappe. Mais les grappes sont construites et maintenues à travers les échanges entre les nœuds [29] responsables donc d'une partie du trafic de contrôle. Nous pensons que pour réduire encore plus efficacement ce trafic, les grappes doivent se mettre en place même en l'absence d'échange. Généralement, les algorithmes de mise en grappe (*clustering*) finissent par regrouper les nœuds qui sont géographiquement proches. Une solution est donc de « créer » les grappes uniquement suivant des critères géographiques. Dans un premier temps, l'espace de déploiement du réseau est partitionné en zones qui vont correspondre aux grappes. Ensuite, il s'agit pour les nœuds d'identifier la zone dans laquelle ils se trouvent en se servant de leurs coordonnées géographiques comme proposé par [30]. Enfin, les informations concernant les nœuds appartenant à une même zone sont « résumées » en celles correspondant à la zone concernée. Cette idée a été utilisée dans [78] pour la conception d'un service de localisation. On a donc une agrégation des informations de routage correspondant à des groupes de nœuds, la constitution des groupes n'impliquant pas d'échanges préalables entre les nœuds mais reposant sur des critères géographiques.

Les protocoles de routage géographique réduisent drastiquement le trafic de contrôle en limitant le contenu des paquets de signalisation aux simples informations d'identification et de localisation du nœud qui envoie. Mais le routage basé uniquement sur les informations géographiques est confronté au problème des maximum locaux. Il s'agit, rappelons-le, du cas où aucune progression géographique vers la destination n'est possible, étant donné qu'aucun voisin n'est plus proche de la destination que la source. De nombreuses solutions s'appuyant, entre autres, sur la planarisation et la construction de structures d'arbre recouvrant existent. Pour être exploitable dans un contexte de mobilité, ces structures doivent être reconstruites fréquemment pour refléter les changements au niveau topologique. Par ailleurs, nous avons vu qu'en effectuant le routage glouton et en n'adoptant une stratégie de contournement de vide qu'au niveau d'un maximum local, le routage géographique allonge les routes et les délais. Détecter en avance l'existence d'une voie sans issue permet de réagir en amont. Dans ce cas, il est nécessaire d'avoir à disposition des informations sur l'existence de nœuds dans une zone au-delà de la proximité immédiate. Cela implique des échanges d'autant plus coûteux que le réseau est grand. Comment alors permettre le passage à l'échelle ? En agrégeant également les informations. Puisque le routage géographique s'appuie sur les informations de localisation des nœuds, il est naturel que les informations agrégées soient aussi de nature géographique. Nous arrivons donc à la même proposition du paragraphe précédent, à savoir, une agrégation des informations sur la base de la localisation géographique des nœuds.

Au vu des éléments précédents, nous faisons l'hypothèse qu'agrèger les informations concernant des groupes de nœuds proches permet d'une part, de réduire les informations contenues dans les tables de routage dans le but d'économiser en terme de trafic de contrôle et gagner en terme de passage à l'échelle et, d'autre part, de permettre aux protocoles de routage géographique d'avoir une connaissance plus étendue de la topologie. Cette connaissance étendue vise à détecter en amont les voies sans issues afin de garantir la quantité et la qualité des routes trouvées. Pour vérifier cette hypothèse, nous proposons un protocole de routage qui maintient, pour chaque nœud, les informations de lien vers les nœuds à proximité et les informations sous forme agrégée vers les autres nœuds plus éloignés.

Par ailleurs nous avons évoqué l'importance de l'auto-organisation par structuration du réseau afin d'établir une certaine stabilité [14]. L'approche développée ici s'inscrit pleinement dans cette idée. En effet, l'agrégation sur la base géographique permet de structurer le réseau en zones. La stabilité se manifeste par la mobilité non apparente d'un nœud tant que celui-ci reste à l'intérieur d'une zone.

### 3.4.2 Idée générale

Nous faisons le constat que pour les destinations éloignées, il n'est pas nécessaire d'avoir à disposition des informations de routage précises. Prenons l'exemple d'un voyageur qui souhaite rejoindre une chambre d'hôtel qu'il a réservé dans un autre pays. Dans un premier temps, il doit rejoindre le pays de destination, ce qui ne requiert pas beaucoup de détails, une vue d'ensemble du continent où se trouve le pays peut suffire. Arrivé dans le pays, il doit maintenant trouver la ville où se trouve son hôtel en utilisant par exemple une carte du pays. Plus il se rapproche de la destination, plus il a besoin d'informations détaillées. Et ainsi de suite jusqu'à arriver dans la ville de destination où il aura besoin d'une carte précise pour trouver le quartier et la rue où se trouve son hôtel. Par analogie, le routage a besoin de plus en plus de détails à mesure que la destination est proche. A partir de ce principe, nous proposons une approche qui donne plus de précisions aux informations de routage locales tout en agrégeant celles se rapportant aux destinations éloignées. Pour un nœud donné, une liste des nœuds voisins est maintenue dans la table de routage en tant qu'hôtes joignables. Les autres nœuds sont agrégés en zones de plus en plus étendues à mesure qu'ils sont éloignés du nœud considéré.

Notre proposition repose sur les mécanismes suivants :

- L'agrégation se fait sur la base de la localisation des nœuds. Nous divisons l'étendue géographique où le réseau est déployé en plusieurs zones d'agrégation que nous allons qualifier d'élémentaires ou d'ordre 1. Un ou plusieurs

- nœuds peuvent être agrégés en zone élémentaire. Plusieurs zones élémentaires peuvent être agrégées à leur tour en zones plus larges d'ordre 2 donc et ainsi de suite ;
- Les nœuds et les zones d'agrégation connues d'un nœud sont inscrits dans sa table de routage ;
  - Les nœuds s'échangent périodiquement leurs informations de routage afin de maintenir leurs tables à jour ;
  - Le choix du prochain saut pour le routage d'un paquet correspond à la passerelle qui permet d'atteindre la destination (nœud ou zone) la plus proche de la destination finale.

L'idée d'une approche hybride topologique-géographique a déjà été proposée dans la littérature. Dans [79] pour les destinations proches, un routage proactif à vecteurs de distance classique est utilisé, pour les destinations éloignées, un routage glouton utilisant comme destination intermédiaire des ancres. La découverte des ancres se fait suivant une approche réactive, c'est-à-dire que la recherche des ancres n'est déclenchée qu'au niveau d'un maximum local. La proposition dans [80] reprend le même principe proactif pour les destinations proches. Par contre, pour les destinations éloignées, il est proposé un routage géographique choisissant de manière aléatoire des points intermédiaires. Ces derniers servent de relais pour les paquets arrivant à un maximum local. Notre principale différence par rapport à ces approches consiste à utiliser uniquement les contenus des tables pour les décisions de routage quelque soit la distance à la destination. Nous proposons donc une procédure de routage plus simple qui demeure inchangée peu importe la destination. Nous avons vu plus haut que trouver le bon timing pour le changement de mode de fonctionnement représente un défi étant donné le caractère imprévisible des réseaux ad hoc mobiles. Par ailleurs, nous pensons qu'anticiper la présence de maximums locaux est plus intéressant en termes de qualité de routes que d'utiliser une procédure de contournement qui allonge la longueur des chemins. Notre système d'agrégation permet d'obtenir des informations sur comment joindre des zones éloignées situées potentiellement derrière un vide là où les deux approches précédentes doivent être suffisamment proche d'un maximum local avant de réagir. Enfin notons que nous avons d'abord choisi une approche proactive pour le remplissage des tables de routage mais il est aussi possible d'utiliser une approche réactive, le concept d'agrégation de nœuds et de zones est indépendant de ce choix.

### 3.4.3 Les deux types de zones pour le routage

#### Zone locale

Chaque nœud définit une *zone locale* qui lui est propre. Il s'agit d'une zone géographique entourant celui-ci. La taille de cette zone peut être définie en termes de

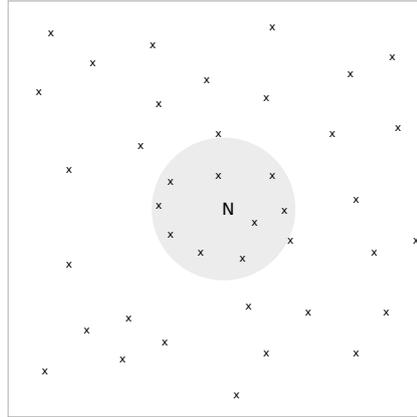


FIGURE 3.1 – Zone locale.

nombre de sauts, c'est-à-dire que les nœuds appartenant à cette zone sont ceux qui se trouvent à un nombre défini de sauts maximum du nœud considéré [24]. L'étendue de cette zone peut être aussi limitée en termes de distance géographique. Dans ce cas, la zone locale est définie par  $d(N, P) \leq l$  où  $d$  est une fonction distance,  $N$  le nœud considéré,  $P$  un point de la zone de déploiement du réseau et  $l$  la distance maximum autour de  $N$  où un nœud doit se trouver pour appartenir à la zone locale.

Un nœud enregistre les informations de lien de tous les nœuds se trouvant à l'intérieur de sa zone locale. En d'autres termes, chaque nœud à l'intérieur de cette zone provoque une entrée dans la table de routage. Il n'y a pas d'agrégation des informations les concernant. Sur l'exemple de la figure 3.1, la zone locale est en gris.

### Zone d'agrégation

Les informations de lien vers certains nœuds du réseau sont agrégées en liens vers certaines zones géographiques appelées *zones d'agrégation*. Nous utilisons les principes suivants pour la constitution des zones d'agrégation :

- Nous définissons des zones d'agrégation *élémentaires* ou d'ordre 1 qui sont identifiables de manière unique ;
- Les zones d'agrégation peuvent être agrégées en zones d'agrégation plus larges, d'ordre supérieur qui sont aussi identifiables de manière unique ;
- L'ordre d'agrégation de chaque zone est connu.

Nous définissons donc une hiérarchie de zones d'agrégation : plusieurs zones d'agrégation d'ordre  $n$  sont agrégées en zones d'agrégation d'ordre  $n + 1$ . Par ailleurs, les zones d'agrégation élémentaires, c'est-à-dire, d'ordre 1 résultent d'un partitionnement globale de la zone de déploiement du réseau. Nous précisons que nous parlons d'un partitionnement général, nous n'imposons pas un mode de partitionnement en particulier. Par contre lors de l'implémentation de notre proposition, un mode de partitionnement logique doit être adopté (voir chapitre 4). Aussi dans les figures qui

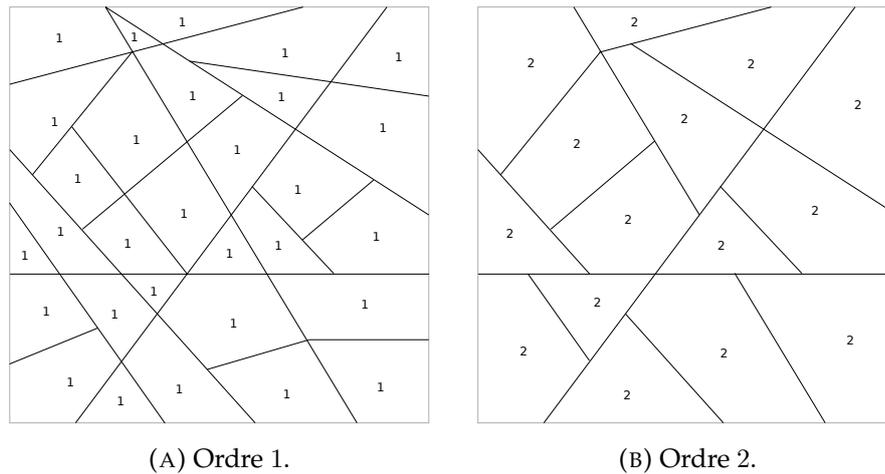


FIGURE 3.2 – Exemple de zones d'agrégation.

suivent, nous allons illustrer le partitionnement selon des découpages arbitraires.

Sur la figure 3.2a, la zone de déploiement du réseau a été partitionnée en zones d'agrégation d'ordre 1. Ensuite, sur la figure 3.2b, les zones d'agrégation d'ordre 1 proches géographiquement ont été agrégées en zones d'agrégation d'ordre 2. Il en résulte un partitionnement du réseau en zones d'agrégation d'ordre 2. Notons qu'une zone d'agrégation d'ordre 1 ne peut appartenir qu'à une seule et unique zone d'agrégation d'ordre 2.

#### 3.4.4 Hypothèses

Pour la suite nous faisons les hypothèses suivantes :

- Chaque nœud possède un identifiant unique ;
- Chaque nœud a accès à sa position géographique à l'aide d'un GPS par exemple ;
- Un service de localisation permet de trouver la position d'un nœud à partir de son identifiant [81] ;
- La zone de déploiement est partitionnée en zones d'agrégation telles que définies dans la section précédente ;
- Pour simplifier la description de notre solution, nous considérons uniquement le cas statique. La mobilité fera l'objet d'une étude au chapitre 5.

#### 3.4.5 Structure d'une table de routage

La table de routage d'un nœud enregistre, d'une part, les informations de lien vers les nœuds dans sa zone locale, et d'autre part, les informations agrégées de lien vers les nœuds situés dans les zones d'agrégation. Nous divisons donc la table en deux parties :

TABLE 3.1 – Structure d’une table de routage.

Zone(A)	Destination(D)	Passerelle(GW)	Coût(C)
locale	nœud	nœud	...
	...		
locale	nœud	nœud	...
ordre 1	zone	nœud	...
	...		
ordre 1	zone	nœud	...
ordre 2	zone	nœud	...
	...		
ordre 2	zone	nœud	...
	...		
ordre <i>max</i>	zone	nœud	...
	...		
ordre <i>max</i>	zone	nœud	...

- Informations locales : contenant les routes vers les nœuds appartenant à la zone locale. Chaque ligne de cette partie correspond à une route vers un nœud ;
- Informations agrégées : contenant les routes vers les zones d’agrégation connues pour contenir au moins un nœud. Chaque ligne de cette partie correspond donc à une route vers une zone d’agrégation.

Le tableau 3.1 montre la structure typique d’une table de routage. A chaque ligne correspond une route. Nous allons expliciter le contenu de chaque colonne :

- *Zone (A)* correspond à la nature de la route. *Locale* signifie qu’il s’agit d’une route menant vers un nœud situé dans la zone locale. *Ordre n* indique qu’il s’agit d’une route menant vers une zone d’agrégation d’ordre *n*, l’ordre maximum étant *max* ;
- *Destination (D)* correspond à l’identifiant de la destination de la route. Il peut s’agir soit d’un nœud soit d’une zone d’agrégation. L’identifiant d’un nœud étant par nature différent de celui d’une zone, la valeur de ce champ permet aussi de distinguer la nature de la destination (nœud ou zone) ;
- *Passerelle (GW)* correspond à l’identifiant du nœud par lequel un paquet à destination de *D* doit passer. Il s’agit donc du prochain saut pour joindre *D* ;
- *Coût (C)* correspond au coût pour aller du nœud courant vers *D* selon une métrique spécifique comme le nombre de sauts par exemple.

### 3.4.6 Maintenance des tables de routage

Dans notre approche, les décisions de routage reposent exclusivement sur la consultation des tables. Il est important que ces dernières contiennent des informations à jour. Pour cela, nous proposons un mécanisme similaire à celui utilisé par les protocoles proactifs comme dans DSDV [17]. Chaque nœud envoie périodiquement le contenu de sa table de routage uniquement à ses voisins à un saut. La différence se situe au niveau du traitement des informations de routes reçues : un nœud doit agréger certaines informations avant de les enregistrer.

#### Décision d'agrégation

Comme évoqué plutôt, la table de routage d'un nœud représente sa connaissance, d'une part, des informations de lien vers les nœuds situés dans sa zone locale et, d'autre part, des informations agrégées de lien vers les nœuds dans les zones d'agrégation. En d'autres termes, toutes les informations provenant de l'extérieur de la zone locale sont agrégées. De plus, nous avons posé comme principe que plus une information concernant un nœud ou une zone est éloignée, plus le niveau d'agrégation est élevé. En termes de zones, cela se traduit par une information représentée par une zone d'agrégation d'ordre plus élevé. L'ordre de la zone d'agrégation est donc fonction de la distance entre le nœud recevant l'information de route et le nœud concerné par celle-ci. L'ordre de la zone d'agrégation qui représentera une information reçue est donné par l'équation 3.1.

$$ordre = \begin{cases} 1 & \text{if } 0 < d(N, P) \leq x_1 \\ 2 & \text{if } x_1 < d(N, P) \leq x_2 \\ \dots & \\ n & \text{if } x_{n-1} < d(N, P) \leq x_n \end{cases} \quad (3.1)$$

où  $N$  représente le nœud recevant l'information,  $P$  le nœud ou la zone concernée par l'information,  $d$  une fonction distance et  $x_1 < x_2 < \dots < x_n$  des distances de références.

Sur l'exemple de la figure 3.3 nous avons une représentation des zones d'agrégations considérées par le nœud  $N$  avec leurs ordres respectifs. A partir du partitionnement sur la figure 3.2,  $N$  identifie la zone d'agrégation d'ordre 1 à laquelle il se trouve. Puis il considère les zones d'agrégation d'ordre 1 adjacentes à sa propre zone. Les autres zones d'agrégation d'ordre 1 sont agrégées en zone d'agrégation d'ordre 2. Pour définir les zones d'agrégation d'ordre 2 qu'il va considérer,  $N$  identifie d'abord celle à laquelle il appartient et applique de nouveau les règles précédentes et ainsi de suite. Nous constatons que plus on s'éloigne de  $N$ , plus l'ordre des zones d'agrégation est élevé.

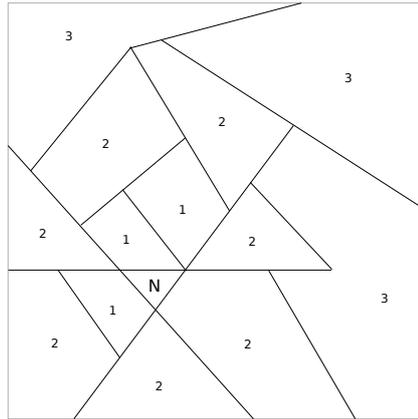


FIGURE 3.3 – Zones d'agrégation de N.

### Conventions

Dans la suite, nous allons représenter une route, c'est-à-dire, une ligne de la manière suivante:  $\{zone, destination, passerelle, coût\}$ . Notons que pour communiquer une route, seuls les champs *Destination* et *Coût* sont nécessaires. En effet, le champ *Zone* est déduit du champ *Destination*. Le champ *Passerelle* correspond à l'identifiant du nœud par lequel la route a été reçue. Un message de route sera donc représenté par  $\{destination, coût\}$ , le coût étant celui calculé par la passerelle.

### Initialisation

A l'initialisation, un nœud que nous allons identifier par N ajoute à sa table de routage, notée  $T(N)$ , une entrée correspondant à ses propres informations, c'est-à-dire,  $\{local, N, N, 0\}$ .

### Heuristique

Dans cette partie, nous allons décrire l'heuristique suivi par un nœud recevant une information de routage pour mettre à jour sa table de routage.

Soit un nœud N recevant le message  $\{D, C\}$  de la part d'un de ses voisins à un saut noté GW. Dans la suite,  $C'$  correspond à la mise à jour du coût C à la fin du traitement du message. Par exemple, si on prenait comme métrique le nombre de sauts,  $C' = C + 1$ .

- Si D est un nœud :
  - Si D appartient à la zone locale :
    - S'il n'y a pas encore de route vers D à l'intérieur de  $T(N)$ , la route  $\{locale, D, GW, C'\}$  est ajoutée à  $T(N)$  ;
    - S'il y a déjà une route vers D, celle-ci est remplacée si le coût de la nouvelle route est moindre.

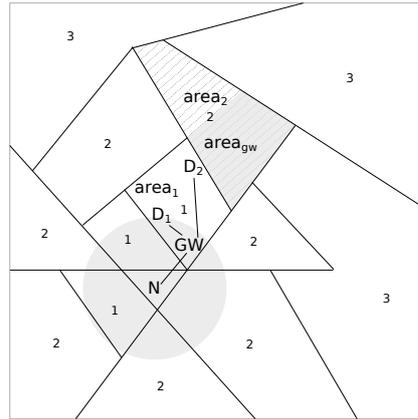


FIGURE 3.4 – Exemple de mise à jour de table.

- Si  $D$  n'appartient pas à la zone locale :
  - Identifier la zone d'agrégation d'ordre 1 contenant  $D$ . Notons la par  $Z$  ;
  - Reprendre avec  $D = Z$ , c'est-à-dire, faire comme si le message  $\{Z, C\}$  a été reçu.
- Si  $D$  est une zone d'agrégation :
  - Trouver la plus petite zone d'agrégation considérée par  $N$  contenant  $D$ . Notons par  $A$  cette zone et par  $n$  son ordre :
    - Si une route vers  $A$  n'existe pas encore dans  $T(N)$  ou existe mais avec un coût supérieur, ajouter la route  $\{\text{ordre } n, A, \text{GW}, C'\}$  ;
    - Sinon, ignorer.

Nous constatons qu'à mesure qu'une information se propage, elle est de plus en plus agrégée. Une information ne se propage pas telle quelle à l'échelle du réseau ce qui constitue la principale différence par rapport à un protocole de routage par approche topologique proactif classique. Cela permet des avantages en termes de consommation de bande passante pour la maintenance des tables comme nous verrons avec plus détails plus loin.

### Exemple

Sur la figure 3.4 nous avons une représentation de la zone locale et des zones d'agrégation du nœud  $N$ . Nous faisons l'hypothèse que la table de  $N$  vient d'être initialisée et ne contient donc qu'une seule entrée correspondant à ses propres informations.  $N$  reçoit les messages de route suivants de la part d'un nœud voisin identifié par  $\text{GW}$  :

- $\{D_1, C\}$  :  $D_1$  appartient à la zone de localité. Ainsi  $N$  ajoute l'entrée  $\{\text{locale}, D_1, \text{GW}, C'\}$  à sa table ;
- $\{D_2, C\}$  :  $D_2$  se trouve en dehors de la zone de localité. La plus petite zone d'agrégation contenant  $D_2$  est  $\text{area}_1$ . Par la suite, la table est enrichie avec l'entrée  $\{\text{ordre } 1, \text{area}_1, \text{GW}, C'\}$ . L'information a été agrégée ;

- $\{area_{gw}, C\}$  :  $area_{gw}$  est une zone d'agrégation d'ordre 1. A la distance où se trouve  $area_{gw}$ ,  $N$  n'enregistre dans sa table que des zones d'agrégation au moins d'ordre 2. Ainsi  $area_{gw}$  est agrégée en  $area_2$  d'ordre 2 qui est la plus petite zone d'agrégation considérée par  $N$  contenant  $area_{gw}$ . Aussi l'entrée ajoutée par  $N$  est  $\{ordre\ 2, area_2, GW, C'\}$ .

### 3.4.7 Procédure de routage

Il s'agit ici, pour un nœud devant envoyer un paquet, de déterminer le prochain saut auquel le paquet sera confié afin que le paquet soit remis à la destination. Notre proposition est un routage géographique reposant sur l'utilisation de tables de routage. Ainsi, le choix du prochain saut repose uniquement sur la consultation de la table de routage dans un contexte géographique.

Quand un nœud reçoit un paquet, classiquement, il vérifie d'abord s'il est le destinataire ou pas. S'il ne l'est pas, il applique la procédure de sélection du prochain saut. Ce processus est répété par chaque nœud jusqu'à ce que le paquet arrive à destination.

Nous rappelons l'hypothèse selon laquelle la source connaît la localisation géographique de la destination soit en l'inférant à partir de son identifiant soit en utilisant un service de localisation. Un service de localisation a pour rôle de résoudre un identifiant en localisation géographique. Tous les protocoles de routage géographique ont recours à ce service. Une étude comparative des solutions existantes est disponible dans [81]. Une autre hypothèse que nous avons faite concerne la connaissance de la localisation des zones d'agrégation. Pour chaque zone d'agrégation, nous définissons un *point de référence*. Ce point de référence peut être le barycentre de la zone. La localisation géographique d'une zone correspond à celle de son point de référence.

Nous proposons un processus de sélection du prochain saut similaire à celui du routage glouton. Pour un protocole de routage purement géographique, c'est-à-dire, glouton, le prochain saut correspond au nœud permettant de se rapprocher le plus de la destination. Dans notre cas, le prochain saut correspond à la passerelle qui permet de joindre le nœud ou la zone d'agrégation qui est le plus proche de la destination.

### Heuristique

Nous décrivons l'heuristique suivi par un nœud recevant un paquet et choisissant le prochain nœud auquel ce paquet sera transféré.

Soit  $N$  un nœud recevant un paquet dont la destination finale est  $D$ .



plus d'inscription de nouvelles entrées ni de mises à jour. Pour un protocole de routage proactif classique, une nouvelle information de route provoque une mise à jour des tables de tous les nœuds. Dans notre proposition, cette information ne provoque des modifications qu'à un nombre limité de nœuds. Quand un nœud reçoit une information de route, il va l'agrèger si nécessaire. Plus la destination correspondant à l'information est éloignée, plus la zone d'agrégation à laquelle elle sera intégrée est étendue. Or plus la zone d'agrégation est vaste, plus la probabilité qu'une information de route pouvant être agrégée dans la même zone a déjà été reçue. Dans ce cas, il n'y a pas de mise à jour de la table et l'information n'est donc plus propagée. Ainsi, par rapport à un protocole de routage proactif classique, notre proposition nécessite un temps de convergence plus court, réduisant le délai entre le déploiement du réseau et sa disponibilité.

### Taille des tables de routage

La taille d'une table de routage correspond au nombre d'entrées à l'intérieur de celle-ci. Elle influence donc la quantité de mémoire nécessaire au niveau de chaque nœud mais surtout a un impact sur la consommation du protocole en termes de bande passante, étant donné que les informations qui y sont enregistrées sont échangées entre les nœuds. Pour qu'un protocole passe à l'échelle, il est donc important que la taille des tables de routage n'augmente pas ou peu avec la taille du réseau. Un protocole proactif classique a tendance à enregistrer les routes vers tous les nœuds. Notre proposition, en agrégeant ces routes, réduit donc le nombre d'entrées enregistrées dans les tables. De plus, en effectuant une agrégation graduelle, c'est-à-dire, en considérant des zones d'agrégation de plus en plus grandes avec la distance, le nombre d'entrées dans les tables croît de manière logarithmique avec le nombre de nœuds dans le réseau avec une densité constante. En effet, à densité constante, en augmentant le nombre de nœuds, on augmente la surface occupée par le réseau, mais on peut aussi utiliser des zones d'agrégation plus grandes. L'augmentation du nombre d'entrées potentielles dans les tables est freinée par la croissance de la taille des zones d'agrégation.

Ainsi par rapport à un protocole de routage proactif classique, notre proposition génère moins de trafic de contrôle tout en préservant la qualité des routes comme le montreront les résultats au chapitre 5.

### Taux d'acheminement de paquets

Il s'agit du rapport entre le nombre de paquets reçus par les destinataires et le nombre total de paquets émis. Par rapport à un protocole de routage proactif classique, notre proposition pourrait donner un taux d'acheminement inférieur. En effet l'agrégation rend les informations de routage moins précises en contrepartie d'une taille moyenne de tables de routage réduite. Par rapport à un routage géographique

pur, le taux d'acheminement doit être plus élevé. Deux raisons en sont à l'origine. D'abord les zones atteignables à partir d'un nœud ne sont pas limitées au seul voisinage de celui-ci. Elles peuvent se situer loin des voisins à un saut qui sont les seuls pris en compte par le routage géographique pur. Ainsi la probabilité de connaître une route menant vers une zone proche de la destination est plus élevée augmentant ainsi les chances d'atteindre celle-ci. Ensuite, les maximums locaux n'ont pas d'effet sur la propagation des informations de routes. Ainsi certaines zones d'agrégation peuvent être accessibles à travers une passerelle située plus loin de la destination considérée par rapport à la source. Notre proposition permet donc dans ce cas de contourner les vides, offrant un meilleur taux d'acheminement que le routage géographique pur.

En résumé, le taux d'acheminement de paquets de notre solution devrait se situer entre celui proposé par un routage proactif classique et un routage géographique pur. Des évaluations quantitatives sont nécessaires pour vérifier ces appréciations qualitatives.

### Mise en œuvre

Afin d'évaluer quantitativement notre proposition, une mise en œuvre des différentes propositions doit être réalisée. Pour réaliser ce travail, nous devons considérer les points suivants :

- Partitionnement du réseau pour les zones d'agrégation. Dans ce chapitre, nous avons illustré le fonctionnement de notre proposition avec un partitionnement quelconque de la zone de déploiement des nœuds. Étant donné les hypothèses que nous avons posées à savoir, localisation et identification unique de toutes les zones d'agrégation, il est plus intéressant d'utiliser un pavage régulier de l'espace de déploiement. À part sa simplicité, ce type de pavage facilitera l'identification des partitions résultantes. De plus, les zones d'agrégation étant elles-mêmes des agrégations de zones sauf celles d'ordre 1 et pouvant à leur tour être agrégées, il y a tout intérêt à organiser les partitions de manière hiérarchique. Enfin, un pavage régulier résulte en des zones d'agrégations de même ordre qui ont la même taille. Ainsi, dans le chapitre 4, nous proposons un partitionnement de la zone de déploiement en zones hiérarchiques sur un modèle de *quadtree*. Un système d'adressage pour nœuds et zones d'agrégations basé sur ce partitionnement est ensuite proposé apportant à la fois des informations de nature géographiques et hiérarchiques.
- Procédure de mise à jour des tables et d'acheminement des paquets. Par la suite, il s'agit d'utiliser le partitionnement et le système d'adressage développé dans le chapitre 4 pour préciser et implémenter les algorithmes de mise à jour des tables et d'acheminement des paquets. La prise en compte de la mobilité implique entre autres l'enrichissement des informations retenues pour

chaque route. Ainsi chaque partie de la présente section sera reprise une à une au chapitre 5 en tenant compte de ces éléments.

Une fois ces étapes effectuées, l'évaluation quantitative peut être réalisée. Nous avons choisi d'utiliser le simulateur ns-3 [82]. Les résultats sont détaillés au chapitre 5.

### 3.5 Conclusion

Nous avons souligné l'importance de la faculté à passer à l'échelle pour certaines applications des réseaux sans fil ad hoc. Cette propriété s'acquiert par l'utilisation, entre autres, d'un protocole de routage passant également à l'échelle. Nous avons introduit un nouveau modèle de routage. Les routes sont enregistrées dans des tables de routage qui sont maintenues par les échanges entre nœuds voisins à un saut. Une méthode d'agrégation de routes par l'intermédiaire de zones d'agrégation est proposée afin de réduire la taille des tables de routage. L'objectif est de réduire la consommation en bande passante pour la maintenance du protocole de routage. Les zones d'agrégation résultent du partitionnement de la zone de déploiement des nœuds du réseau. L'acheminement des paquets repose uniquement sur la lecture des tables de routages en tenant compte des aspects géographiques. L'accès à des informations de routage relatives à des destinations éloignées même sous forme agrégée autorise l'anticipation de certains maximums locaux et leur contournement. Les discussions sur les performances ont abouti à une prévision de volume de trafic de contrôle plus faible qu'avec un routage ad hoc classique et une meilleure découverte de routes par rapport à un routage géographique pur. En d'autres termes nous avons un compromis entre un routage proactif classique et un routage géographique pur. L'objectif est maintenant de vérifier ces prédictions de manière quantitative en développant le modèle. Cette mise en œuvre repose sur un partitionnement particulier de la zone de déploiement ainsi qu'un adressage hiérarchique et géographique des nœuds et des zones d'agrégation qui seront décrits au chapitre 4. Par la suite, le chapitre 5 reprendra chaque partie de la section précédente en prenant en compte les propositions du chapitre 4 ainsi que la mobilité.

## Chapitre 4

# Autostructuration par adressage hiérarchique géographique

### 4.1 Introduction

Nous avons introduit au chapitre précédent une proposition de routage reposant sur la combinaison de deux approches : une approche topologique classique au sein d'une zone limitée, et au-delà, une approche géographique à travers l'agrégation des nœuds suivant des critères géographiques. Un groupe de nœuds agrégés est représenté par une zone d'agrégation. Nous avons vu que les zones d'agrégations sont issues d'un partitionnement de la zone de déploiement du réseau. De plus, elles doivent respecter les propriétés suivantes :

- Être identifiables de manière unique ;
- Être agrégeables entre elles, c'est-à-dire, avoir une structure hiérarchique.

A l'avant dernière section du chapitre précédent, nous avons évoqué la nécessité d'un partitionnement régulier donnant des zones d'agrégation de même ordre ayant la même taille. Cela facilite l'organisation hiérarchique des partitions résultantes.

Dans ce chapitre, nous proposons deux principes utilisés pour la mise en œuvre de l'agrégation des nœuds :

- Un mode de partitionnement hiérarchique du plan ;
- Un adressage des nœuds sur la base des régions issues du partitionnement précédent.

Le mode de partitionnement proposé consiste à diviser récursivement le plan afin d'obtenir des régions hiérarchiques. Ensuite une numérotation est mise en place afin d'identifier chaque région. Un numéro indique à la fois la localisation géographique d'une région et sa position dans la hiérarchie.

Les numéros de régions sont utilisés pour attribuer des adresses hiérarchiques aux nœuds. Selon la nature des informations géographiques accessibles aux nœuds, nous distinguons deux cas de figure :

- Positionnement absolu : par GPS, un nœud identifie la région dans laquelle il se trouve et utilise le numéro de celle-ci en tant qu'adresse. Il n'y a donc aucun échange nécessaire entre les nœuds pour l'acquisition des adresses ;

- Positionnement relatif : le partitionnement du plan est utilisé comme plan d'adressage. Un nœud configure son adresse en échangeant avec ses voisins déjà configurés dans le but de respecter ce plan d'adressage.

De manière générale, l'adressage ou l'assignation d'adresse aux nœuds est préalable à toute forme de communication dans un réseau puisqu'elle sert à l'identification des protagonistes. Il est aussi un prérequis pour le routage car il est bien-sûr nécessaire d'identifier et de localiser la destination avant de pouvoir y envoyer un paquet. Dans notre proposition de routage hybride, les adresses jouent un rôle majeur car en plus d'apporter des informations de localisation, elles sont utiles pour la mise en œuvre de l'agrégation géographique des nœuds.

Ce chapitre est organisé comme suit : dans un premier temps, nous allons expliquer les motivations ayant conduit à notre proposition d'adressage. Puis, nous expliquerons notre mode de partitionnement hiérarchique. Enfin, nous détaillerons deux moyens de mise en œuvre de l'adressage selon la nature des informations géographiques accessibles aux nœuds, suivis des évaluations qualitatives et quantitatives.

## 4.2 Problématiques et motivations

L'adressage est un problème important en réseau car il sert à l'identification des nœuds et il constitue un prérequis aux communications entre les nœuds. Par ailleurs, tout comme le routage, l'adressage dans un réseau ad hoc doit faire face aux propriétés particulières de celui-ci, à savoir le fonctionnement distribué, l'égalité des nœuds, la mobilité etc. Ainsi, aucune entité centrale ni extérieure n'est responsable de l'adressage des nœuds contrairement aux usages dans les réseaux classiques. Dans les réseaux classiques construits sur des liens filaires, l'affectation d'adresses repose sur un ou plusieurs serveurs centraux qui répondent aux requêtes de demande d'adresse des nouveaux équipements. Dans un réseau ad hoc, les nouveaux nœuds obtiennent leur adresse en interagissant avec des nœuds déjà configurés, on parle alors de protocole d'autoconfiguration d'adresses. Un tel protocole est donc tenu d'attribuer les adresses de manière totalement distribuée avec un minimum de coût en termes de bande passante et en s'adaptant aux changements de topologies dans le réseau.

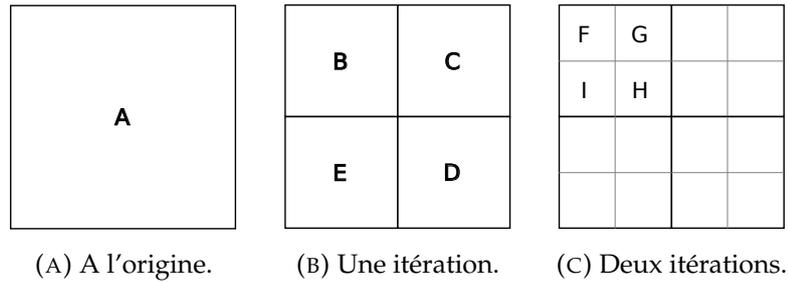
L'adressage de tous les nœuds du réseau grâce à un protocole de configuration d'adresses est un prérequis au fonctionnement d'un protocole de routage. En effet, ce dernier s'appuie sur les adresses des nœuds pour l'acheminement des paquets. Des incohérences au niveau des adresses peuvent ainsi conduire à un routage inefficace. L'adressage et le routage sont donc deux fonctions liées. Chacune peut être conçue de manière indépendante, mais des travaux soulignent les intérêts d'une synergie entre les deux. Dans [83], les auteurs proposent un protocole d'adressage intégré avec le protocole de routage *Optimized Link State Routing Protocol* (OLSR). Il

s'agit de mutualiser les informations nécessaires à la maintenance des deux protocoles. Les résultats montrent des économies en matière de consommation de bande passante. Dans d'autres cas, le routage s'appuie sur un type particulier d'adressage. Par exemple, dans [33], le routage exploite un adressage hiérarchique et ne fonctionnerait pas avec un adressage à plat. En effet, les algorithmes de routage de cette proposition nécessitent obligatoirement des adresses ayant des relations hiérarchiques entre elles. Nous adhérons à cette idée d'intégration entre adressage et routage. En effet, nous proposons un système d'adressage conçu spécialement pour notre proposition de routage.

Nous avons déjà évoqué les avantages de l'auto-organisation par structuration [14] [84]. Notre proposition de routage repose sur une agrégation sur critères géographiques des nœuds en zones d'agrégations issues d'un partitionnement de la zone de déploiement du réseau. Ces zones doivent avoir une structuration hiérarchique pour avoir plusieurs niveaux d'agrégation. Il s'agit donc de partitionner le réseau en régions organisées hiérarchiquement : une région est formée de plusieurs régions de niveau hiérarchique inférieur tandis que plusieurs régions peuvent constituer une région de niveau hiérarchique supérieur. L'adressage que l'on propose relie un nœud à la région dans laquelle elle se trouve. Nous avons donc automatiquement une structuration hiérarchique des nœuds à travers leurs adresses.

Nous avons vu au chapitre précédent les avantages de la mise en place d'une hiérarchie des nœuds pour adresser la problématique du passage à l'échelle dans les protocoles de routage [85]. Le principal inconvénient de la hiérarchisation est le coût lié à la maintenance de celle-ci. Dans notre proposition, il ne s'agit pas d'organiser à proprement parler une hiérarchie des nœuds. Ce sont les adresses des nœuds qui ont des propriétés hiérarchiques. On peut donc dire que les nœuds sont structurés hiérarchiquement à travers leurs adresses. Cette structuration est héritée de l'organisation hiérarchique des régions auxquelles les nœuds s'identifient pour acquérir leur adresse. Ainsi, sa maintenance ne nécessite pas d'échanges entre les nœuds, il suffit au nœud de vérifier régulièrement la région dans laquelle elle se trouve et de mettre à jour son adresse en cas de changement. Un autre avantage de ce mécanisme est qu'il n'est pas nécessaire d'élire des nœuds ayant des responsabilités particulières pour sa mise en œuvre, rendant la structuration plus robuste notamment face à la mobilité. Pour qu'un nœud puisse identifier la région à laquelle il appartient, il doit avoir accès à sa position géographique en utilisant un GPS, par exemple. Il existe des techniques permettant aux nœuds de connaître leur localisation géographique relativement à d'autres nœuds [86]. Nous avons donc aussi exploré le cas où seules ces informations sont accessibles aux nœuds afin d'étudier la faisabilité de notre système d'adressage dans ces conditions.

L'adresse d'un nœud dépend de la région issue de la partition dans laquelle il

FIGURE 4.1 – Partitionnement suivant un modèle de *quadtree*.

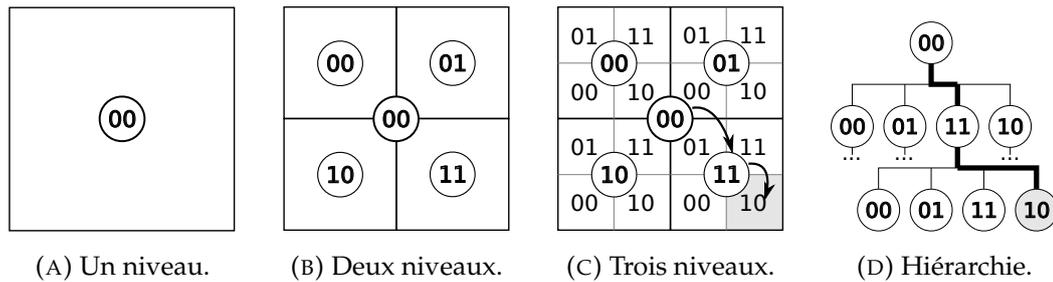
se trouve. En conséquence, l'adresse d'un nœud change en fonction de sa position dans le réseau. Nous faisons donc une distinction entre l'identifiant unique et inchangé d'un nœud et son adresse qui dépend de sa localisation.

Enfin, comme nous l'avons vu au chapitre 2, pour la majorité des protocoles d'adressage, l'unicité des adresses fournies est obligatoire, c'est-à-dire qu'aucune adresse ne doit être partagée par deux ou plusieurs nœuds. Notre système d'adressage est moins contraignant dans le sens où des adresses communes sont proposées. En effet, les adresses renseignent sur la localisation géographique des nœuds. Deux ou plusieurs nœuds proches géographiquement peuvent donc avoir la même adresse sans compromettre le processus de routage, la destination finale étant toujours un nœud ayant un identifiant unique. Par contre, deux nœuds éloignés ne peuvent pas avoir la même adresse. Il existe d'autres avantages à l'utilisation d'adresses qui ne sont pas uniques comme nous les verrons plus loin.

## 4.3 Partitionnement du plan

### 4.3.1 Partitionnement en régions hiérarchiques

On considère un partitionnement du plan suivant un modèle de *quadtree*. Le plan est divisé récursivement en quatre régions. Sur la figure 4.1, la région A est divisée en quatre (figure 4.1b) puis chaque sous région est divisée ensuite en quatre (figure 4.1c) et ainsi de suite. Pourquoi quatre ? L'idée principale est de diviser une grandeur en deux de manière récursive. Appliquée à un espace à deux dimensions, cela équivaut à une division en quatre. A l'issue du partitionnement, on obtient une hiérarchie de régions. Chaque région peut avoir exactement zéro ou quatre *régions-filles*. Sur la figure 4.1, les régions-filles de la région A sont B, C, D et E. On appelle *régions-sœurs* d'une région, les trois autres régions formant les régions-filles issues de la division en quatre d'une région appelée *région-mère*. Sur notre exemple, les régions-sœurs de B sont C, D et E, leur région-mère étant A. Les régions *élémentaires* sont les plus petites régions obtenues après arrêt du partitionnement. Une région élémentaire n'a donc

FIGURE 4.2 – Partitionnement hiérarchique du plan (*top-down*).

pas de régions-fille. F, G, H et I sont des exemples de régions élémentaires (figure 4.1c).

### 4.3.2 Numérotation des régions

Un nombre que l'on va, par la suite, appeler *numéro* est attribué à chaque région. Le numéro est choisi de manière à fournir des informations, d'une part sur la position de la région considérée dans la hiérarchie, et d'autre part, sur sa localisation géographique. Pour cela, chaque sous-partie d'un numéro représente un niveau hiérarchique donné. Deux régions appartenant à un même niveau ont donc des sous-parties en commun dans leurs numéros. Par ailleurs, d'après le mode de partitionnement utilisé, une région peut être localisée en considérant successivement et à partir du niveau le plus haut les régions dans lesquelles elle se trouve, à la manière d'un zoom progressif sur une zone. Le numéro d'une région dont chaque sous-partie correspond à un niveau donné, renseigne toutes les régions de niveau supérieur contenant celle-ci et permet, de ce fait, sa localisation. Dans la suite, pour des questions pratiques, on va utiliser la représentation en base 2 des numéros. En effet, il suffit dans ce cas d'effectuer des décalages binaires pour trouver les sous-parties d'un numéro et d'identifier ainsi les régions de niveau supérieur contenant la région représentée par le numéro.

### 4.3.3 Algorithme

Notons que, puisqu'une région a exactement zéro ou quatre régions filles, chaque région peut avoir comme numéro 00, 01, 11 ou 10. Quatre régions-sœurs doivent chacune occuper ces quatre numéros. Un ordre de numérotation des régions-sœurs est convenu et gardé tout au long du partitionnement : 00 - 01 - 11 - 10 de façon horaire ou anti-horaire. Il s'agit d'un choix arbitraire. Sur l'exemple de la figure 4.2, on suit le sens des aiguilles d'une montre. En gardant cet ordre de numérotation, on s'assure que deux régions d'un même niveau et adjacentes verticalement ou horizontalement (resp. diagonalement) ont exactement un bit (resp. deux bits) de différence entre leurs numéros. Pour la suite, on va choisir un sens horaire de numérotation.

Maintenant considérons une région carrée du plan, de taille arbitraire, à laquelle nous allons appliquer l'algorithme de partitionnement. Par convention, nous allons donner à cette région le numéro 00 (figure 4.2a). Remarquons qu'on aurait pu lui attribuer n'importe quel des trois autres numéros possibles.

Une itération consiste à diviser chacune des plus petites régions en quatre sous-régions. Une des sous-régions obtenues est numérotée par 00. A partir de ce point de départ, nous allons numéroter les trois autres sous-régions dans l'ordre défini plus haut, suivant un sens horaire. Cette opération est illustrée sur la figure 4.2b qui montre le résultat d'une itération à partir d'une région initiale. Par ailleurs, le point de départ (sous-région de numéro 00) est décalé dans un sens anti-horaire à chaque itération. La figure 4.2c affiche le résultat d'une deuxième itération. On constate le décalage du point de départ : sur la figure 4.2b, 00 est attribué à la région « nord-ouest » tandis que sur la figure 4.2c, celui-ci est attribué à la région « sud-ouest » en considérant les plus petites régions-filles. Si on appliquait une troisième itération, il serait attribué à la région « sud-est » et ainsi de suite. Décaler le point de départ, c'est-à-dire, la sous-région avec le numéro 00 à chaque itération donne une propriété d'autosimilarité au partitionnement. En effet, toutes les quatre itérations, le point de départ fait un tour complet. Par exemple, si le point de départ actuel est la région « nord-ouest », après quatre itérations, le point de départ est de nouveau la région « nord-ouest ». Grâce à cette propriété, il est possible de partitionner une zone donnée à l'infini. L'avantage est donc de pouvoir partitionner une zone de déploiement de réseau de taille arbitraire. Par ailleurs, jusqu'ici nous avons considéré un partitionnement qui part de l'espace total vers les régions élémentaires (approche *top-down*). Avec l'autosimilarité, il est aussi possible de faire l'inverse, c'est-à-dire, partir regrouper des régions élémentaires pour former une région supérieure et ainsi de suite (approche *bottom-up*). L'avantage est qu'il est toujours possible d'agrandir la zone partitionnée. D'autres propriétés seront explicitées plus loin. Sur la figure 4.2d, on a une représentation de la hiérarchie des régions. L'algorithme 4.1 détaille le processus de partitionnement d'une région selon l'approche *top-down*. L'argument `taille_maximale` spécifie la taille maximale de région élémentaire désirée. L'algorithme peut être initialisée avec n'importe quel entier naturel pour l'argument `niveau_courant`.

```
Partitionner (région, taille_maximale, niveau_courant)
{
  Si (région est élémentaire ET taille > taille_maximale)
    AjouterRégionsFilles (région, niveau_courant)
  Si (région n'est pas élémentaire)
    Pour chaque région-fille r de région
      Partitionner (r, taille_maximale, ++niveau_courant)
}
```

```

AjouterRégionsFilles (région, niveau_courant)
{
    région1.numero = (région.numero<<2)+Numero(niveau_courant)
    région2.numero = (région.numero<<2)+Numero(niveau_courant+1)
    région3.numero = (région.numero<<2)+Numero(niveau_courant+2)
    région4.numero = (région.numero<<2)+Numero(niveau_courant+3)
    région.régions_filles = {région1, région2, région3, région4}
}
Numero (niveau_courant)
{
    Selon (niveau_courant%4)
    {
        0: retourner 00
        1: retourner 01
        2: retourner 11
        3: retourner 10
    }
}

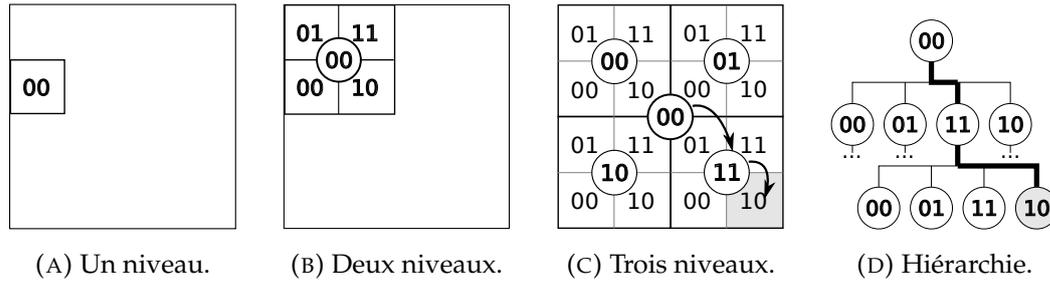
```

ALGORITHME 4.1 – Algorithme de partitionnement (top-down).

Comme évoqué plus haut, on constate une propriété d'autosimilarité de la disposition des numéros. En effet, toutes les quatre itérations, on retrouve le même point de départ de la numérotation, c'est-à-dire, la région de numéro 00. Par conséquent, il est aussi possible de passer par une approche *bottom-up* pour le partitionnement : au lieu de diviser une région initiale, on peut regrouper 4 régions afin de former une région-mère, qui sera à son tour regroupée avec d'autres régions-mères et ainsi de suite. Au contraire de la construction *top-down* précédente, à chaque fois qu'on monte d'un niveau dans la hiérarchie, le point de départ est décalé dans le sens horaire (figure 4.3). Le partitionnement *bottom-up* est utile quand la taille de la zone à partitionner n'est pas connue d'avance. Les deux approches sont compatibles et peuvent être utilisées ensemble dans le cas où une zone est d'abord partitionnée selon l'approche *top-down* et qu'ensuite, la zone partitionnée est étendue selon l'approche *bottom-up*.

Jusqu'ici, nous avons parlé du numéro *local* d'une région qui vaut 00, 01, 11 ou 10. Ce numéro permet de situer une région par rapport à ses régions-sœurs qui sont donc de même niveau hiérarchique. Pour situer une région par rapport à l'ensemble du plan partitionné, nous allons utiliser son numéro *global*. Il s'agit de la concaténation de son numéro local avec celui de toutes ses régions-mères par ordre hiérarchique décroissant. Par exemple, sur les figures 4.3c et 4.3d, la région élémentaire grisée a comme numéro local 10 et comme numéro global 001110.

Par ailleurs, il est possible de trouver les numéros globaux de toutes les régions élémentaires issues d'un partitionnement à l'aide d'une équation. En effet, on peut voir le plan partitionné comme une matrice dont les numéros globaux de chaque région correspondent aux coefficients. Soit  $M(n)$  cette matrice avec  $n$  le nombre de

FIGURE 4.3 – Partitionnement hiérarchique du plan (*bottom-up*).

niveaux désirés. Les coefficients de la matrice  $M(n)$  sont donnés par l'équation (4.1).

$$M(n) = \begin{pmatrix} f(n+2)||M(n-1) & f(n+1)||M(n-1) \\ f(n+3)||M(n-1) & f(n)||M(n-1) \end{pmatrix} \quad (4.1)$$

Avec

- $M(0) = ()$
- $||$  l'opérateur de concaténation
- $f(x) = \begin{cases} 00 & \text{si } x \bmod 4 = 0 \\ 10 & \text{si } x \bmod 4 = 1 \\ 11 & \text{si } x \bmod 4 = 2 \\ 01 & \text{si } x \bmod 4 = 3 \end{cases}$

#### 4.3.4 Propriétés

Pour générer un partitionnement avec un nombre donné de niveaux désirés, l'algorithme n'a besoin que des coordonnées géographiques d'une seule région spéciale en plus de sa dimension. Dans l'approche *top-down*, il s'agit de la région correspondant à la zone à partitionner. Dans l'approche *bottom-up*, il s'agit de la région élémentaire qui aura le numéro 00. Il n'est donc pas nécessaire de mémoriser l'ensemble des régions issues d'un partitionnement donné. Les régions-filles de régions de même niveau ont la même structure. Ainsi, l'élargissement de la zone de partitionnement ne modifie pas le partitionnement déjà en place. En d'autres termes, lors de l'utilisation du partitionnement pour l'adressage des nœuds, l'agrandissement de la zone partitionnée ne modifie pas les adresses des nœuds déjà configurés.

Dans l'approche *bottom-up*, le décalage de la région ayant comme adresse 00 à chaque augmentation de niveau permet une croissance de la zone partitionnée en spirale. Ainsi, tout point déterminé du plan est atteignable et donc adressable au bout d'un nombre fini d'augmentations de niveau. Cette croissance en spirale est montrée sur la figure 4.4. La spirale dessinée passe, par ordre croissant de niveau, par toutes les régions dont le numéro local est 00, en gris, et contenant la région élémentaire originelle qui est aussi colorée en gris.

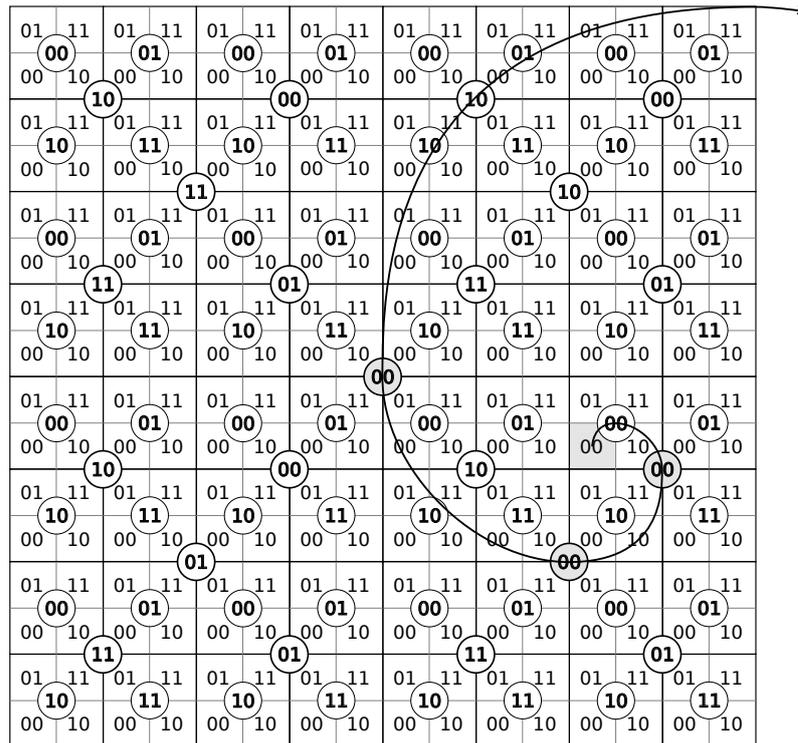


FIGURE 4.4 – Croissance en spirale du partitionnement.

### 4.3.5 Adressage

Nous proposons d'utiliser l'agencement des numéros des régions après partitionnement d'une zone en tant que plan d'adressage des nœuds dans un réseau. Puisqu'il s'agit d'un partitionnement géographique, pour que le plan d'adressage soit respecté, il est nécessaire que les nœuds aient accès à leurs positions géographiques. Selon la nature de ces informations qui peut être absolue ou relative aux autres nœuds du réseau, nous distinguons deux moyens d'affectation des adresses aux nœuds. Nous rappelons que l'adresse est à distinguer de l'identifiant invariable du nœud. Ce dernier peut, par exemple, être l'adresse MAC du nœud. Le reste de ce chapitre consistera à décrire ces deux approches.

## 4.4 Adressage hiérarchique géographique avec information de positionnement absolu

### 4.4.1 Hypothèses

Nous faisons l'hypothèse que chaque nœud a accès à sa localisation géographique au moyen d'un système de positionnement global comme le GPS ou local dans lequel il utilise un ou plusieurs éléments statiques, fixés à l'avance et connus de tous comme points de références pour la déduction de sa position [86].

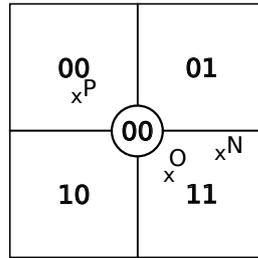


FIGURE 4.5 – Exemple d’adressage par positionnement absolu.

Une région carrée recouvrant au moins la zone de déploiement du réseau est partitionnée selon l’algorithme décrit dans la section précédente. Le même partitionnement est connu par tous les nœuds. Pour cela, les coordonnées de la région partitionnée ainsi que la taille d’une région élémentaire sont partagées entre les nœuds qui exécutent l’algorithme et aboutissent ainsi au même partitionnement.

#### 4.4.2 Acquisition d’adresse

En utilisant ses coordonnées géographiques, un nœud identifie la région élémentaire dans laquelle il se trouve et utilise le numéro *global* de celle-ci en tant qu’adresse. A titre d’exemple, sur la figure 4.5, le nœud N se trouve dans la zone élémentaire de numéro global 0010. Ainsi N a comme adresse 0010. De la même manière, le nœud P a comme adresse 0000. Par conséquent, deux nœuds appartenant à une même région élémentaire ont la même adresse. Sur l’exemple précédent, N et O partagent l’adresse 0010. Nous verrons que notre protocole de routage prend en compte cette situation.

#### 4.4.3 Changements topologiques

##### Mobilité

Quand un nœud change de position, le processus d’acquisition d’adresse est refait : il identifie la nouvelle région élémentaire dans laquelle il se trouve à partir de ses nouvelles coordonnées géographiques, puis change son adresse en accord avec le numéro global de cette région. La mobilité des nœuds n’induit donc pas de trafic supplémentaire. Par contre, il est nécessaire que les nœuds contrôlent périodiquement leurs positions géographiques pour que leurs adresses soient cohérentes avec le plan d’adressage. Que se passe-t-il si un nœud sort de la zone partitionnée ? Il lui suffit d’agrandir cette zone. En effet, comme vu plus haut, le partitionnement peut être étendu à l’infini grâce à ses propriétés d’autosimilarité.

## Partitionnement et fusion

Il s'agit de phénomènes provoqués par la mobilité des nœuds. Les nœuds sont donc juste tenus de mettre à jour leurs adresses en cas de changement de position. Ainsi le partitionnement et la fusion ne nécessitent aucune gestion particulière.

### 4.4.4 Discussions

#### Non-unicité des adresses

L'unicité des adresses a toujours fait partie des critères recherchés par les protocoles d'adressage dans les réseaux ad-hoc : deux ou plusieurs nœuds ne sont pas autorisés à utiliser la même adresse. En effet, l'adresse sert aussi comme identifiant. Dans notre approche, nous distinguons l'identifiant invariable d'un nœud et son adresse qui indique sa localisation. Il est donc tout à fait possible que deux ou plusieurs nœuds partagent la même adresse à condition qu'ils soient géographiquement proches les uns des autres, plus précisément, à portée les uns des autres. De cette manière, un paquet à destination d'un autre nœud avec la même adresse peut être transmis sans routage. Nous pouvons faire l'analogie avec les personnes qui habitent dans une même maison et qui utilisent la même adresse postale. Un des avantages est l'augmentation, par rapport à une approche classique, du nombre de nœuds adressables pour une longueur maximale d'adresse donnée. Cette approche résout aussi les problèmes de granularité dans le cas où il y a une forte concentration de nœuds dans une région donnée. Un système d'adressage hiérarchique par approche *top-down* peut être dans ce cas saturé car le nombre de nœuds adressables de manière unique par niveau est limité.

#### Intégration avec un protocole de routage

L'intégration avec un protocole de routage fait partie des propriétés recherchées pour un protocole d'autoconfiguration d'adresse [83] [33]. Notre système d'adressage a été conçu pour satisfaire les contraintes hiérarchiques et géographiques de notre modèle de routage (voir chapitre 3). Mais la structure hiérarchique de système d'adressage peut être profitable pour d'autres protocoles de routage hiérarchique [85]. Les adresses représentent aussi des informations de nature géographique. Elles sont donc aussi utilisables avec un protocole de routage géographique.

#### Compatibilité avec les applications sur IP

Notre solution ne peut pas fonctionner directement avec les applications sur IP puisqu'elle ne fournit pas des adresses IP. Nous rappelons que nous faisons une distinction entre identifiant et adresse. Nous proposons d'utiliser l'adresse IP d'un nœud en tant qu'identifiant. Ainsi, notre système d'adressage et le routage qui s'appuie dessus sont cachés aux applications s'appuyant sur IP.

## 4.5 Adressage hiérarchique géographique par information de positionnement relatif

Dans la section précédente, nous avons développé un système d'adressage des nœuds basé sur l'exploitation des informations de localisation absolue (GPS). Les nœuds peuvent aussi avoir accès uniquement à des informations de localisation relatives aux autres nœuds [86]. Nous avons donc décidé d'explorer la possibilité d'adapter notre solution à ces conditions. C'est l'objet de cette section.

### 4.5.1 Hypothèses

Nous faisons l'hypothèse que chaque nœud a accès à sa localisation géographique relative à son voisinage. Différentes approches existent pour accéder à cette information [86]. Nous pouvons citer celles reposant sur l'analyse des temps d'arrivée des paquets provenant des voisins [87], l'angle d'arrivée de ces paquets [88] ou les deux en même temps [89]. D'autres approches exploitent la puissance des signaux reçus [90].

Chaque nœud a connaissance de ses nœuds voisins à un saut. Un nœud signale sa présence par l'envoi, à ses voisins à un saut, de paquets de signalisation couramment appelés paquets « HELLO ». La liste des voisins est consignée dans une table qui est régulièrement mise à jour. Ces paquets consomment de la bande passante même en l'absence d'opérations du protocole. Aussi des solutions ont été proposées afin de réduire leurs coûts comme dans [91] ou [92].

Chaque nœud a connaissance d'un plan d'adressage qui correspond à la distribution des numéros de régions après partitionnement du plan selon l'algorithme décrit plutôt dans ce chapitre. La taille des régions pour l'exécution de l'algorithme n'est pas importante car un nœud n'acquiert pas une adresse par l'intermédiaire de sa présence dans une région. Seule l'organisation des numéros et leurs positions relatives les uns par rapport aux autres sont prises en compte. Pour avoir la même répartition des numéros, chaque nœud exécute l'algorithme de partitionnement avec le même nombre de niveau désiré et la même approche *top-down* ou *bottom-up*.

### 4.5.2 Procédure générale d'autoconfiguration

Nous proposons un algorithme distribué dont l'objectif est de configurer les nœuds avec des adresses cohérentes au sens du plan d'adressage. En d'autres termes, quand tous les nœuds sont configurés, la répartition de leurs adresses en fonction de leurs localisations géographiques est équivalente à celle des numéros des régions après partitionnement. Notons qu'il s'agit d'une répartition relative, c'est-à-dire que les adresses de deux nœuds voisins correspondent aux numéros de deux régions

voisines.

Un nœud souhaitant joindre un réseau existant envoie une *requête d'adresse*. Par la suite, un de ses voisins lui attribue une adresse respectant le plan d'adressage en s'appuyant sur sa position relative par rapport à ses propres voisins. Plus précisément, nous distinguons deux types de nœuds configurés :

- Un nœud *standard* ne répond pas aux requêtes d'adresse des nouveaux nœuds ;
- Un nœud *agent d'adresse* (AA) répond aux requêtes d'adresse provenant des nouveaux nœuds. Il attribue sa propre adresse à ceux-ci. En d'autres termes, un AA et les nœuds qu'il a configurés partagent la même adresse comme les nœuds appartenant à une même région. Nous précisons que les AA ne sont pas élus : un nœud choisit lui-même de devenir AA en fonction de l'arrivée de nouveaux nœuds dans son voisinage.

Par ailleurs, un réseau connexe possède un identifiant de réseau qui est choisi aléatoirement par le premier nœud configuré. Cet identifiant est indispensable pour la gestion des fusions.

### 4.5.3 Acquisition d'adresse

#### Nouveau nœud

Un nouveau nœud souhaitant acquérir une adresse envoie de manière périodique à tous ses voisins à un saut des requêtes d'adresse.

- Si un AA se trouve à portée, celui-ci répond avec sa propre adresse. Aussi le nouveau nœud acquiert la même adresse et le même identifiant de réseau que l'AA qui lui a répondu ;
- Si aucun AA se trouve à portée, les requêtes ne sont pas satisfaites. Au bout d'un nombre limité de requêtes non satisfaites, le nouveau nœud en déduit qu'il est le premier nœud du réseau. Ainsi il choisit l'adresse 00. En même temps, il crée un identifiant de réseau de manière aléatoire.

#### Devenir agent d'adresse (AA)

Quand un nœud standard reçoit de manière répétée des requêtes d'adresse provenant d'un même nœud, il en déduit qu'elles ne sont pas satisfaites. La raison est l'absence d'AA dans le voisinage du nœud source des requêtes. Il décide donc de devenir agent d'adresse afin de configurer le nouveau nœud. Pour cela, il ne peut plus utiliser la même adresse que son propre agent d'adresse sinon tous les nœuds du réseau auraient la même adresse.

Pour devenir agent d'adresse, un nœud standard envoie une requête spéciale que nous allons appeler *requête AA* à tous les agents d'adresse à sa portée. A la réception d'une telle requête, un agent d'adresse propose une ou plusieurs adresses

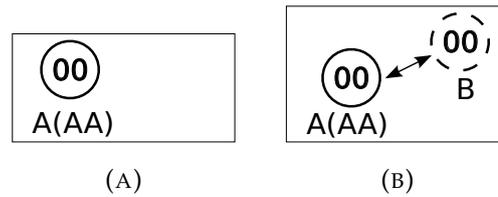


FIGURE 4.6 – Configuration du premier nœud et d'un nœud standard.

en réponse. Plus précisément, il s'agit pour chaque agent d'adresse contacté de sélectionner les adresses que le futur agent d'adresse devrait avoir pour respecter le plan d'adressage. Pour ce faire, chaque agent d'adresse tient compte de sa position relative, d'une part, par rapport aux agents d'adresse à portée, et d'autre part, par rapport au nœud à l'origine de la requête AA. Enfin, en réponse à une requête AA, un agent d'adresse communique les propositions d'adresses mais aussi sa propre adresse ainsi que celles des agents d'adresse à sa portée.

A la réception des réponses, le nœud souhaitant devenir agent d'adresse liste les adresses proposées en supprimant celles déjà occupées par les agents d'adresse et leurs voisins respectifs. Enfin, il choisit comme adresse celle proposée par le plus grand nombre d'AA ou, s'il n'y a pas de majorité, sélectionne aléatoirement une parmi les plus proposées. Si ce choix aléatoire introduit une incohérence par rapport au plan d'adressage, elle sera détectée et résolue par le mécanisme de fusion décrit plus bas.

### Exemple

Sur la figure 4.6a, le nœud A démarre. Après quelques envois infructueux de requêtes d'adresse, il s'auto-affecte l'adresse 00 et devient AA. Plus tard, le nœud B souhaite joindre le réseau (figure 4.6b). Pour se configurer, B envoie des requêtes que A répond avec sa propre adresse. Ainsi B aura aussi l'adresse 00.

Sur la figure 4.7a, le nœud C rejoint le réseau. Ses requêtes d'adresse ne sont entendues que par B. Celui-ci, n'étant pas AA, ne répond pas dans un premier temps. Mais constatant la répétition des envois de requêtes traduisant l'absence de réponse donc de AA à portée de C, le nœud B décide de devenir lui-même AA. Il envoie donc des requêtes AA à tous les AA dans son voisinage à un saut. Il s'agit ici de A et E. Sur la figure 4.7b nous avons un extrait du plan d'adressage avec les adresses utilisées par les AA. Les propositions d'adresse de A sont hachurées verticalement. Ces propositions tiennent compte de la position relative de A par rapport à D qui est son voisin, et de B. De même E propose les adresses hachurées diagonalement. Au final, B choisit l'adresse 01 qui est majoritaire. Il peut par la suite configurer C avec la même adresse (figure 4.7c).

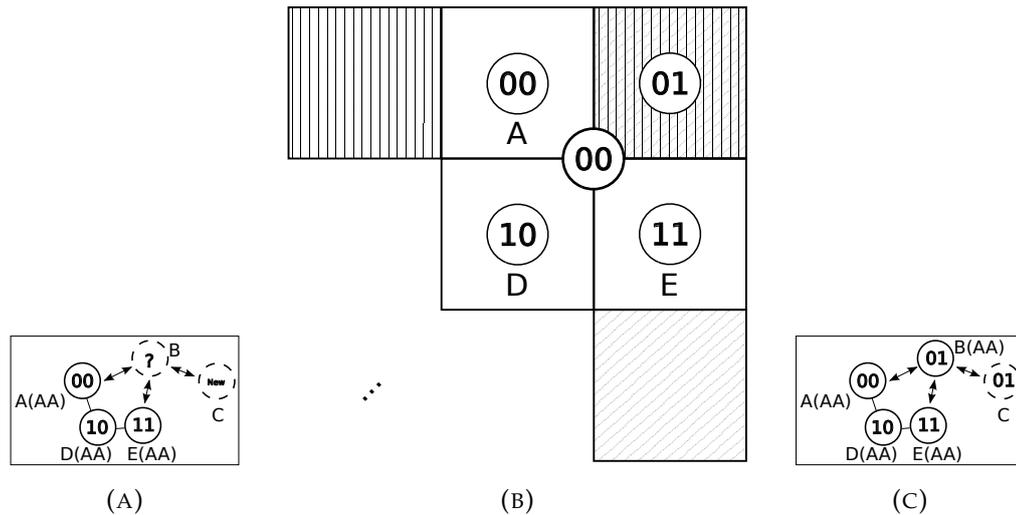


FIGURE 4.7 – Configuration d'un nœud en AA.

#### 4.5.4 Changements topologiques

##### Mobilité d'un nœud

Un nœud qui change de position qu'il soit AA ou nœud standard, agit comme un nouveau nœud : il envoie des requêtes d'adresse pour obtenir une nouvelle adresse compatible avec sa nouvelle position afin de respecter le plan d'adressage. Un nœud détecte sa mobilité ou celle de son voisinage en observant les changements dans sa table de voisinage. En plus de l'apparition ou de la disparition des voisins, un changement de position peut être aussi détecté en contrôlant la validité de son adresse compte tenu de sa position relative. Les nouveaux voisins du nœud n'ont pas de procédure particulière à suivre à part considérer l'arrivée d'un nouveau nœud et lui attribuer une adresse.

##### Partitionnement

Le partitionnement d'un réseau n'affecte pas la validité des adresses des nœuds dans les partitions résultantes. En effet, chaque partition correspond juste à un extrait du plan d'adressage. Il n'y a donc pas d'action spécifique à entreprendre en cas de partitionnement. Les nouveaux nœuds peuvent donc rejoindre normalement les partitions.

##### fusion

Nous rappelons d'abord que l'identifiant de réseau est un nombre choisi aléatoirement par le premier nœud fondateur d'un réseau et adopté par tous les nœuds le rejoignant. L'objectif est d'identifier facilement les nœuds membres d'un même réseau. Deux réseaux qui se développent sans entrer en contact possèdent donc des

identifiants de réseau différents. Une fusion apparaît quand des nœuds ayant des identifiants de réseau différents entrent en contact.

Il est nécessaire que tous les nœuds provenant de l'un des groupes rejoignent l'autre en acquérant de nouvelles adresses afin d'aboutir à un réseau unique. Comment décider lequel des deux groupes doit rejoindre l'autre ? Une possibilité consiste à comparer les identifiants de réseaux. Par la suite, les nœuds ayant des valeurs plus petites que celles de leurs voisins doivent demander une nouvelle adresse. A la fin, tous les nœuds finiront par rejoindre le réseau ayant le plus grand identifiant. Cette solution peut être la moins optimale dans le cas où la partition ayant l'identifiant le plus grand contient le moins de nœuds. Ainsi, il peut être intéressant de prendre en compte la taille des partitions afin de minimiser le nombre de nœuds qui changent de partition. L'évaluation de la taille d'un réseau peut être effectuée de manière distribuée [93].

#### 4.5.5 Discussions

##### Consommation de bande passante

Quelle est la quantité de trafic généré par le protocole durant les processus de configuration ? Toutes les opérations de notre protocole reposent sur des communications locales à un saut. Le nombre de paquets de configuration envoyés par un nœud ne dépend donc pas du nombre total de nœuds dans le réseau mais uniquement du nombre de voisins. Il s'agit là d'une propriété intéressante pour le passage à l'échelle. Par contre, la maintenance des informations concernant le voisinage implique l'envoi périodique de paquets « HELLO ». Notons qu'il est possible de réduire l'impact de ces derniers sur la consommation de bande passante en adaptant la fréquence des envois selon la mobilité des nœuds par exemple [94].

##### Latence

Il s'agit du temps qui s'écoule entre l'arrivée d'un nœud et son obtention d'une adresse valide. Le concept d'AA réduit la latence. En effet, si un nouveau nœud a un AA dans son voisinage, il est configuré après échange de deux messages correspondant à la requête d'adresse et la réponse, en faisant l'hypothèse qu'il n'y a pas de perte au niveau du canal de communication. Si un nouveau nœud n'a pas d'AA dans son voisinage, il faut ajouter le temps nécessaire à un nœud voisin pour devenir AA. Un nœud standard devient AA après échanges avec les AA dans son voisinage. Ainsi, le temps de configuration des nœuds ne dépend pas du nombre total de nœuds dans le réseau. Par ailleurs, plus il y a de nœuds déjà configurés, plus il y a de AA, étant donné qu'un nœud configuré a au moins un AA dans son voisinage. Ainsi, plus il y a de nœuds déjà configurés, plus la probabilité pour un nouveau nœud de trouver directement un AA dans son voisinage est grande, réduisant le délai nécessaire à sa configuration au minimum. De ce fait, nous pouvons avancer que, dans le cas où tous les nœuds d'un réseau s'allument en même temps,

le temps de convergence, temps nécessaire à tous les nœuds pour se configurer, n'est pas linéaire avec la taille du réseau.

### Faiblesses

Nous retenons les faiblesses suivantes :

- Quand un nœud devient agent d'adresse, il change d'adresse. Même si la nouvelle adresse est proche de l'ancienne dans le plan d'adressage, il est nécessaire de mettre à jour les informations au niveau des serveurs de localisation chargés de résoudre les identifiants en adresse alors que le nœud n'a pas changé de position. Cela induit du trafic de contrôle supplémentaire. De plus, le chemin emprunté par un paquet peut s'allonger car celui-ci arriverait à l'ancienne adresse avant de rejoindre éventuellement la nouvelle. Il faut dans ce cas qu'une règle de renvoi soit définie au niveau de l'ancienne position sinon le paquet est perdu ;
- Par rapport à l'adressage utilisant des informations de localisation absolue, la mobilité et la fusion sont plus complexes à cause de l'acquisition d'adresse uniquement à partir de nœuds déjà configurés. Nous avons vu qu'en cas de changement de position, un nœud acquiert une nouvelle adresse après échanges de messages avec ses nouveaux voisins. Ainsi, plus les nœuds sont mobiles, plus le trafic de contrôle associé à la reconfiguration des adresses est important car l'acquisition d'une adresse nécessite toujours des échanges de paquets avec le voisinage. La propriété de passage à l'échelle de cette proposition se perd donc avec la mobilité.
- Les techniques de localisation relative ont des précisions limitées. Les erreurs de positionnement peuvent s'accumuler puisque la position d'un nouveau nœud est définie par rapport à celle d'un nœud précédent dont la propre position a été définie par rapport à un autre et ainsi de suite. Il y a donc un risque d'incohérence des adresses par rapport au plan d'adressage.

#### 4.5.6 Simulation

Dans cette sous-section, nous nous intéressons à l'évaluation quantitative de la capacité du protocole à passer à l'échelle . Pour cela, nous avons développé un modèle de simulation pour l'outil ns-3 [82] (ns-3.22).

#### Scénario et métriques

Nous considérons le cas où tous les nœuds d'un réseau connexe s'allument en même temps. Ce scénario tend à rallonger le délai de configuration des nœuds, c'est-à-dire, le temps nécessaire pour qu'un nœud acquiert une adresse valide au sens du plan d'adressage. Nous avons vu que plus il y a de nœuds déjà configurés dans le réseau, plus ce délai est court grâce au concept d'AA. Dans le cas d'un allumage simultané, dans un premier temps chaque nœud pense être seul car aucun AA, ni

nœud standard susceptible de se transformer en AA n'existe dans son voisinage. Il crée donc son propre réseau en s'allouant l'adresse 00 tout en devenant AA et en choisissant un identifiant de réseau aléatoire. Peu après le démarrage des nœuds, un nombre important de fusions est déclenché car les nœuds voisins possèdent des identifiants de réseau différents. De nombreuses reconfigurations d'adresses s'effectuent et à la fin de cet état transitoire, tous les nœuds devraient partager le même identifiant de réseau et avoir des adresses correctes au sens du plan d'adressage.

Nous proposons d'étudier l'influence de cet état transitoire sur le passage à l'échelle à travers deux métriques :

- Le *temps de convergence* qui correspond au temps s'écoulant entre l'allumage des nœuds et l'acquisition d'une adresse correcte par le dernier nœud. En d'autres termes, il s'agit du temps de latence maximal pour acquérir une adresse valide au sens du plan d'adressage à l'échelle du réseau dans le cas d'un allumage simultané de tous les nœuds ;
- Le *nombre de paquets de configuration* qui correspond au nombre de paquets échangés entre les nœuds durant la phase de configuration, c'est-à-dire, jusqu'à la convergence. Il s'agit des requêtes d'adresse, AA ainsi que les réponses correspondantes. Les paquets « HELLO » ne sont pas comptabilisés car ceux-ci sont envoyés périodiquement avant ou après la convergence.

Nous considérons un réseau statique, c'est-à-dire, avec des nœuds immobiles. En effet, il n'y a pas de convergence dans le cas où les nœuds sont mobiles.

### Paramètres

Les métriques sont mesurées pour des nombres croissants de nœuds allant de 100 à 1000. Les nœuds distribués de manière uniforme aléatoire dans une région carrée avec une densité de 0.375 nœuds par unités  $\times$  unités. Chaque nœud a une portée de 3 unités. Ces valeurs assurent la connexité du réseau. La densité est gardée constante pour toutes les simulations. Le modèle wifi configuré dans ns-3 est le « 802.11b non-Qos MAC », avec « rate control » désactivé et mode ad hoc activé. Le protocole de transport choisi pour l'acheminement des paquets de configuration est UDP. Toutes les moyennes indiquées dans la suite sont présentées avec leurs intervalles de confiance à 95% respectives.

### Résultats

La figure 4.8 montre l'évolution du temps de convergence moyen pour différents nombres de nœuds  $n$  variant de 100 à 1000. Nous constatons que le délai de configuration de tous les nœuds est logarithmique avec  $n$ . Nous illustrons ce résultat en dessinant la fonction  $O(\log(n))$  fit sur la figure qui est une fonction logarithmique de  $n$  modulée pour suivre les valeurs issues des simulations. La principale raison de ce résultat vient de la configuration des nœuds par leurs voisinages. Une

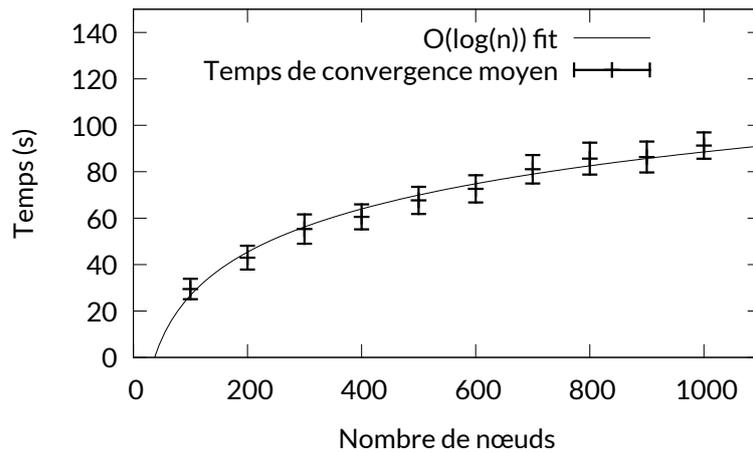


FIGURE 4.8 – Temps de convergence.

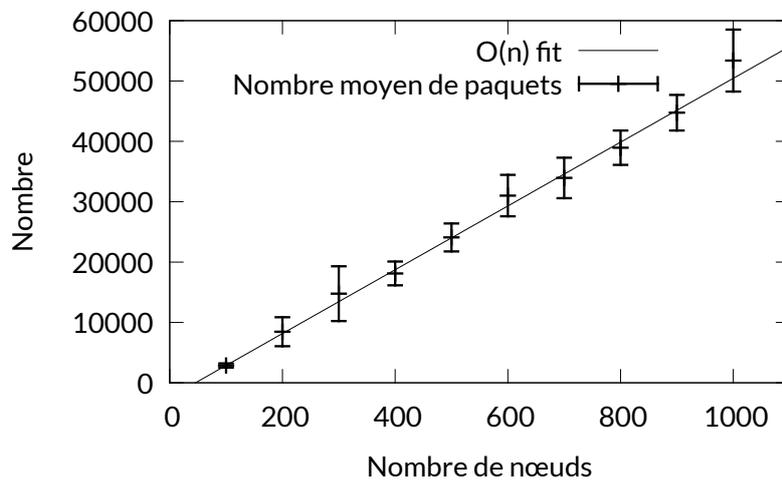


FIGURE 4.9 – Paquets de contrôle.

fois configuré, un nœud peut à son tour configurer les autres nœuds dans son voisinage. Ainsi, le nombre de nœuds « configurables » augmente exponentiellement avec le nombre de nœuds déjà configurés. Et plus il y a de nœuds déjà configurés, plus rapide est la configuration des autres nœuds. Par ailleurs, dans le cas de l'allumage simultané, le nombre de nœuds configurés augmente exponentiellement avec le temps même si cette augmentation est ralentie par les fusions nécessitant des reconfigurations d'adresses.

Sur la figure 4.9, nous constatons qu'en moyenne, le nombre total de paquets de contrôle, c'est-à-dire, les requêtes d'adresse, les requêtes AA et les réponses, générés jusqu'à la convergence augmente linéairement avec le nombre de nœuds  $n$  dans le réseau. Sur la même figure nous avons la fonction  $O(n)$  fit qui indique l'évolution linéaire de la courbe. Ce résultat est logique car plus il y a de nœuds à configurer,

plus il y a de paquets de configuration envoyés. Par ailleurs cette linéarité montre que le nombre moyen de paquets de contrôle nécessaire pour configurer un nœud ne dépend pas du nombre total de nœuds dans le réseau.

Le temps de convergence en  $O(\log(n))$  et le nombre de paquets de configuration en  $O(n)$  montrent la capacité du protocole à passer à l'échelle car le délai de configuration et la bande passante consommée par le protocole n'explorent pas avec le nombre de nœuds dans le réseau.

## 4.6 Conclusion

Les contributions apportées dans ce chapitre répondent aux besoins évoqués au chapitre précédent. Il s'agit, d'une part, d'une méthode de partitionnement de la zone de déploiement du réseau, et d'autre part, d'un mode d'adressage des nœuds permettant leur agrégation géographique.

Nous avons proposé un partitionnement récursif du plan suivant un modèle de *quadtrees* : une région est divisée récursivement en quatre parties égales. Le résultat est un ensemble de régions hiérarchiques. Les régions issues du partitionnement sont identifiées à l'aide d'un numéro. Les numéros sont conçus pour traduire l'organisation hiérarchique des régions tout en renseignant sur leurs localisations dans l'ensemble. Par la suite, nous avons proposé d'utiliser les numéros de région en tant que plan d'adressage afin de fournir aux nœuds des adresses de nature à la fois hiérarchique et géographique. Pour attribuer les adresses, nous proposons deux mécanismes d'adressage selon la nature des informations de localisation accessibles aux nœuds. D'une part, si les nœuds ont à disposition des informations de localisation absolue de type GPS, ils identifient les régions dans lesquelles ils se trouvent et utilisent leurs numéros comme adresses. Dans ce cas, il n'y pas d'échanges nécessaires entre les nœuds. D'autre part, si les nœuds n'ont à disposition que des informations de localisation relative, les adresses sont attribuées suivant un protocole s'appuyant sur des échanges à un saut. L'évaluation de ce protocole montre sa capacité à monter à l'échelle pour un réseau stationnaire. Par contre, cette propriété se perd avec la mobilité à cause des fréquentes reconfigurations d'adresses. De plus, le protocole est exposé à l'accumulation des erreurs de localisation. Les études concernant ce protocole d'adressage par localisation relative n'ont donc pas été poussées plus loin.

Les mécanismes développés dans ce chapitre, à savoir le partitionnement et l'adressage, seront utilisés pour mettre en œuvre notre proposition de protocole de routage à vecteurs de distance hiérarchique géographique. Dans la suite, nous avons décidé de s'appuyer uniquement sur l'adressage par localisation absolue qui présente le moins d'inconvénients dans la mise en pratique.

## Chapitre 5

# Routage à vecteurs de distance hiérarchique géographique avec zones d'agrégation

### 5.1 Introduction

Nous avons introduit un concept de routage à vecteurs de distance géographique au chapitre 3. Il a été conclu que pour mettre en œuvre les différents mécanismes de cette proposition, il est nécessaire de développer deux principes : d'une part, un mode de partitionnement de la zone de déploiement du réseau, et d'autre part, un système d'adressage des nœuds permettant l'agrégation de ces derniers. Nous avons donc proposé, au chapitre 4, un partitionnement hiérarchique avec numérotation des régions. Les numéros des régions sont ensuite utilisés comme adresse des nœuds qui y sont localisés. Dans ce contexte, une adresse peut représenter une agrégation de plusieurs nœuds. Dans ce chapitre, nous allons mettre en œuvre ces principes pour appliquer notre idée de routage à vecteurs de distance hiérarchique géographique avec zones d'agrégation.

Dans un premier temps, nous allons expliciter les problématiques et motivations qui nous ont amené à développer cette proposition. Par la suite, nous allons, à partir du concept de routage dans les réseaux ad hoc présenté dans le chapitre 3, détailler comment obtenir un protocole fonctionnel avec les deux principes cités précédemment. Nous terminerons ce chapitre par l'analyse des résultats des évaluations par simulation du protocole proposé.

### 5.2 Problématiques et motivations

Nous n'allons pas reprendre tous les éléments justifiant notre proposition de protocole de routage à vecteurs de distance géographique faite dans le chapitre 3 mais nous soulignerons surtout l'importance de l'introduction de la hiérarchisation dans cette approche pour les aspects de passage à l'échelle et de mobilité.

Nous rappelons ici l'hypothèse posée au chapitre 3 selon laquelle agréger les informations concernant des groupes de nœuds, d'une part, rend les protocoles de routage classiques insensibles au facteur d'échelle avec des performances comparables, et d'autre part, constitue un moyen d'améliorer les routages géographiques en évitant les techniques comme la planarisation. L'objectif de ce chapitre est de vérifier cette hypothèse en mettant en œuvre notre approche et en analysant les résultats des simulations évaluant la faculté de notre protocole de routage à passer à l'échelle et sa capacité à fonctionner dans un contexte de mobilité.

Notre proposition s'appuie donc sur une agrégation d'informations relatives à des nœuds proches géographiquement sous forme de zones d'agrégation. Plus précisément, on considère plusieurs niveaux d'agrégation correspondant à des zones d'agrégation plus ou moins étendues. En d'autres termes, il existe une hiérarchie des zones d'agrégation d'où l'exploitation du partitionnement hiérarchique développé au chapitre précédent. Le choix d'une hiérarchisation est justifié, entre autres, par le constat que celle-ci a toujours fait partie des moyens utilisés pour permettre le passage à l'échelle [85]. Mais celle-ci nécessite souvent une construction et une maintenance coûteuses surtout dans un contexte de mobilité. En s'appuyant sur les informations de nature géographique, nous pouvons mettre en place et maintenir une hiérarchie de régions sans intervention des nœuds. Cette hiérarchie est ensuite exploitée par notre proposition de routage pour accroître sa capacité à monter à grande échelle.

Par ailleurs, pour faire face au caractère dynamique des réseaux ad-hoc, l'auto-organisation des nœuds constitue une approche sérieuse [14]. Nous mettons en œuvre cette idée à travers la structuration des nœuds avant toute opération de routage. Cette structuration préalable apporte de la stabilité, utile pour gérer la mobilité. En effet, à titre d'exemple, la mobilité des nœuds à l'intérieur d'une zone d'agrégation est transparente aux nœuds à l'extérieur, ne déclenchant pas des opérations de maintenance comme la mise à jour des tables, consommatrices de la bande passante.

### 5.3 Mise en œuvre générale

Il s'agit d'appliquer l'idée de protocole de routage à vecteurs de distance décrite au chapitre 3 en utilisant le mode de partitionnement de l'espace en régions hiérarchiques ainsi que l'adressage des nœuds vus au chapitre 4.

Dans un premier temps, la zone de déploiement du réseau est partitionnée en régions hiérarchiques. Ensuite un nœud définit deux types de zones :

- Une zone *locale* qui est définie par un ensemble de régions entourant la région élémentaire dans laquelle celui-ci se trouve. Les informations concernant les nœuds à l'intérieur de cette zone ne sont pas agrégées. Autrement dit, notre

nœud enregistre les routes individuelles vers chaque nœud situé dans la zone locale. Chaque nœud situé dans cette dernière correspond à une entrée dans la table de routage ;

- Plusieurs zones d'agrégation qui correspondent chacune à une région issue du partitionnement préalable et sélectionnée de telle manière que plus une zone d'agrégation est éloignée géographiquement, plus elle est de grande taille. Une zone d'agrégation résume les informations concernant l'ensemble des nœuds à l'intérieur de celle-ci. Une zone d'agrégation correspond à une entrée dans la table de routage.

Les entrées des tables de routage correspondent à des vecteurs de distance vers des nœuds ou zones. Les tables sont mises à jour par des échanges directs, c'est-à-dire, entre des nœuds voisins. L'acheminement des paquets s'appuie sur la lecture des tables de routage en considérant des aspects géographiques. Ces caractéristiques nous ont conduits à nommer notre protocole *Hierarchical Geographical Distance Vector routing protocol* (HGDV) pour la suite.

## 5.4 Hypothèses faites par la suite

Nous faisons les hypothèses suivantes :

- Les nœuds ont accès à leurs coordonnées géographiques par l'intermédiaire d'un GPS ou tout autre système de positionnement similaire ;
- Avant toute opération, la zone de déploiement du réseau est partitionnée selon le modèle détaillé au chapitre 4. L'algorithme de partitionnement est exécuté par tous les nœuds avec les mêmes données initiales que sont : la taille de la zone à partitionner et la taille des régions élémentaires de sorte que tous les nœuds aboutissent exactement au même résultat ;
- Chaque nœud acquiert une adresse en utilisant ses coordonnées géographiques et le résultat du partitionnement du réseau ;
- Un service de localisation se charge de la résolution des identifiants en adresse. Ce service est utilisé pour localiser le destinataire d'une communication. [81] propose un état de l'art des services de localisation pour les réseaux ad-hoc. Notons qu'il est possible d'intégrer le service de localisation avec notre système de régions hiérarchiques issues du partitionnement comme dans *Grid Location Service* (GLS) [95]. Il s'agit de faire en sorte que le service de localisation s'appuie sur notre mode de partitionnement du réseau dans ses mécanismes afin d'en tirer des avantages en termes de simplicité par exemple.

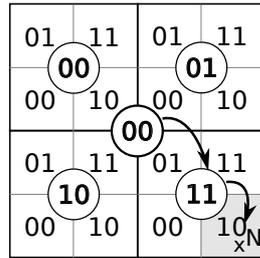


FIGURE 5.1 – Rappel sur le partitionnement et l’adressage.

## 5.5 Rappel sur le partitionnement et l’adressage des nœuds et des zones d’agrégation

Nous faisons un rappel du partitionnement à l’origine de l’adressage des nœuds et des zones d’agrégation qui relèvent d’une grande importance par la suite.

Nous précisons d’abord que « zone d’agrégation » est un terme générique introduit au chapitre 3 pour désigner une zone géographique agréant les nœuds qui s’y trouvent et qui est une destination de route. Dans ce chapitre, nous utilisons les régions issues du partitionnement développé au chapitre précédent en tant que zones d’agrégation.

La zone de déploiement du réseau est partitionnée de manière récursive en régions. A chaque itération, chaque région est divisée en quatre sous-régions de niveau inférieur numérotées de 00 à 11 en binaire. Il s’agit de numéros *locaux*. Le numéro *global* d’une région est la concaténation des numéros locaux des régions contenant celle-ci, par ordre de niveau décroissant. Les numéros des régions élémentaires, c’est-à-dire, les plus petites régions sont utilisés en tant qu’adresses des nœuds qui y sont situés.

Sur la figure 5.1, nous avons un partitionnement à trois niveaux. La région élémentaire en gris, de niveau 1 a comme numéro local 10. Elle est contenue dans la région de niveau 2, de numéro local 11, elle-même appartenant à la région de niveau 3, 00. Le numéro local de la région grisée est donc 001110. Ainsi le nœud N qui est situé dans cette région a comme adresse 001110.

### 5.5.1 Choix de la taille d’une région élémentaire

Nous proposons de choisir la taille d’une région élémentaire de telle sorte que les nœuds appartenant à une même région élémentaire sont à portée les uns des autres. En effet il est plus cohérent par rapport au routage de faire en sorte que deux nœuds ayant la même adresse puisse communiquer entre eux. L’adresse représente la destination finale d’un paquet et arrivé à destination celui-ci ne devrait plus faire l’objet

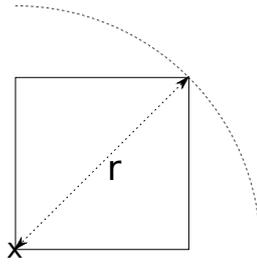


FIGURE 5.2 – Choix de la taille d'une région élémentaire.

d'un routage. Mais si deux nœuds appartenant à la même région élémentaire ne sont pas à portée l'un de l'autre, le routage entre les deux reste possible à condition que l'un se trouve dans la zone locale de l'autre. En considérant que les nœuds ont la même portée  $r$ , il faut que la distance maximale séparant deux nœuds appartenant à une région élémentaire soit égale à cette portée. Cette distance maximale correspond à la diagonale du carré représentant la région qui doit donc avoir comme longueur  $r$ , c'est-à-dire, un côté égal à  $\frac{r}{\sqrt{2}}$  (figure 5.2).

Notons que tous les nœuds considèrent la même taille de région élémentaire pour exécuter l'algorithme de partitionnement même s'ils ont des portées différentes afin d'arriver au même partitionnement. Nous verrons dans les simulations l'influence de ce paramètre sur les performances du protocole.

## 5.6 Définition des zones à partir du partitionnement du plan

Au chapitre 3 nous avons proposé deux types de zones utilisées pour les opérations du protocole : la zone locale et la zone d'agrégation. Cette section détaille comment définir précisément ces zones au travers de notre partitionnement hiérarchique de l'espace de déploiement des nœuds. Ces deux zones étant propres à chaque nœud, nous décrirons la suite du point de vue d'un nœud.

### 5.6.1 Zone locale et paramètre $k$

Pour rappel, la zone locale d'un nœud est une région géographique d'un certain rayon dont celui-ci est le centre. Pour la mise en œuvre, nous définissons la zone locale comme l'étendue géographique de rayon «  $k$  régions élémentaires » entourant la région élémentaire du nœud considéré.  $k$  est un paramètre du protocole qui a pour rôle de moduler l'étendue de la zone locale pour chaque nœud. Sur la figure 5.3, nous avons deux exemples de zones locales du nœud  $N$  pour  $k = 1$  et  $k = 2$ , en gris et entourées de tiret discontinu.

Un nœud retient les routes vers tous les autres nœuds à l'intérieur de sa zone locale. Ainsi, plus cette zone est large, plus le nombre d'entrées dans les tables de

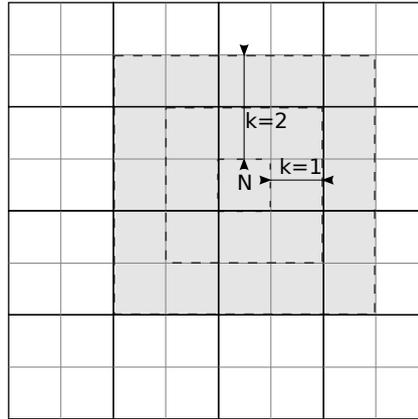


FIGURE 5.3 – Zone locale.

routage est grand. Des simulations ont été conduites dans le but de rechercher la valeur optimale du paramètre  $k$  et surtout d'étudier son influence sur les performances du protocole.

### 5.6.2 Zones d'agrégation et paramètre $m$

L'ensemble des zones d'agrégation d'un nœud correspond à une étendue géographique entourant ce dernier. Nous voulons faire en sorte que ces zones aient comme propriété de s'agrandir avec l'éloignement. Nous avons vu que notre partitionnement produit des régions hiérarchiques dont les étendues sont directement liées avec leur niveau dans la hiérarchie. Ainsi, pour avoir des zones d'agrégation de plus en plus larges avec la distance, il suffit, pour un nœud, de sélectionner des régions hiérarchiques de niveaux de plus en plus élevés avec l'éloignement. En d'autres termes, une zone d'agrégation correspond à une région issue de notre partitionnement hiérarchique. En conséquence, un nœud considère des zones d'agrégation de différents niveaux. Plus précisément, nous définissons donc une zone d'agrégation de niveau  $n$  comme une région de niveau  $n$  adjacente à la région de niveau  $n$  à laquelle appartient le nœud considéré. Au chapitre 3, nous avons utilisé le terme général « ordre » pour indiquer la place d'une zone dans la hiérarchie. En utilisant notre mode de partitionnement, l'équivalent de l'ordre d'une zone est le niveau de la région correspondante.

Sur la figure 5.4, nous avons les zones d'agrégation considérées par le nœud  $N$ . Nous avons noté le niveau de chaque zone. Nous remarquons que plus l'éloignement à  $N$  est important, plus la zone d'agrégation correspondante est grande et de niveau élevé. Par ailleurs, pour chaque niveau, un nœud considère au minimum trois et au maximum huit zones d'agrégation.

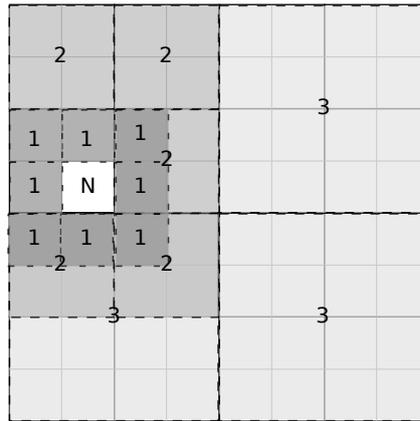


FIGURE 5.4 – Zones d'agrégation.

Une zone d'agrégation représente tous les nœuds qui y sont situés géographiquement. En d'autres termes, elle agrège de manière géographique les nœuds et, par extension, les routes vers ces derniers. Ainsi, dans la table de routage, en dehors de la zone locale, une entrée ne correspond plus à une route vers un nœud mais vers une zone d'agrégation contenant un ou plusieurs nœuds.

Comme pour la zone locale, il est aussi intéressant de contrôler la zone d'influence des zones d'agrégation : nous voulons définir une limite au-delà de laquelle les informations de routage ne sont plus prises en compte même sous forme agrégées. Nous introduisons donc le paramètre  $m$  qui est le niveau maximum des zones d'agrégation prises en compte. Puisque le niveau d'une zone d'agrégation traduit son étendue, plus  $m$  est petit, plus la surface totale de l'ensemble des zones d'agrégation est petite. Sur la figure 5.5, nous montrons l'étendue de cette surface, colorée en gris, pour  $m = 2$  (5.5a) et  $m = 3$  (5.5b). Nous évaluerons avec des simulations l'impact de ce paramètre sur les performances du protocole.

## 5.7 Structure des tables

La table de routage est l'outil principal utilisé par chaque nœud pour décider du prochain saut pour l'acheminement des paquets. Pour rappel, elle contient les informations de routes, d'une part, vers les nœuds situés dans la zone locale, et d'autre part, vers ceux situés dans les zones d'agrégation sous-forme agrégée. Nous proposons ici une structure de table prenant en compte les précédentes définitions de la zone locale et des zones d'agrégation. Par ailleurs, la table de routage contient aussi les informations nécessaires pour la gestion de la mobilité. La table est initialisée à vide et se remplit au fur et à mesure des mises à jours selon un mécanisme que nous décrirons plus bas.

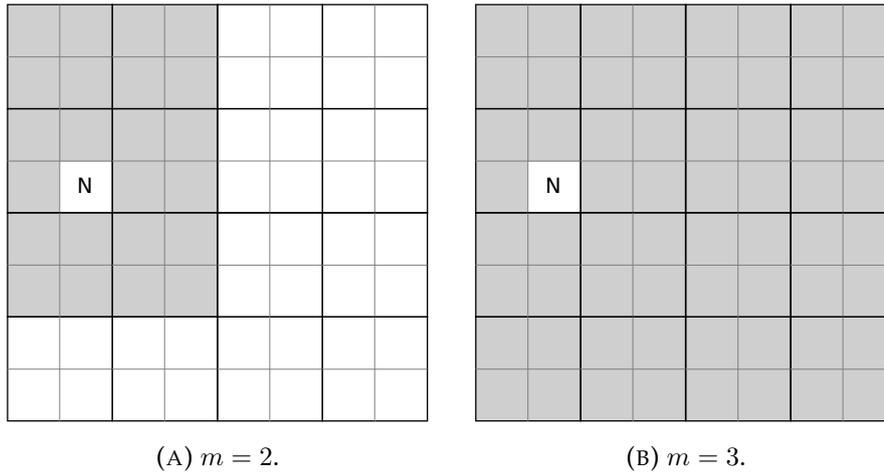


FIGURE 5.5 – Ensemble des zones d'agrégation de tous les niveaux en fonction de  $m$ .

La structure d'une table est présentée sur le tableau 5.1. Nous avons séparé par un trait la partie supérieure correspondant à la zone locale et la partie inférieure correspondant à l'ensemble des zones d'agrégation. Chaque ligne de la table représente une route. Passons en revue les différents champs constituant cette dernière :

- Le champ *Zone* ( $A$ ) sert à reconnaître si une route concerne un nœud de la zone locale (0) ou une zone d'agrégation. Dans le deuxième cas, la valeur de cette colonne indique le niveau de la zone d'agrégation concernée. Ainsi la valeur maximale de cette colonne correspond au paramètre  $m$  ;
- Le champ *Destination* ( $D$ ) renseigne la destination joignable. Il peut s'agir d'un nœud situé dans la zone locale, auquel cas deux informations sont fournies : l'adresse de  $D$  au sens du plan d'adressage et son identifiant unique  $ID$ . Dans le cas où la destination correspond à une zone d'agrégation, seule l'adresse est importante étant donnée que l'adresse localise et identifie de manière unique une région. La partie identifiant peut être renseignée par une valeur par défaut que nous allons noter  $AREA$  dans la suite ;
- Le champ *Passerelle* ( $GW$ ) indique le nœud voisin par lequel un paquet à destination de  $D$  doit être transféré. Comme pour le champ  $D$ , il est mentionné l'adresse et l'identifiant  $ID$  du nœud passerelle ;
- Le champ *Coût* ( $C$ ) correspond au coût nécessaire pour atteindre  $D$  avec la présente route. Nous avons choisi comme métrique le nombre de sauts même s'il est évidemment envisageable d'en utiliser une autre ;
- Le champ *Numéro de séquence* ( $S$ ) est utile pour gérer la mobilité. Dans un contexte de mobilité, la table de routage est aussi dynamique avec des entrées devant être supprimées en fonction des changements dans la topologie. Les numéros de séquence servent à enregistrer les mises à jour les plus récentes dans les tables. Surtout, ils permettent de surmonter le problème du

TABLE 5.1 – Structure d’une table de routage.

Zone (A)	Destination (D)		Passerelle (GW)		Coût (C)	Numéro de séquence (S)	Installation (I)
0	adresse	ID	adresse	ID	sauts	numéro	instant
0	adresse	ID	adresse	ID	sauts	numéro	instant
1	adresse		adresse	ID	sauts	numéro	instant
2	adresse		adresse	ID	sauts	numéro	instant
2	adresse		adresse	ID	sauts	numéro	instant
...							
m	adresse		adresse	ID	sauts	numéro	instant
m	adresse		adresse	ID	sauts	numéro	instant

« comptage à l’infini » [17], caractéristique des protocoles de routage à vecteurs de distance, dans lequel le coût associé à une route augmente petit à petit jusqu’à l’infini alors que la destination associée n’existe plus ;

- Le champ *Installation* (*I*) indique l’instant où l’entrée a été ajoutée ou mise à jour. La valeur de cette colonne sert à vieillir les routes afin de supprimer celles qui n’ont plus été mises à jour depuis longtemps.

Dans la suite, nous allons écrire une entrée de la table de la manière suivante : {A, D.addr, D.id, GW.addr, GW.id, C, S, I}. Nous utiliserons aussi la notation pointée pour accéder à la valeur d’un champ d’une entrée. Par exemple, pour une route *R*, la valeur du champ *zone* est *R.A*.

## 5.8 Maintenance des tables

### 5.8.1 Principe

Les tables sont mises à jour périodiquement. Les nœuds s’échangent de manière régulière les contenus de leur table par l’intermédiaire de paquets contenant les informations de routes. La fréquence des envois de ces paquets de mise à jour est un paramètre du protocole. Plus les mises à jour sont fréquentes, plus les routes enregistrées sont fiables mais plus la consommation en bande passante est élevée. Il y a donc un compromis à trouver.

#### Gestion de la mobilité

A cause de la mobilité des nœuds, les routes enregistrées dans les tables peuvent se périmérer. C’est le cas d’un nœud qui s’est déplacé et qui est devenu hors de portée.

De plus, les informations de routes reçues peuvent ne plus être à jour, compromettant le contenu des tables si elles sont prises en compte ou propagées. Pour gérer ces deux situations, deux champs de la table de routage sont exploités : l'*installation* et le *numéro de séquence*.

La durée de vie de chaque entrée d'une table de routage est vérifiée périodiquement. En général, la vérification est faite à chaque envoi de paquets de mise à jour. Si elle dépasse un certain seuil, cela signifie qu'elle n'a plus été mise à jour et n'est plus fiable. Elle doit être supprimée. Le seuil est généralement défini comme un multiple de la période de mise à jour des tables afin d'exclure les effets des pertes de paquets à travers le canal de communication.

La durée de vie d'une entrée indique sa « fraîcheur » du point de vue locale, c'est-à-dire, par rapport au nœud qui maintient la table. Mais pour les informations de routes circulant sur le réseau, leur nouveauté est contrôlé par l'intermédiaire de leurs numéros de séquence. Le numéro de séquence est un entier positif associé à une route pour une destination donnée. C'est le nœud créateur de l'information de route qui l'initialise et l'incrémente à chaque envoi de mises à jour. Le nœud créateur indique aussi son identifiant dans le champ destination. Le numéro de séquence est gardé quand l'information de route est agrégée, c'est-à-dire, quand la destination est agrégée en une zone d'agrégation. Pour une destination donnée, un nœud ne prend en compte une mise à jour de route que si le numéro de séquence associé est supérieur ou égal à celui de l'entrée enregistrée dans sa table. Le numéro de séquence sert aussi à distinguer les informations de routes valides de celles qui sont périmées. Des numéros de séquences pairs sont assignés aux routes valides. Ainsi, à chaque envoi d'une mise à jour, un nœud incrémente de 2 le numéro de séquence de la route dont il est l'initiateur, les autres nœuds ne modifiant pas celui-ci. Par contre, les mises à jour correspondant à des routes périmées ont des numéros de séquence impairs. Si un nœud constate qu'une entrée dépasse le seuil de durée de vie maximale, il incrémente de 1 le numéro de séquence associé. A la réception d'une mise à jour avec un numéro de séquence impair, un nœud propage immédiatement l'information. Cela permet d'accélérer la dissémination des informations de routes périmées, améliorant la fiabilité des routes enregistrées dans les tables.

### 5.8.2 Contenu des informations de route

Pour communiquer une information de route, seuls les éléments suivants sont nécessaires :

- Zone indiquant d'une part la nature de la destination (nœud ou zone d'agrégation) et le niveau s'il s'agit d'une zone d'agrégation ;
- Adresse et identifiant de la destination ;

- Adresse de la passerelle. Pour un nœud recevant une mise à jour, l'identifiant de la passerelle est celle du nœud par lequel le paquet a été reçu ;
- Coût ;
- Numéro de séquence.

Une information de route a donc la forme  $\{A, D.addr, D.id, GW.addr, C, S\}$ . Par convention, nous allons remplir par AREA le champ D.id d'une information de route vers une zone d'agrégation. En effet, leurs adresses suffisent à les différencier.

### 5.8.3 Initialisation

Un nœud, notons  $N$ , initialise sa table de routage avec ses propres informations :  $\{0, N.addr, N, N.addr, N, 0, 0, inf\}$ . Le champ installation a été mis à l'infini car cette entrée ne se périmera jamais.

### 5.8.4 Algorithme

Afin de traiter les mises à jour de route, chaque nœud exécute l'algorithme 5.1 où  $T$  est la table de routage du nœud recevant la mise à jour et  $R$  l'entrée créée à partir de cette dernière. Par ailleurs, nous notons par  $ZL$  la zone locale et  $\{ZA_n\}$  l'ensemble des zones d'agrégation de niveau  $n$ .

```
Recevoir (R)
{
  Pour i allant de R.A à m
  {
    # Si R se réfère à un nœud appartenant à la zone locale
    Si (i == 0 ET R ∈ ZL)
      Traiter (R)
    # Si R se réfère à une zone d'agrégation
    Si (i != 0)
    {
      r = R
      r.A = i
      # Calculer de la zone d'agrégation d'ordre i
      Si (R.A == 0)
        r.D.addr = r.D.addr >> 2 * (i - 1)
      Sinon
        r.D.addr = r.D.addr >> 2 * (i - R.A)
      # Si r fait partie des zones d'agrégation de niveau i
      # à considérer par le nœud courant
      Si (r ∈ {ZAi})
        Traiter (r)
    }
  }
}
```

```

Traiter (R)
{
  # Si la route n'existe pas encore dans la table
  Si (R  $\notin$  T)
  {
    Si (R.S est pair)
    {
      R.C++
      Ajouter R dans T
    }
  }
  Sinon
  {
    Soit R_existant  $\in$  T tel que R_existant.D == R.D

    Si (R.S est pair)
    {
      Si (R.A == 0)
      {
        # Si on a un numéro de séquence plus récent ou
        # une route avec un coût plus faible
        Si (R.S > R_existant.S OU (R.S == R_existant ET R.C < R_existant.C))
          Remplacer R_existant par R

        # Si le nœud s'est déplacé et par conséquent a changé d'adresse
        Si (R.D.ID == R_existant.D.ID ET R.D.addr != R_existant.D.addr)
        {
          Remplacer R_existant par R
          # Supprimer toutes les routes existantes dont le nœud
          # est la passerelle
          Supprimer (r) avec r  $\in$  T tel que r.GW = R_existant.D
        }
      }
      Sinon
      {
        Si (R.C < R_existant.C)
          Remplacer R_existant par R
      }
    }

    Si (R.S est impair ET R.S > R_existant.S)
      Supprimer (R_existant)
  }
}

```

```

Supprimer (R)
{
  Si (R.A == 0)
  {
    Effacer R de  $T$ 
    Propager R avec R.S++
    Supprimer (r) avec  $r \in T$  tel que  $r.GW = R\_existant.D$ 
  }
  Sinon
  {
    # L'entrée est maintenue mais réactualisée si au moins une entrée
    # de niveau inférieur pouvant être agrégée au sein de la zone
    # considérée existe.
    # Pour rappel, l'existence d'une entrée de niveau  $i$ 
    # implique celles de toutes les entrées de niveau supérieur
    # à  $i$  jusqu'à  $m$  (c.à.d de toutes les zones d'agrégation de niveau
    # de plus en plus grand contenant la zone considérée) dans la
    # table de routage
    Si (  $\exists r_2 \in T, \exists j, R.D.addr = r.D.addr \gg 2j$  )
      R.GW = r.GW
    Sinon
    {
      Effacer R de  $T$ 
      Propager R avec R.S++
    }
  }
}

```

ALGORITHME 5.1 – Algorithme de traitement des mises à jour de route.

Dans un premier temps, une entrée de table potentielle est générée à partir de l'information de route reçue. Cette route a comme *destination* un nœud ou une zone d'agrégation. Selon notre mode de partitionnement hiérarchique, une région, sauf celle du niveau le plus haut, est contenue dans une ou plusieurs régions de niveaux supérieurs. Ainsi, à partir de la route précédente, plusieurs entrées sont créées avec comme destinations toutes les régions de niveaux supérieurs jusqu'à  $m$  contenant la destination initiale qui peut être un nœud ou une zone d'agrégation. Notons que pour trouver l'adresse de ces régions de niveau supérieur, il suffit d'effectuer plusieurs décalages binaires sur l'adresse initiale. Ensuite, l'appartenance de toutes ces routes à la zone locale ou aux zones d'agrégation est vérifiée. Si la destination est un nœud appartenant à la zone locale, l'entrée est traitée. Si la destination est une zone d'agrégation de niveau  $m$  au maximum, l'entrée est traitée uniquement si elle appartient à l'une des zones d'agrégation de même niveau considérées par le nœud courant (voir figure 5.5). Les entrées qui ne respectent pas ces conditions sont supprimées.

Le traitement d'une entrée consiste à vérifier dans un premier temps si la table contient déjà une route vers la même destination. Si la réponse est négative, la nouvelle entrée est inscrite dans la table de routage à condition que son numéro de

séquence soit pair. Si la réponse est positive, deux cas se présentent : soit le numéro de séquence est pair, soit le numéro de séquence est impair.

Dans le cas où le numéro de séquence est pair, il y a, de nouveau, deux cas :

- Si l'entrée reçue correspond à un nœud, elle remplace l'entrée dans la table si son numéro de séquence est plus grand. Elle la remplace également si son numéro de séquence est identique à celle de l'entrée existante mais que le coût qui lui est associé est inférieur. Par ailleurs, si l'entrée reçue correspond à un nœud présent dans la table mais avec une adresse différente, le nœud associé a donc changé de position. Il est donc nécessaire que les routes ayant ce nœud comme passerelle soient supprimées. En effet la passerelle ayant changée, les destinations peuvent ne plus être accessibles ;
- Si l'entrée reçue correspond à une zone d'agrégation, elle remplace l'entrée dans la table si le coût qui lui est associé est inférieur à celui de l'entrée existante. La route vers une zone d'agrégation hérite du numéro de séquence d'une route vers un nœud qui a été agrégé. Ainsi, il est inutile de comparer les numéros de séquence de deux routes vers une même zone d'agrégation car ces numéros peuvent provenir de deux agrégations différentes. Par contre, il est toujours utilisé pour indiquer la validité d'une route à travers sa parité.

Dans le cas où le numéro de séquence est impair, il s'agit d'une information indiquant une route qui n'est plus valide. Si le numéro de séquence est plus grand que celui de l'entrée existante, il faut appliquer la procédure de suppression d'entrée sur cette dernière.

La procédure de suppression d'entrée dépend de la nature de la destination : nœud ou zone d'agrégation.

- Si l'entrée à supprimer a comme destination un nœud :
  - Elle est supprimée et une mise à jour de route correspondant avec un numéro de séquence incrémenté de 1 est propagé immédiatement ;
  - De plus, on applique la procédure de suppression à toutes les entrées ayant comme passerelle la destination de cette entrée.
- Si l'entrée à supprimer a comme destination une zone d'agrégation et que la passerelle de celle-ci est le nœud par lequel la mise à jour de route a été reçue :
  - On cherche une entrée existante dont la destination peut être agrégée en la zone d'agrégation de l'entrée à supprimer. En d'autres termes, on cherche une entrée dans la table dont la destination est contenue dans la zone d'agrégation de l'entrée à supprimer ;
  - Si une telle entrée existe, alors la passerelle de celle-ci est utilisée comme nouvelle passerelle de l'entrée à supprimer, de même que le numéro de séquence et le coût. Donc, l'entrée n'est finalement pas supprimée ;

01 11 00 10	01 11 00 10	01 11 00 10	01 11 00 10
01 11 00 10	01 11 00 10	01 11 00 10	01 11 00 10
01 11 00 10	01 11 00 10	01 11 00 10	01 11 00 10
01 11 00 10	01 11 00 10	01 11 00 10	01 11 00 10

Figure 5.6 shows a 4x4 grid of routing table entries. Each entry is a 2x2 sub-table with bits (01, 11) in the top row and (00, 10) in the bottom row. A central 2x2 area is shaded gray. A dashed line separates the top two rows from the bottom two rows. A node 'N' is indicated at the bottom-left of the central area, and a gateway 'GW' is indicated at the top-left of the central area. Various entries are circled: (00) in the top-left, (01) in the top-right, (10) in the middle-left, (11) in the middle-right, (00) in the bottom-left, and (01) in the bottom-right. Labels 'A', 'B', 'x', and 'Nx' are also present near the central area.

FIGURE 5.6 – Exemple de mise à jour de table.

- S’il n’existe aucune entrée respectant ces conditions, alors l’entrée est supprimée et une mise à jour de route avec un numéro de séquence incrémenté de 1 est propagée immédiatement.

### 5.8.5 Exemple

Sur la figure 5.6, le nœud  $N$  reçoit des informations de mise à jour de la part du nœud  $GW$ . Nous supposons que  $k = 1$  et  $m = 2$  et que la table de routage de  $N$  vient juste d’être initialisée et ne contient donc qu’une entrée correspondant à ses propres informations. La zone locale de  $N$  est délimitée par un trait discontinu tandis que l’ensemble de ses zones d’agrégation de niveau 1 et 2 est coloré en gris (voir la figure 5.5 pour les détails). Dans la suite, la fonction `now()` retourne l’instant présent. Le nœud  $N$  reçoit les informations de routes suivantes :

- $\{0, 00101101, A, 00101100, 1, 10\}$ . A partir de cette information, les entrées potentielles générées ont comme destination :
  - Le nœud  $A$  à l’adresse  $00101101$  (destination initiale). Ce nœud se trouve dans la zone d’agrégation, ainsi la route  $\{0, 00101101, A, 00101100, GW, 2, 10, now()\}$  est ajoutée à la table ;
  - La zone d’agrégation  $00101101$  de niveau 1. Cette zone compte parmi les zones d’agrégation de niveau 1 considérées par  $N$ , ainsi la route  $\{1, 00101101, AREA, 00101100, GW, 2, 10, now()\}$  est inscrite dans la table ;
  - La zone d’agrégation  $001011$  de niveau 2. Cette zone fait partie des zones d’agrégation de niveau 2 considérées par  $N$ , ainsi la table est enrichie avec

- la route {2, 001011, AREA, 00101100, GW, 2, 10, now ()}.
- {0, 00101111, B, 00101100, 1, 8}. Les entrées potentielles correspondantes ont comme destination :
    - Le nœud B d'adresse 00101111 (destination initiale). D'après son adresse, ce nœud se trouve en dehors de la zone locale, aussi cette entrée est ignorée ;
    - La zone d'agrégation 00101111 de niveau 1. Cette zone ne fait pas partie des zones d'agrégation de niveau 1 considérées par N. L'entrée n'est donc pas prise en compte ;
    - La zone d'agrégation 001011 de niveau 2. Cette zone fait partie des zones d'agrégation de niveau 2 considérées par N. Dans sa table, ce dernier a une entrée pour la même destination avec un coût égal à 2 qui est donc plus important. Aussi l'entrée est mise à jour avec {2, 001011, AREA, 00101100, GW, 1, 8, now ()};
  - {2, 000010, AREA, 00101100, 4, 10}. L'entrée potentielle correspondante est :
    - La zone d'agrégation 000010 de niveau 2 (destination initiale, zone hachurée sur la figure). Cette zone ne fait pas partie des zones d'agrégation de niveau 2 considérées par N donc l'entrée est ignorée.

Maintenant supposons que B n'est plus joignable. A reçoit donc l'information de route {0, 00101111, B, 00101100, 1, 9}. Les entrées potentielles générées ont comme destination :

- Le nœud B d'adresse 00101111 (destination initiale). D'après son adresse, ce nœud se trouve en dehors de la zone locale, aussi la mise à jour est ignorée ;
- La zone d'agrégation 00101111 de niveau 1. Cette zone ne fait pas partie des zones d'agrégation de niveau 1 considérées par N. La mise à jour n'est donc pas prise en compte ;
- La zone d'agrégation 001011 de niveau 2. Une entrée avec la même destination existe dans la table. La procédure de suppression est appliquée. N cherche donc dans sa table une entrée dont la destination peut être agrégée en cette zone d'agrégation, en d'autres termes, une entrée dont la destination est contenue dans celle-ci. A respecte cette condition. Aussi l'entrée existante est mise à jour pour avoir la même passerelle, le même numéro de séquence et le même coût que l'entrée correspondant à A. Ainsi l'entrée devient {2, 001011, AREA, 00101100, GW, 2, 10, now ()}.

## 5.9 Procédure de routage

### 5.9.1 Principe

L'acheminement d'un paquet de la source à la destination nécessite que chaque nœud intermédiaire choisisse le prochain saut auquel le paquet doit être transféré.

Le routage des paquets résulte uniquement de prises de décision locales. Pour choisir le prochain saut, un nœud se base sur son adresse, celle de la destination et sur le contenu de sa table de routage. L'idée principale est de trouver le nœud ou la zone d'agrégation qui permet de se rapprocher le plus de la destination, ou de la zone d'agrégation à laquelle elle appartient, et de transférer le paquet à la passerelle correspondante. Il s'agit de parcourir la table de routage en comparant chaque entrée avec l'adresse du nœud courant et celle de la destination.

### 5.9.2 Algorithme

Un nœud que nous allons identifier par  $S$  reçoit un paquet à destination de  $D$ . S'il est le destinataire, il n'y a pas de procédure de routage. Sinon, il cherche d'abord dans sa table de routage une entrée dont le champ destination correspond à  $D$ . Si aucune entrée n'est trouvée, le choix du prochain saut s'effectue en suivant l'algorithme 5.2.

```

Soit  $l > 0$  le plus petit niveau tel que
     $S.addr \ll 2*(l-1) == D.addr \ll 2*(l-1)$  # Niveau de la plus grande zone
                                           # d'agrégation contenant  $D$  mais
                                           # qui n'est pas commune à  $S$  et  $D$ 

Si ( $l > m$ )
     $l = m$ 

 $l\_ref = l$ 

Pour  $i$  allant de  $l$  à 1
{
     $D\_courant = D.addr \ll 2*(i-1)$  # Adresse de la zone d'agrégation
                                   # de niveau  $i$  contenant
                                   # la destination
     $D\_ref = D.addr \ll 2*(l\_ref-1)$  # Adresse de la zone d'agrégation
                                   # correspondant à
                                   # la dernière route potentielle trouvée

    Chercher la route  $R$  dont la destination est une zone d'agrégation
    de niveau  $i$  se rapprochant le plus de la zone ayant comme
    adresse  $D\_courant$  et appartenant à la zone d'agrégation
    ayant comme adresse  $D\_ref$ 

    Si ( $R$  existe)
         $l\_ref = i$ 
}

```

ALGORITHME 5.2 – Choix du prochain saut.

Il s'agit dans un premier temps de trouver le niveau  $l$  de la plus grande zone d'agrégation contenant la destination mais ne contenant pas la source avec  $l \leq m$ .

01 11 00 10	01 11 00 10	01 11 00 10	01 11 00 10
01 11 00 10	01 11 00 10	01 11 00 10	01 11 00 10
01 11 00 10	01 11 00 10	01 11 00 10	01 11 00 10
01 11 00 10	01 11 00 10	01 11 00 10	01 11 00 10
01 11 00 10	01 11 00 10	01 11 00 10	01 11 00 10
01 11 00 10	01 11 00 10	01 11 00 10	01 11 00 10

FIGURE 5.7 – Exemple de choix du prochain saut.

Ensuite, l'idée est, pour chaque niveau  $i$  allant de  $l$  à 1, de sélectionner dans la table, si elle existe, une route dont la zone d'agrégation de niveau  $i$  correspondante est :

- la plus proche de celle de niveau  $i$  contenant la destination ;
- contenue dans la zone d'agrégation de la dernière route sélectionnée.

Le prochain saut correspond à la passerelle de la dernière route sélectionnée suivant le processus itératif précédent. A l'issue de ce dernier, si aucune route ne remplit les critères de sélection, nous considérons que le routage est un échec.

### 5.9.3 Exemple

Sur la figure 5.7, le nœud N reçoit des paquets à destination des nœuds A, B et C. Nous supposons que  $k = 1$  et  $m = 2$ . Nous faisons aussi l'hypothèse que la table de routage contient les entrées correspondant aux zones d'agrégation de niveau 1 et 2, en gris sur la figure. Prenons chacune des destinations précédentes et voyons comment N effectue le choix du prochain saut.

Le nœud A dont l'adresse est 001011100 se trouve dans la zone locale de N. Ainsi, ce dernier sélectionne la route dont la destination correspond au couple identifiant-adresse A-001011100. La passerelle correspondante est choisie comme prochain saut.

Le nœud B dont l'adresse est 001011111 se trouve en dehors de la zone locale. Aussi la table de routage de N ne peut pas contenir une entrée dont la destination est B-001011111. N applique donc l'algorithme 5.2. La plus grande zone d'agrégation contenant B mais qui ne contient pas N est celle de niveau 2 dont l'adresse est

001011. En effet, la zone d'agrégation de niveau 3 (0010) contenant B contient également N. Maintenant N parcourt sa table par ordre de niveau décroissant en commençant par le niveau 2 :

- La zone d'agrégation de niveau 2 contenant B est 001011. La zone d'agrégation de niveau 2 connue de N qui s'en rapproche le plus est 001011. La route correspondante est donc temporairement retenue ;
- La zone d'agrégation de niveau 1 contenant B est 00101111. La zone d'agrégation de niveau 1 connue de N qui s'en rapproche le plus est 00101100. De plus, cette dernière est contenue dans la zone de niveau 2 retenue précédemment (001011). Nous ne pouvons pas descendre plus bas en ce qui concerne le niveau. Le prochain saut est donc la passerelle de la route menant vers la zone d'agrégation 00101100.

Le nœud C dont l'adresse est 00101111 se trouve en dehors de la zone locale. La plus grande zone d'agrégation contenant B mais qui ne contient pas N est celle de niveau 3 dont l'adresse est 001011. Le paramètre  $m$  est égal à 2, ainsi le parcours de la table s'effectue à partir du niveau 2 :

- La zone d'agrégation de niveau 2 contenant C est 000100. La zone d'agrégation de niveau 2 connue de N qui s'en rapproche le plus est 001101. La route correspondante est donc temporairement retenue ;
- La zone d'agrégation de niveau 1 contenant C est 00010010. La zone d'agrégation de niveau 1 connue de N qui s'en rapproche le plus est 00101100. Cette dernière n'est pas contenue dans la zone d'agrégation de niveau 2 retenue précédemment. La route correspondante n'est donc pas sélectionnée. D'ailleurs, aucune autre zone d'agrégation de niveau 1 connue de N n'appartient à cette zone de niveau 2. Le prochain saut est donc la passerelle de la route ayant comme destination la zone d'agrégation de niveau 2 001101.

## 5.10 Simulation

Nous avons développé les algorithmes précédents afin d'évaluer notre protocole de routage via des simulations. Pour cela, nous avons utilisé le simulateur ns-3 [82] (ns-3.24).

### 5.10.1 Objectifs

Les principaux objectifs des différentes simulations sont les suivants :

- Étudier l'influence des paramètres du protocole que sont la taille d'une zone élémentaire,  $k$  et  $m$ . L'idée est de chercher des valeurs optimales pour ces paramètres ;
- Tester les performances du protocole : sa faculté à monter à grande échelle, son comportement selon la densité du réseau et sa gestion de la mobilité ;

- Comparer avec les approches à vecteurs de distance classique et géographique.

### 5.10.2 Métriques

Dans nos simulations, nous avons mis en relief les métriques suivants pour l'analyse des performances :

- Le temps de convergence : dans un contexte statique, il s'agit du temps nécessaire pour que la table de routage de tous les nœuds se stabilise, c'est-à-dire qu'il n'y a plus de nouvelles entrées ;
- La taille de la table de routage est le nombre moyen d'entrées dans une table de routage d'un nœud. Il s'agit, d'une part, d'un indicateur de la quantité en mémoire nécessaire pour exécuter correctement, et d'autre part, de la consommation en bande passante car les contenus sont échangés périodiquement entre les nœuds ;
- La quantité de trafic de contrôle mesure la bande passante consommée par le protocole pour son bon fonctionnement. Il s'agit principalement du trafic lié aux informations de mise à jour de route échangées entre les nœuds. Nous quantifions ce trafic en termes de nombre de paquets et en volume de données ;
- Le taux de succès d'acheminement est le rapport entre le nombre de paquets de données reçus et le nombre total de paquets envoyés ;
- L'optimalité est mesurée par l'intermédiaire du *stretch factor* qui est le rapport entre le nombre de sauts parcourus par un paquet de données selon le routage considéré et le nombre de sauts correspondant au plus court chemin possible.

### 5.10.3 Paramètres communs

Chaque simulation est lancée avec un ensemble de paramètres précis. Certains de ces paramètres sont communs à toutes les simulations. Il s'agit principalement des caractéristiques des nœuds :

- Modèle radio : wifi 802.11b à 2 Mbps, 250 m de portée ;
- Modèle MAC : sans QoS, mode ad-hoc ;
- Trafic de données :
  - Flux CBR à 2kbps de paquets de 64 octets ;
  - 10 flots entre 20 nœuds sélectionnés aléatoirement, 10 sources et 10 destinataires distincts, d'une durée de 10 s.
- Mobilité : pour simuler la mobilité, nous avons utilisé le modèle *random way point* dans lequel chaque nœud sélectionne une destination de manière aléatoire avant d'y mettre le cap à une vitesse constante, choisie de manière aléatoire. Arrivé à destination, le nœud observe une pause d'une durée bien définie avant de choisir une nouvelle destination et ainsi de suite. Nous avons choisi une vitesse maximale de 20 m/s pour les nœuds. Par ailleurs la durée

des flots de données est paramétrée à 100 s dans les scénarios de mobilité afin de bien capturer les effets de cette dernière.

Notons que toutes les moyennes seront présentées avec leurs intervalles de confiance à 95% respectifs.

#### 5.10.4 Influence de la taille de région élémentaire

Précédemment, nous avons proposé de choisir une taille de région élémentaire faisant en sorte que tous les nœuds situés dans la même région élémentaire sont à portée les uns des autres. Nous voulons tester l'impact sur les performances du choix d'une région élémentaire assez grande pour que des nœuds situés à l'intérieur de celle-ci ne puissent potentiellement pas communiquer directement entre eux. Notons par  $c = \frac{r}{\sqrt{2}}$  la longueur du côté de région élémentaire satisfaisant les conditions du premier cas avec  $r$  la portée des nœuds. Nous avons comparé les performances obtenues avec un côté de longueur  $2c$ , c'est-à-dire, une région élémentaire quatre fois plus large en matière de surface.

Nous avons déployé 20 à 300 nœuds en gardant une densité constante de 1 nœud par 9000 m<sup>2</sup>. Cette densité permet au réseau d'être connexe. Dans ce cas, les pertes de paquets sont uniquement dues à l'échec du routage. Les nœuds sont statiques afin de ne pas prendre en compte les pertes de performance liées à la mobilité. Les flots de données sont envoyés après convergence. Par ailleurs, nous avons fixé de manière arbitraire  $k$  à 1 et  $m$  à 4.

#### Taille des tables de routage

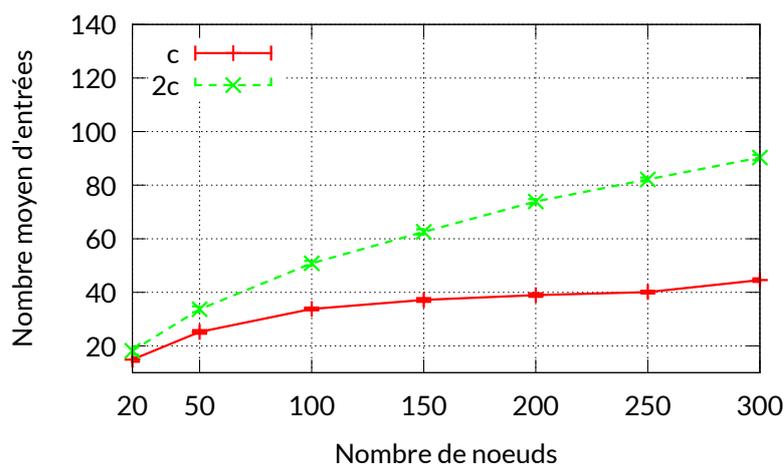


FIGURE 5.8 – Taille des tables et de région élémentaire.

Sur la figure 5.8, nous avons le nombre moyen d'entrées dans les tables de routage des nœuds en fonction du nombre de nœuds dans le réseau et selon les deux

valeurs de taille de région élémentaire. Nous constatons qu'augmenter la taille de région élémentaire rend les tables de routage plus volumineuses. En effet, l'augmentation de la taille de région élémentaire élargit la zone locale. Une zone locale plus large signifie plus de nœuds à l'intérieur et donc plus d'entrées dans la table. Par ailleurs, des régions élémentaires plus grandes conduisent à des zones d'agrégation plus importantes mais n'impactent pas significativement la taille des tables. En effet, même si le nombre de nœuds agrégés est plus élevé, au niveau de la table de routage, il y a toujours huit entrées au maximum pour chaque niveau.

### Trafic de contrôle

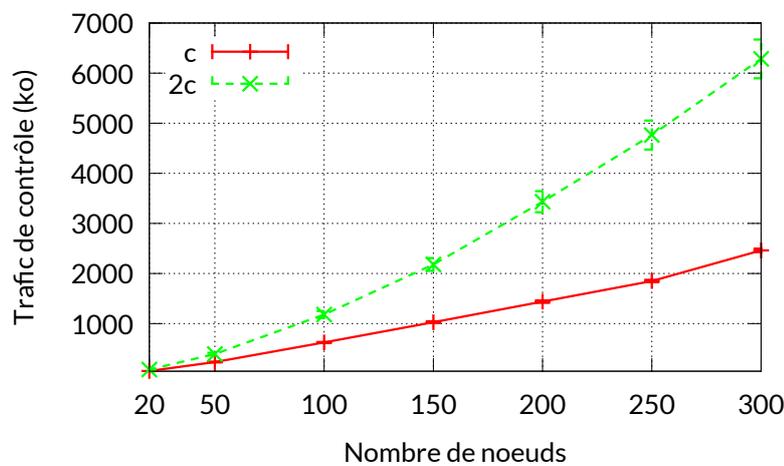


FIGURE 5.9 – Trafic de contrôle selon la taille de région élémentaire.

La figure 5.9 montre le trafic de contrôle généré pendant toute la durée d'une simulation en fonction du nombre de nœuds. À l'image du nombre d'entrées dans les tables de routage, le trafic de contrôle augmente également avec la taille de la région élémentaire. Ces résultats sont logiques car le trafic de contrôle a pour origine les échanges de contenu de tables de routage. Aussi, les mêmes explications que dans le paragraphe précédent s'appliquent.

### Taux de succès d'acheminement et optimalité

Les simulations montrent que pour les deux tailles de région élémentaire, le taux de succès d'acheminement des paquets est de quasiment 100% dans tous les cas. Sur la figure 5.10, nous avons une comparaison du *stretch factor* entre les deux tailles de région élémentaire. Nous voyons que pour  $2c$ , les routes sont meilleures. Le fait que deux nœuds appartenant à une même région élémentaire ne puissent potentiellement pas communiquer directement entre eux ne provoque donc pas des pertes de qualité en matière d'acheminement.

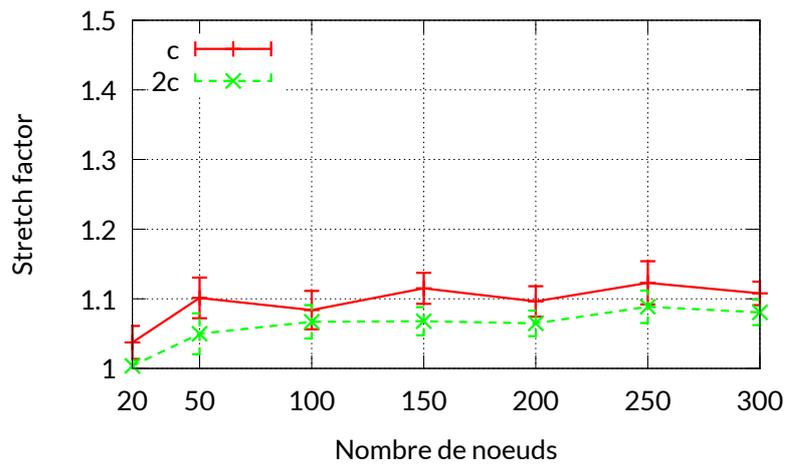


FIGURE 5.10 – Optimalité selon la taille de région élémentaire.

### Synthèse

Nous avons voulu tester si des régions élémentaires assez grandes pour que deux nœuds partageant une même région soient hors de portée l'un de l'autre impacte négativement le processus de routage. Nous avons observé le contraire avec des routes plus optimales mais au prix d'un volume de trafic de contrôle plus important. Aussi, pour la suite, nous allons utiliser des régions élémentaires de côté égal à  $\frac{r}{\sqrt{2}}$ . Au final, augmenter cette taille est, à peu de choses près, équivalent à élargir la zone locale et les zones d'agrégation avec les paramètres  $k$  et  $m$ .

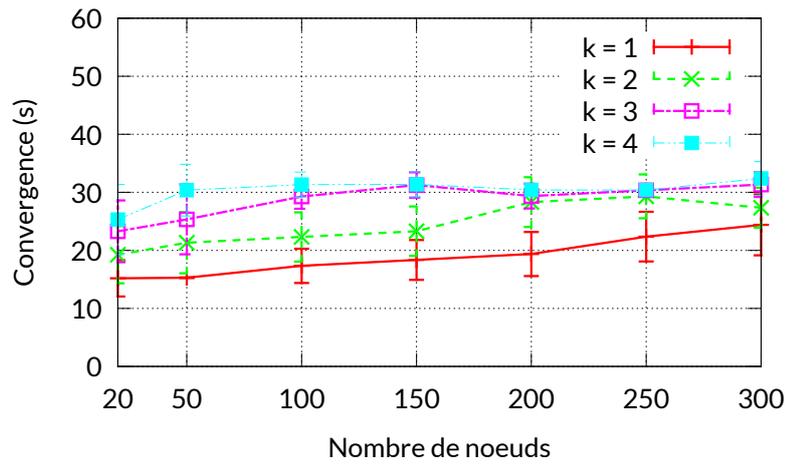
#### 5.10.5 Influence des paramètres $k$ et $m$ et passage à l'échelle

Dans cette sous-section, nous allons analyser les performances de notre protocole selon différentes valeurs de  $k$  et  $m$ . Pour cela nous avons considéré deux scénarios où les nœuds sont statiques :

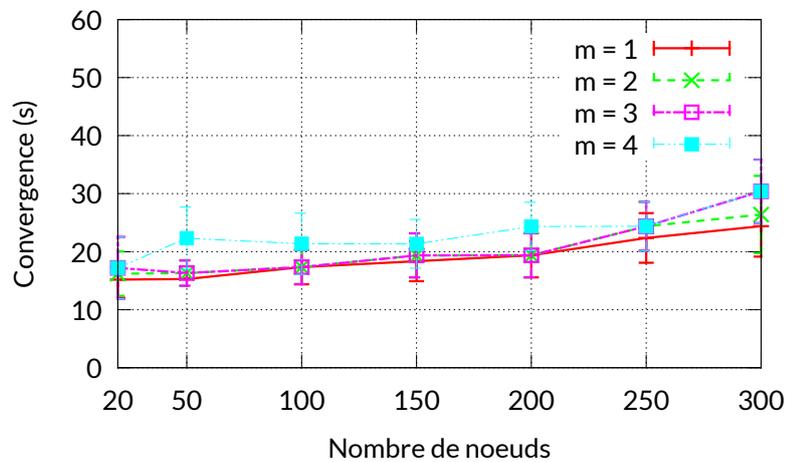
- Une densité constante à 1 nœud par 9000 m<sup>2</sup> pour 20 à 300 nœuds déployés ;
- 200 nœuds pour différentes densités allant de 0.25 à 1.5 nœuds par 9000 m<sup>2</sup>.

#### Temps de convergence

Sur les deux figures 5.11, nous avons les évolutions du temps de convergence en fonction du nombre de nœuds, la densité étant gardée constante. De manière générale, notre premier constat est que le temps de convergence augmente légèrement avec la taille du réseau. Il s'agit d'un avantage pour le déploiement de réseaux à grand nombre de nœuds. En effet, attendre la convergence est nécessaire pour que le routage soit opérationnel, ou du moins soit au maximum de son potentiel. Ainsi, le temps de convergence de notre protocole ne devrait pas trop ralentir le temps de déploiement des réseaux de grande taille.



(A)  $m$  fixé à 1.



(B)  $k$  fixé à 1.

FIGURE 5.11 – Temps de convergence selon différentes valeurs de  $k$  et  $m$ .

Plus  $k$  et  $m$  sont élevés, plus le temps de convergence augmente. En effet, en augmentant ces paramètres nous obtenons plus d'entrées dans les tables qui concernent des informations de routes vers des destinations éloignées. Si nous augmentons uniquement  $k$  (figure 5.11a), la taille de la zone locale s'agrandit et le nombre d'informations qui provoquent des mises à jour est plus important que si nous augmentons uniquement  $m$  auquel cas la taille de la zone locale reste inchangée (figure 5.11b).

### Taille des tables de routage

Les figures 5.12 montrent l'évolution du nombre moyen d'entrées dans les tables en fonction du nombre de nœuds dans le réseau. De manière générale, les courbes ont une forme logarithmique ce qui est un indicateur de capacité à monter à grande échelle.

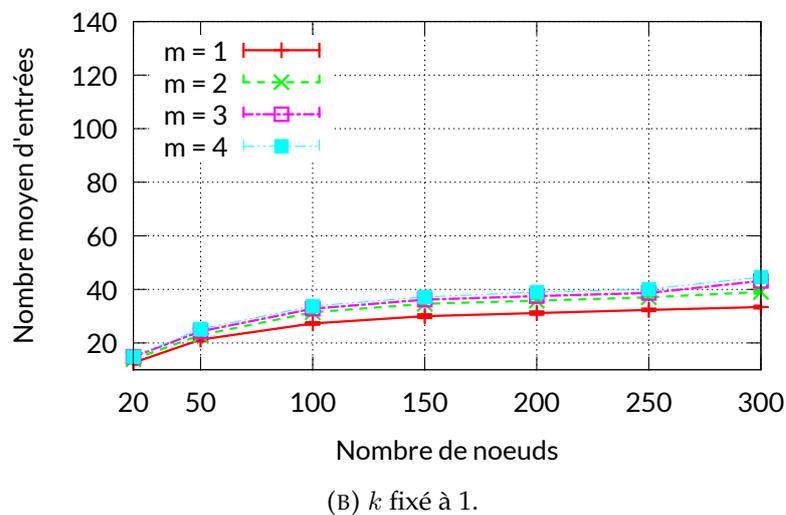
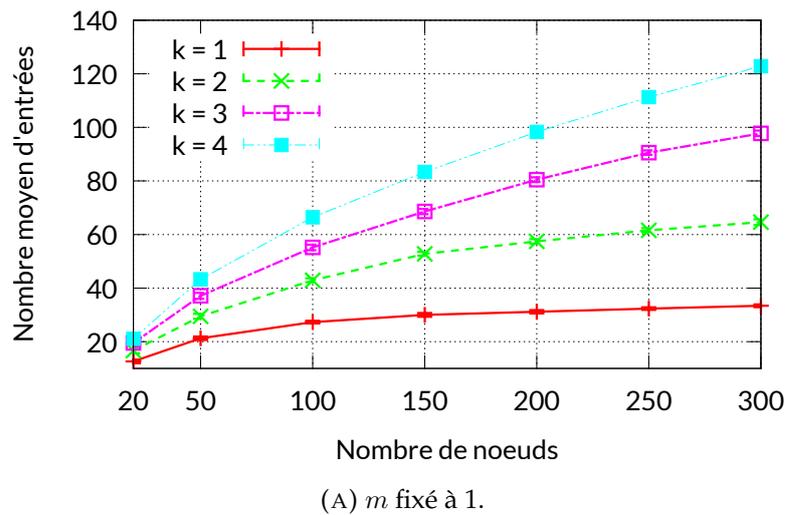
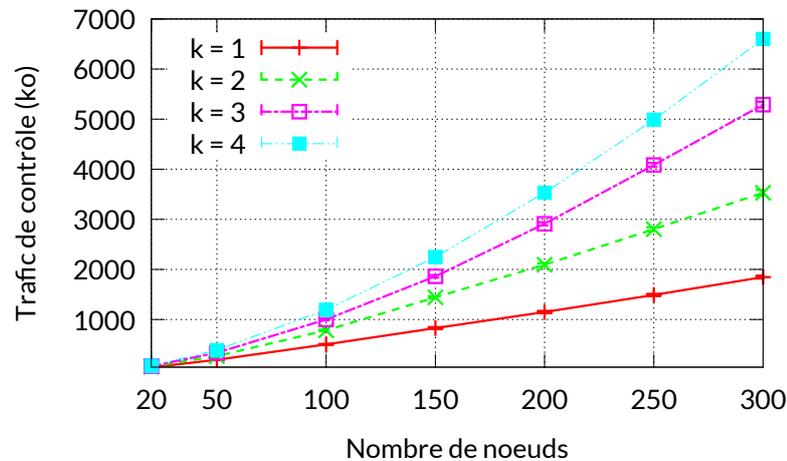


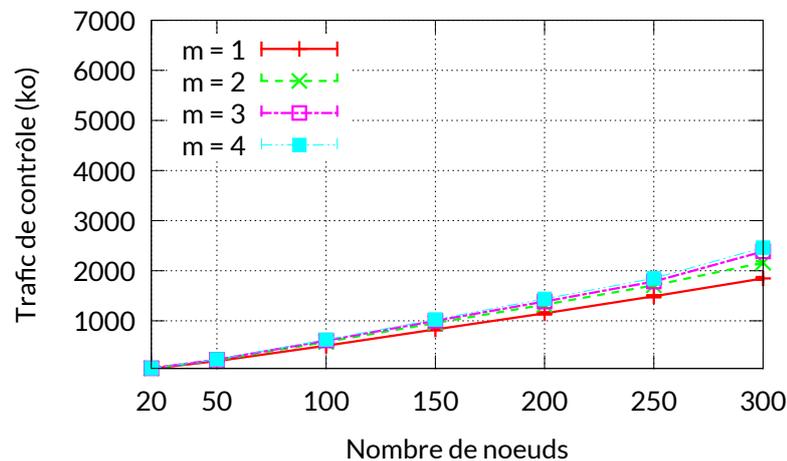
FIGURE 5.12 – Taille des tables de routage selon différentes valeurs de  $k$  et  $m$ .

En augmentant  $k$  (figure 5.12a), nous constatons que les courbes ont tendance à perdre leur forme logarithmique et la taille des tables devient de plus en plus linéaire avec la taille du réseau. En effet, plus  $k$  est grand, donc plus la zone locale est étendue, plus le nombre de nœuds contenus dans cette dernière augmente. Pour  $k$  suffisamment grand, chaque table finira par enregistrer une route pour chaque nœud dans tout le réseau d'où la linéarité de la taille de la table avec le nombre total de nœuds dans ce cas. Au contraire, quand  $m$  croît (figure 5.12b), la taille des tables augmente d'au maximum 8 entrées. Rappelons que pour chaque niveau, un nœud considère au maximum huit zones d'agrégation. Ainsi la courbe garde toujours sa forme logarithmique. De plus, un accroissement exponentiel du nombre de nœuds ne se traduit pas en une augmentation exponentielle du nombre d'entrées. Ce dernier est borné avec le mode de partitionnement mis en place.

Trafic de contrôle



(A)  $m$  fixé à 1.



(B)  $k$  fixé à 1.

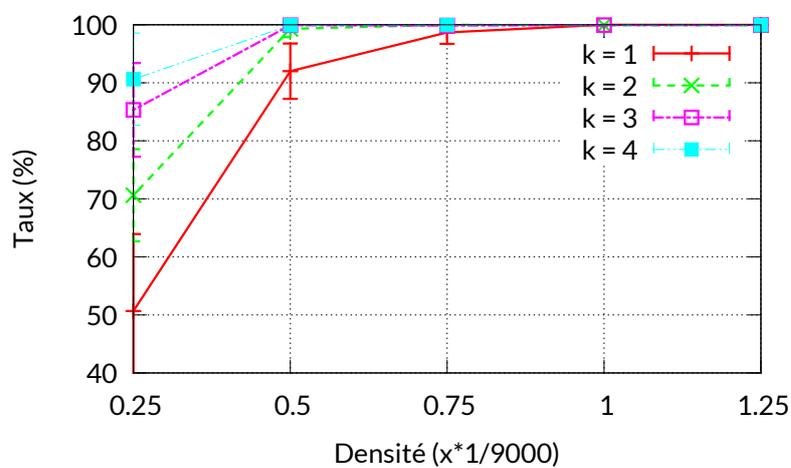
FIGURE 5.13 – Trafic de contrôle selon différentes valeurs de  $k$  et  $m$ .

Il s'agit du prix à payer sur le plan de la bande passante pour l'ajout et la mise à jour des entrées. Il existe donc une corrélation avec la taille des tables de routage. Aussi le trafic de contrôle devrait être plus sensible à la valeur du paramètre  $k$  que celle de  $m$ . Sur les figures 5.13, les courbes représentent le trafic de contrôle total généré pendant toute la durée d'une simulation en fonction du nombre de nœuds et selon plusieurs valeurs de ces paramètres.

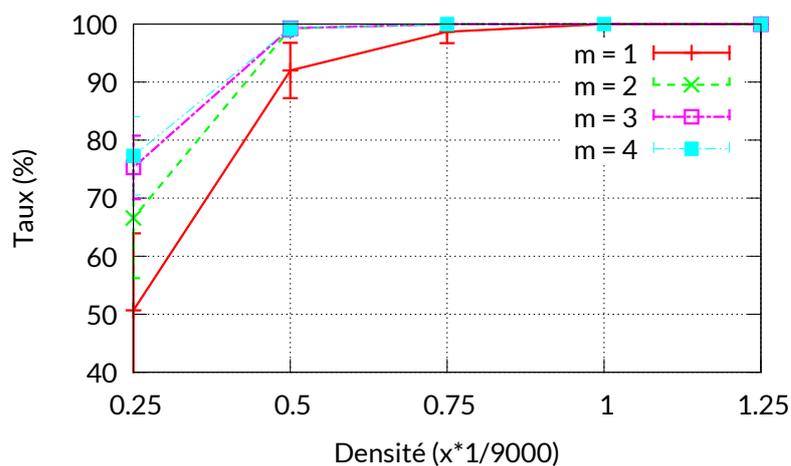
Nous constatons que plus  $k$  est élevé, plus les courbes adoptent une forme exponentielle (figure 5.13a). Nous pouvons interpréter ce résultat comme l'augmentation du trafic de contrôle rapporté à un nœud avec le nombre total de nœud dans le réseau, cette augmentation étant d'autant plus prononcée que  $k$  est élevé. Au contraire, sur la figure 5.13b, pour toutes les valeurs de  $m$  simulées, le trafic de contrôle présente une quasi-linéarité avec le nombre de nœuds dans le réseau. Cela signifie que

le trafic généré par chaque nœud est peu sensible à la taille du réseau. En effet, peu importe la valeur de  $m$ , c'est surtout l'étendue de la zone locale qui influence le plus la taille des tables de routage et, par extension, le trafic de contrôle. Ainsi, pour  $k$  fixé, la faible évolution de la taille des tables en fonction de celle du réseau se traduit par une quasi-linéarité du trafic total de contrôle. Enfin, nous notons la différence conséquente entre le trafic de contrôle généré pour des valeurs élevées de  $k$  et  $m$ , différence qui s'accroît avec le nombre total de nœuds dans le réseau. Par exemple, à 300 nœuds, pour  $k = 4$ , le trafic généré est presque trois fois plus élevé que pour  $m = 4$ .

### Taux de succès d'acheminement



(A)  $m$  fixé à 1.



(B)  $k$  fixé à 1.

FIGURE 5.14 – Taux de succès d'acheminement selon différentes valeurs de  $k$  et  $m$ .

Les figures 5.14 montrent le taux de succès d'acheminement ou *packet delivery ratio* (PDR) en fonction de la densité du réseau pour 250 nœuds dans le réseau. Comme

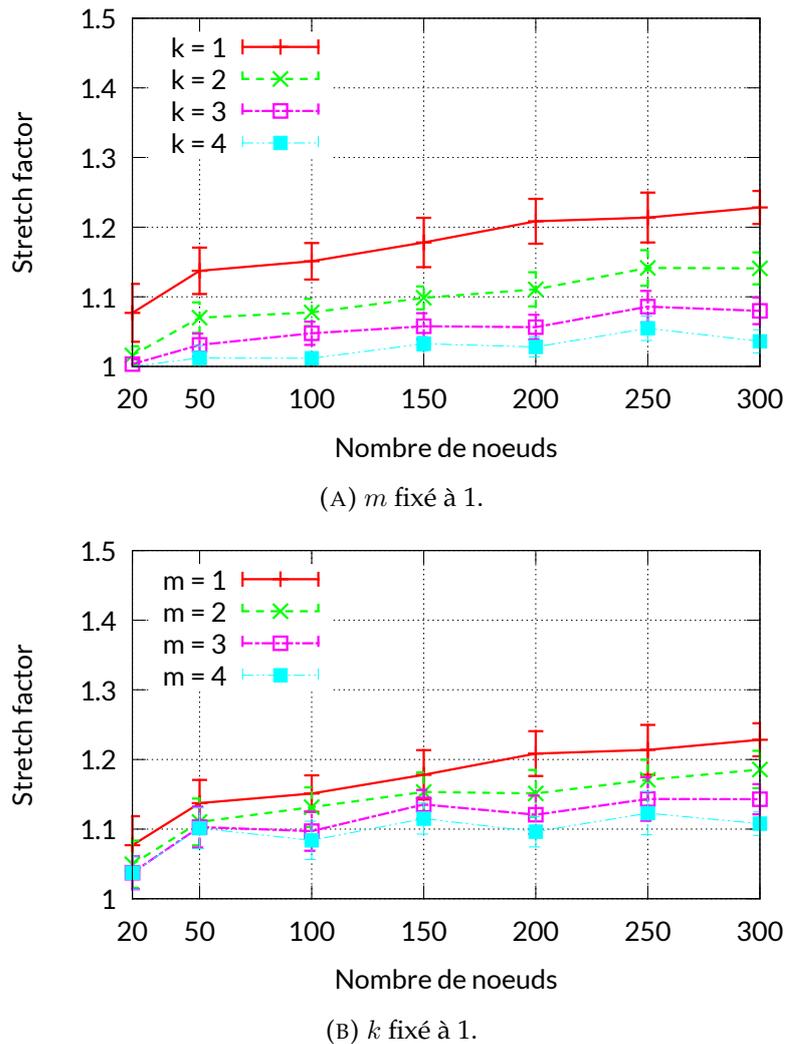
nous l'avons vu au moment de la description de l'algorithme utilisé pour l'acheminement des paquets, ces résultats confirment la non-garantie de livraison des paquets. Comme tout routage s'appuyant sur des critères géographiques pour le choix du prochain saut, comme le routage glouton par exemple, et ne résolvant pas activement les maximums locaux, le PDR décroît avec la densité. En effet, plus la densité est faible, plus un paquet est susceptible de rencontrer un maximum local.

De manière générale, plus les paramètres  $k$  et  $m$  sont élevés, meilleur est le PDR. En effet, quand on augmente ces paramètres, davantage d'informations sur des nœuds ou zones d'agrégation éloignés sont enregistrées, augmentant la probabilité d'anticipation d'un maximum local. Par ailleurs, en faisant une comparaison entre la figure 5.14a et la figure 5.14b, l'augmentation du paramètre  $k$  conduit à un meilleur PDR par rapport à celle de  $m$ . Pour comprendre, rappelons que les informations de mise à jour des routes avec lesquelles sont maintenues les tables et sur lesquelles les décisions d'acheminement sont prises, sont échangées localement mais se propagent nœud par nœud. Nous avons défini des règles d'agrégation qui rendent les routes de moins en moins détaillées avec l'éloignement. Plus le réseau est éparsé, plus les informations de mise à jour effectuent des contours avant d'arriver au niveau d'un nœud en particulier. Mais en faisant des contours, les informations risquent d'être « absorbées » par les zones d'agrégation et perdent ainsi de leur précision. Ce n'est pas le cas des routes qui circulent à l'intérieur de la zone locale qui ne subissent pas d'agrégation. Et plus la zone locale est large, c'est-à-dire plus  $k$  est élevé, plus les informations de mises à jour des routes ont une probabilité de rester à l'intérieur malgré les contournements. D'où les différences en termes de qualité de route et, par conséquent, en termes de PDR que nous constatons.

### **Optimalité des routes**

Les figures 5.15 montrent l'optimalité des routes en fonction du nombre de nœuds dans le réseau pour différentes valeurs de  $k$  et de  $m$ . Notons que la densité est ici gardée constante et, dans les simulations effectuées, le PDR est à 100%.

Dans un premier temps, nous pouvons constater une légère perte d'optimalité des routes pour des nœuds déployés de plus en plus nombreux. Cette dynamique de perte a tendance à être moins prononcée pour des valeurs élevées de  $k$  ou de  $m$ . Ensuite, en comparant les figures 5.15a et 5.15b, nous constatons que les routes obtenues pour des valeurs élevées de  $k$  sont plus proches du plus court chemin que celles obtenues en augmentant  $m$ . L'explication est la même que pour le cas du PDR : l'agrégation a pour conséquence des routes de moins bonne qualité, en particulier, moins optimales.

FIGURE 5.15 – *Stretch factor* selon différentes valeurs de  $k$  et  $m$ .

### Synthèse

Les résultats que nous avons présentés traitent de la capacité à passer à l'échelle de notre proposition ainsi que de l'influence des paramètres de celle-ci.

En premier lieu, nous avons constaté que, dans le cas où les nœuds sont statiques, le temps de convergence augmente de manière limitée avec la taille du réseau. Ceci peut être considéré comme un critère de capacité de passage à l'échelle étant donné que le temps séparant la mise en place d'un réseau et l'instant à partir duquel il est opérationnel est peu sensible au nombre de nœuds déployés. Nous avons également vu que la taille des tables de routage augmente selon une tendance logarithmique avec le nombre de nœuds dans le réseau. Par conséquent le trafic de contrôle généré n'explose pas avec la taille du réseau. Ainsi, il est possible de faire participer un grand nombre de nœuds au réseau sans que la consommation en bande passante liée à la maintenance du protocole n'augmente rapidement.

La comparaison des métriques selon différentes valeurs de  $k$  et de  $m$  montre qu'augmenter  $k$  permet d'avoir des routes de meilleures qualités en termes de taux de succès d'acheminement et d'optimalité des routes, en se basant sur le chemin le plus court, par rapport à un accroissement de la valeur de  $m$ . Mais la taille des tables de routage ainsi que le trafic de contrôle augmente plus rapidement. Au contraire, utiliser des valeurs plus élevées de  $m$  se traduit par une augmentation limitée de la taille des tables de routage et celle du trafic de contrôle au prix de routes de plus faible qualité que celles obtenues avec  $k$  plus grand. En résumé, pour rechercher des routes de meilleures qualités, il est plus intéressant de modifier  $k$ . Par contre, si les ressources sont limitées en ce qui concerne la bande passante du réseau, il faut privilégier la modification de  $m$ .

### 5.10.6 Comparaison avec d'autres protocoles de routage et mobilité

Notre proposition a pour objectif de rendre les approches de routage classique moins coûteuses en exploitant des informations de nature géographique tout en offrant de meilleures routes que les approches géographiques avec résolution active des maximums locaux. Nous avons donc retenu les protocoles suivants pour les études comparatives :

- DSDV [17] : pour rappel, il s'agit d'un protocole de routage à vecteurs de distance classique. La période d'envoi des mises à jour de routes est de 15 s (valeur par défaut) ;
- Routage géographique glouton : il s'agit d'une approche géographique qui ne traite pas les maximums locaux. Le choix du prochain saut correspond au voisin qui offre la meilleure progression géographique vers la destination. Le routage échoue si aucun voisin ne satisfait ce critère ;
- GPSR [36] : protocole de routage géographique qui élimine les maximums locaux en planarisant le graphe du réseau. La période d'envoi des messages de signalisation est de 1 s (valeur par défaut).

Nous avons choisi ces protocoles parce qu'ils sont à l'origine des principes de base des approches topologique et géographique. Les protocoles plus récents s'appuient en quelque sorte sur des optimisations de ces principes. Notre objectif est de tester la validité de notre idée concernant l'agrégation des routes et la qualité de sa mise en œuvre. Une fois cette étape passée, il sera possible de trouver des optimisations à notre proposition et de la comparer ainsi avec d'autres protocoles qui sont aussi plus optimisés. Nous profitons de cette étude comparative pour vérifier en même temps le comportement de notre proposition dans un contexte de mobilité.

Notre protocole a été simulé pour  $k = 1$  et  $m = 4$ . Nous avons vu qu'une valeur élevée de  $k$  offre certes des routes de meilleure qualité (figure 5.14a) mais au prix

d'une augmentation plus soutenue du trafic de contrôle en fonction du nombre de nœuds dans le réseau (figure 5.13a). Aussi, nous avons privilégié une plus grande valeur de  $m$  dont la croissance du trafic de contrôle est plus limitée. Le choix d'une valeur plus élevée pour  $k$  est toutefois plus pertinent dans le cas où la priorité est mise sur la qualité des routes au regard du volume de trafic de contrôle. Par ailleurs, nous avons défini la période d'envoi des mises à jour de route à 8 s.

### Taille de table de routage

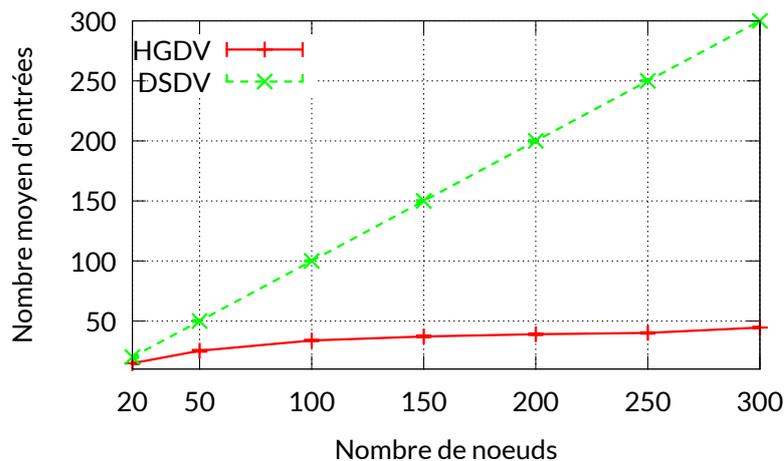


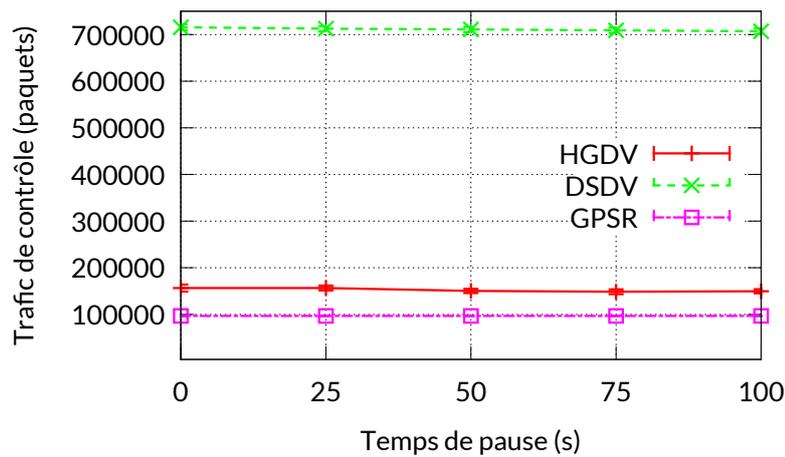
FIGURE 5.16 – Taille moyenne d'une table de routage.

Le figure 5.16 montre l'évolution en fonction du nombre de nœuds dans le réseau entre de la taille moyenne des tables de routage des nœuds exécutant notre protocole et DSDV. Ces résultats ont été obtenus avec des nœuds statiques. Nous avons la confirmation que, pour DSDV, le nombre moyen d'entrées dans les tables est, à quelques unités près, égal au nombre de total de nœuds dans le réseau après convergence dans le cas statique. En effet, pour DSDV, chaque destination possible génère une entrée dans la table. Au contraire, notre proposition, à travers l'agrégation des nœuds, peut générer une entrée pour plusieurs destinations possibles d'où une taille de table inférieure par rapport à DSDV. Par ailleurs, cette agrégation se fait dans un périmètre limité qui est indépendant de la taille du réseau. Aussi, il y a un creusement de la différence entre les deux courbes à mesure que le réseau s'agrandit.

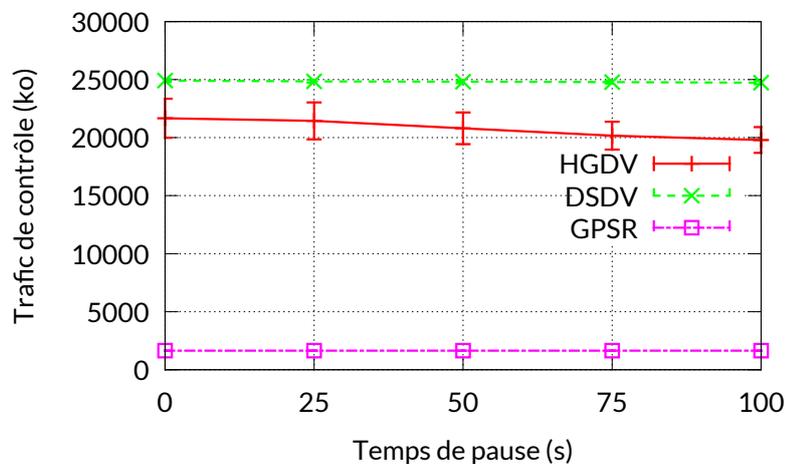
### Trafic de contrôle

Sur les figures 5.17, nous voyons les mesures du trafic de contrôle généré par les trois protocoles en fonction du temps de pause avec le modèle de mobilité *random waypoint*.

Concernant notre protocole, nous constatons une légère diminution du trafic de contrôle pour des temps de pauses élevés, c'est-à-dire, pour une mobilité moindre.



(A) Trafic de contrôle en nombre de paquets.



(B) Trafic de contrôle en ko.

FIGURE 5.17 – Trafic de contrôle en fonction de la mobilité.

En effet, dans ce cas, un nœud « reste » plus longtemps à l'intérieur d'une zone d'agrégation et ne provoque pas de changement dans les tables de routage des nœuds considérant la même zone d'agrégation.

Ces résultats montrent aussi que notre protocole génère des paquets de taille plus importante par rapport aux autres protocoles. En effet, les paquets véhiculent plus d'informations, notamment à cause de la distinction entre identifiant et adresse. La compression de ces informations est une piste d'amélioration intéressante dans le sens où le volume de trafic de contrôle en serait réduit pour le même nombre de paquets échangés.

Par ailleurs, en ce qui concerne le nombre de paquets, nous constatons des valeurs très inférieures par rapport à DSDV. Dans les réseaux sans-fil dont la couche d'accès (MAC) est partagée et concurrente, générer moins de paquets de contrôle

constitue un avantage.

L'effet de l'agrégation des routes est ici visible et se traduit par un trafic généré moindre par rapport à DSDV même avec une fréquence d'envoi de mises à jour de route plus élevée. Il est donc envisageable d'augmenter encore la fréquence de ces mises à jour pour avoir des routes plus fiables tout en maintenant le trafic comparable à celui généré par DSDV.

Le trafic de contrôle généré par GPSR correspond à des signalisations de présence aux voisins à un saut, ce qui explique la faible quantité de donnée de contrôle comptabilisée.

### Taux de succès d'acheminement

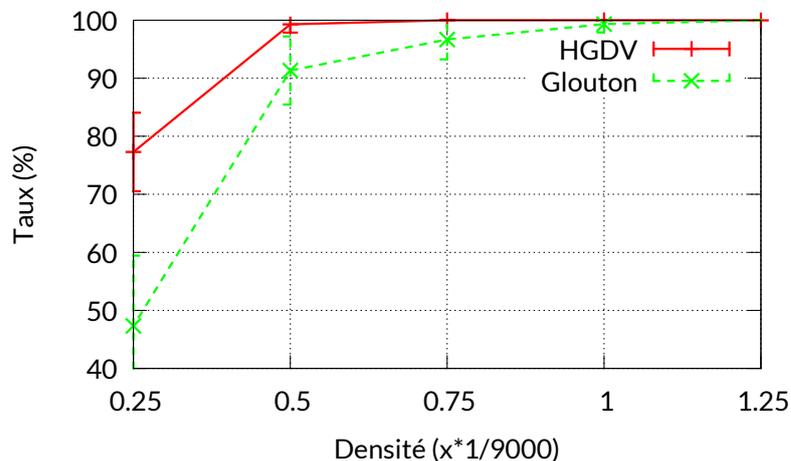


FIGURE 5.18 – Taux de succès d'acheminement par rapport au routage glouton.

La figure 5.18 montre une comparaison du PDR entre notre proposition et le routage glouton en fonction de la densité. Pour rappel, la probabilité de rencontrer des maximums locaux augmente avec un réseau de faible densité. Le routage glouton échoue dès qu'il rencontre un maximum local. Cette courbe illustre donc la capacité de notre proposition à contourner les maximums locaux. Elle montre aussi que, comme prévu, ce n'est pas la totalité des maximums locaux qui sont évités.

Sur la figure 5.19, nous avons une comparaison des PDR en fonction de la mobilité des nœuds, quantifiée par différents temps de pauses, la valeur 0 correspondant à des nœuds constamment en mouvement.

En premier lieu, nous constatons que le comportement de notre proposition face à la mobilité est comparable à celui des deux autres protocoles. Il n'y a pas beaucoup de différence entre les scénarios allant d'une mobilité constante des nœuds à

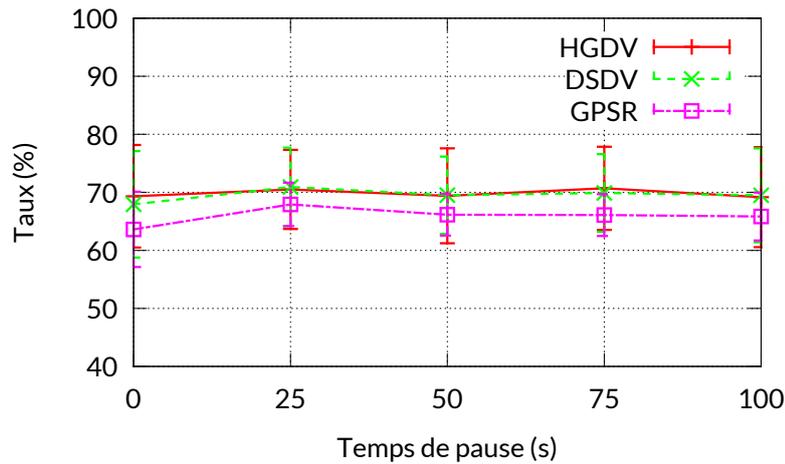


FIGURE 5.19 – Comparaison des taux de succès d'acheminement.

des temps de pause de 100 s. Le principe d'agrégation introduit une certaine forme de stabilité. Les effets de la mobilité sont en quelque sorte « absorbés » par les zones d'agrégation.

Comparée à DSDV (figure 5.19), notre proposition offre un PDR à peu près équivalent. Or, nous avons vu que le volume de trafic de contrôle associé est moindre. Ainsi, comme prévu, l'agrégation des routes permet des économies sur le plan de la bande passante consommée par les paquets de contrôle tout en ne détériorant pas les performances, du moins sur le taux de succès d'acheminement des paquets de données. Par ailleurs, le volume de trafic généré par notre proposition évolue modérément avec la taille du réseau comparé à DSDV qui, rappelons-le, enregistre dans les tables de routage les routes vers tous les nœuds du réseau et dont le volume total de trafic de contrôle évolue donc comme le carré de la taille du réseau. L'écart en ce qui concerne le trafic de contrôle généré ne cesse donc de croître à mesure que le réseau s'agrandit. Il sera alors possible pour notre proposition d'avoir des paramètres  $k$  et  $m$  plus élevés en ne dépassant pas le trafic de contrôle associé à DSDV tout en améliorant la qualité des routes.

Comparée à GPSR (figure 5.19), malgré la largeur de l'intervalle de confiance, nous pouvons dire que dans la majorité des cas, notre proposition a en moyenne un meilleur PDR. Ici, GPSR souffre de la mobilité. GPSR s'appuie sur la planarisation du graphe du réseau pour éviter les maximums locaux. Nous avons vu que la planarisation consiste, au niveau du réseau, à éliminer certains liens entre les nœuds sans provoquer la perte de connexité du réseau. Mais avec la mobilité des nœuds, l'algorithme de planarisation n'a pas toujours des informations topologiques fiables ce qui peut le conduire à éliminer des liens rendant le réseau non-connexe et éliminant ainsi des routes valables. En plus d'offrir un meilleur PDR, notre proposition donne aussi des routes plus proches du plus court chemin comme nous le verrons dans la

sous-section suivante.

### Optimalité

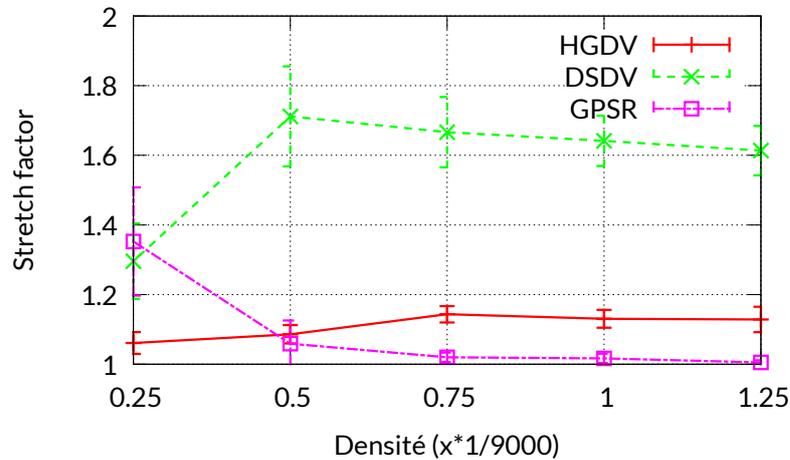


FIGURE 5.20 – Comparaison de l'optimalité par rapport au plus court chemin.

La figure 5.20 montre une comparaison de l'optimalité des routes générées par les trois protocoles, c'est-à-dire, à quel point les routes se rapprochent du plus court chemin possible. Nous constatons qu'à faible densité notre proposition offre les routes les plus courtes. GPSR génère des routes plus longues car les paquets arrivent aux maximums locaux avant de les contourner dans une direction arbitraire qui peut mener vers un chemin plus long. Ces courbes appuient donc notre hypothèse selon laquelle avoir à disposition des routes vers des destinations plus éloignées que le voisinage, même sous forme agrégée, permet d'anticiper les maximums locaux pour les contourner bien avant d'y arriver, donnant des routes plus proches du plus court chemin. Par contre, à plus forte densité, la non-existence des maximums locaux rend GPSR plus intéressant. En effet, en s'appuyant sur des routes agrégées donc moins précises, HGDV ne trouve pas toujours le chemin le plus court. DSDV présente ici les routes les plus longues. La raison est la convergence qui n'est pas encore atteinte au moment où les paquets de données ont commencé à être envoyés. DSDV correspond à notre proposition pour une valeur très élevée de  $k$ , ainsi son temps de convergence est plus long (voir figure 5.11a).

## 5.11 Conclusion

Ce chapitre a porté sur la mise en œuvre de la proposition de routage à vecteurs de distance géographique introduite au chapitre 3 en utilisant les principes développés au chapitre 4 qui sont :

- Un partitionnement hiérarchique de la zone de déploiement du réseau en régions ;

— Un adressage hiérarchique sans maintenance des nœuds et des régions.

Les régions issues du partitionnement sont utilisées pour définir la zone locale et les zones d'agrégation considérées par chaque nœud à travers les paramètres réglables  $k$  et  $m$ . Ensuite, un algorithme de mise à jour des tables de routage qui s'appuie sur une agrégation graduelle des routes avec l'éloignement a été proposé. En ce qui concerne l'acheminement des paquets, il s'agit d'utiliser les tables de routage en tenant compte des aspects géographiques pour sélectionner la passerelle menant au nœud ou à la zone d'agrégation la plus proche de la destination.

Nous avons conduit des simulations sous ns-3. Dans un premier temps, nous avons vérifié notre hypothèse sur le choix de la taille de région élémentaire offrant un compromis entre performances et volume de trafic de contrôle généré. Il s'agit de définir une taille de région élémentaire qui fait en sorte que deux ou plusieurs nœuds à l'intérieur d'une même région élémentaire puissent être à portée les uns des autres.

Par la suite, une comparaison des effets des paramètres  $k$  et  $m$  a été réalisée. Les résultats ont montré qu'augmenter  $k$  a beaucoup plus d'influence sur l'amélioration de la qualité des routes obtenues. En contrepartie, la taille des tables de routage s'accroît vite et avec elle le volume de trafic de contrôle. Au contraire, prendre des valeurs plus élevées de  $m$  offre en comparaison des gains de performances moins soutenus mais plus d'économie en termes de bande passante consommée par les paquets de contrôle. Dans tous les cas, notre proposition présente une capacité de passage à l'échelle à travers le mécanisme d'agrégation qui ne dépend pas du nombre de nœuds dans le réseau mais uniquement des distances considérées. Le choix de ces paramètres doit donc dépendre des caractéristiques du réseau au regard, par exemple de ses ressources au niveau du canal de communication. Le type de service qui sera proposé en exploitant le réseau doit aussi être pris en compte notamment en ce qui concerne le qualité de service (QoS) attendu.

Nous avons également effectué des comparaisons en contexte de mobilité avec le protocole DSDV et GPSR qui représentent respectivement l'approche de routage topologique classique et l'approche géographique. Comparé à DSDV, notre proposition offre un taux de succès d'acheminement des données comparable mais avec un volume de trafic de données moindre. Il s'agit de l'effet de l'agrégation qui réduit la quantité d'échanges nécessaires au bon fonctionnement du routage. En ce qui concerne ces échanges, notre proposition génère des paquets de grande taille mais en nombre moins important. Il s'agit d'un avantage au regard des aspects de concurrence au niveau de la couche d'accès (MAC) dans les réseaux sans fils. Par ailleurs, une amélioration possible consiste en la compression des informations contenues dans les paquets dans le but réduire leur volume totale en gardant le même nombre. L'optimalité supérieure de nos routes par rapport à DSDV peut s'expliquer par la

différence de convergence en présence de mobilité : tandis que DSDV s'appuie sur une convergence au niveau global, notre proposition s'intéresse à la convergence dans les zones proches, notamment la zone locale pour chaque nœud. Cette convergence peut être aussi accélérée en choisissant des fréquences d'envoi de mises à jour de route plus élevées. Par rapport à DSDV, notre solution peut se permettre des périodes d'envoi plus courtes puisqu'elle génère, de base, moins de trafic de contrôle. Par rapport à GPSR, à part un taux de succès d'acheminement supérieur en raison de la difficulté de la mise en œuvre des algorithmes de planarisation en milieu mobile, les routes sont aussi plus optimales car les maximums locaux sont évités plus en amont.

A partir de ces résultats, nous pouvons avancer que :

- Le concept d'agrégation permet de réduire le volume de trafic de contrôle des protocoles de routage à vecteurs de distance tout en préservant les performances ;
- Des informations de routes agrégées permettent d'anticiper les maximums locaux et résultent en des routes plus optimales.

Il s'agit de notre hypothèse de départ qui est donc confirmée à travers les résultats des simulations.

Nous avons donc conçu un protocole fonctionnel et conforme aux attentes en ce qui concerne l'application du principe d'agrégation et les performances obtenues. Mais pour vérifier avec le plus de rigueur les hypothèses faites, une formalisation mathématique est nécessaire. Cette dernière pourra aussi être utilisée pour prouver les caractéristiques propres au protocole et à l'acheminement comme la non-existence de boucle de routage par exemple.



## Chapitre 6

# Conclusion générale

Cette thèse s'intéresse à la problématique du routage dans les réseaux sans fil ad hoc, en particulier en ce qui concerne le passage à l'échelle. Nous avons évoqué deux approches existantes qui sont extrêmes. D'un côté, l'approche classique s'appuie sur des informations quasi-complètes concernant la topologie du réseau, par exemple l'ensemble de tous les liens ou de toutes les destinations possibles. Ces informations sont stockées dans les tables de routage dont le contenu sert de base pour les prises de décisions pour l'acheminement des paquets. Plus le réseau contient de nœuds, plus la quantité d'informations à maintenir est importante, rendant la proportion de bande passante consommée par les paquets de contrôle non négligeable. D'un autre côté, l'approche géographique maintient uniquement les informations de liens vers les nœuds dans le voisinage à un saut ce qui la rend insensible au nombre de nœuds dans tout le réseau. On a donc une approche passant à l'échelle. L'acheminement des paquets repose sur la comparaison entre la localisation géographique des nœuds voisins et celle du destinataire pour s'en rapprocher de plus en plus à chaque étape. Mais il n'existe pas toujours de nœud intermédiaire permettant cette progression. Cette « voie sans issue » est appelée maximum local. Les mécanismes utilisés pour la résolution des maximums locaux comme la planarisation et la technique de traversée de faces produisent des routes qui peuvent ne pas être optimales. En effet, trouver les routes optimales impliquent en général une connaissance parfaite de la topologie pour, par exemple, appliquer l'algorithme du plus court chemin.

Au vu de ces éléments, nous avons étudié une solution qui consiste à résumer les informations de topologie afin de rendre les tables de routage plus légères, et par conséquent, le trafic de contrôle moins important, tout en mettant à disposition du routage géographique plus d'informations dans le but de produire de meilleures routes. Nous proposons pour cela une agrégation des nœuds sur la base de critères géographiques. Les informations de lien vers plusieurs nœuds sont résumées en une information de lien vers une zone géographique que nous appelons zone d'agrégation. On a donc une réduction de la quantité d'information à retenir pour une approche topologique mais ces informations agrégées sont aussi exploitables par un routage géographique puisqu'elles sont de nature géographique.

Pour mettre en œuvre cette agrégation de nœuds, nous proposons deux principes :

- Un partitionnement du plan en régions hiérarchiques ;
- Un adressage hiérarchique des nœuds sur la base de ce partitionnement. Plus précisément, les nœuds acquièrent des adresses hiérarchiques en identifiant simplement la région dans laquelle ils se trouvent. On a donc une autostructuration hiérarchique des nœuds à travers leurs adresses et cela, sans nécessité de maintenance.

Nous exploitons ces deux principes pour proposer une approche de routage combinant les aspects topologiques et géographiques. Il s'agit d'un protocole de routage à vecteurs de distance hiérarchique géographique que nous avons nommé HGDV pour *Hierarchical Geographical Distance Vector protocol*. Il repose sur l'utilisation de tables de routage dont les entrées correspondent à des routes vers des nœuds ou des zones d'agrégation. HGDV se comporte comme un protocole de routage ad hoc classique dans un périmètre local et comme un protocole de routage géographique au-delà. A l'intérieur du périmètre local, des informations de liens vers des nœuds sont maintenues tandis qu'à l'extérieur, des informations de liens vers des zones d'agrégation de plus en plus larges avec l'éloignement sont enregistrées.

Comme le protocole, l'acheminement des paquets relève aussi de deux aspects : local et géographique. Notons que, grâce aux deux principes que nous avons présentés plus haut, un même type d'adresse est utilisé à la fois pour l'approche classique et l'approche géographique. Pour l'aspect géographique, il se différencie des approches existantes dans le sens où pour rejoindre une destination, ce n'est pas un nœud intermédiaire offrant une progression géographique positive qui est choisi mais plutôt une zone d'agrégation se rapprochant le plus de la destination. L'avantage est qu'une zone d'agrégation représente une information à plus grande distance, au-delà des nœuds voisins et offre une possibilité de contourner les maximums locaux potentiels entre la zone et le nœud source.

En proposant ce protocole, nous avons fait l'hypothèse qu'agréger les informations concernant les nœuds offre la possibilité, d'une part, de réduire le volume de trafic généré par les routages classiques tout en maintenant des performances raisonnables, et d'autre part, d'anticiper les problèmes de maximums locaux à travers les informations agrégées de topologie dépassant le voisinage immédiat. L'idée est donc d'utiliser l'agrégation comme principe permettant à une approche classique de passer à l'échelle et celle géographique d'offrir de meilleures routes. Les simulations qu'on a conduites ont confirmé ces hypothèses en montrant un volume de trafic moindre par rapport à une approche classique représentée par DSDV et un taux de succès d'acheminement et une optimalité supérieure par rapport à une approche géographique représentée par GPSR.

## Perspectives

Les thématiques suivantes peuvent faire l'objet de travaux correspondant à des améliorations ou extensions de ceux proposés dans le cadre de cette thèse.

### Service de localisation

Comme tout routage géographique, notre proposition repose sur un service de localisation résolvant les identifiants des nœuds en adresses. Des avantages pourraient être obtenus par l'intégration de ces deux services. Dans ce cas, le service de localisation utiliserait notre proposition de routage pour l'acheminement des requêtes et de réponses de localisation. Par ailleurs, notre partitionnement hiérarchique peut être exploité pour la conception d'un service autonome de localisation hiérarchique.

### Geocast

L'agrégation des nœuds géographiquement proches sous forme de régions peut être exploitée pour faire du multicast reposant sur des critères géographiques appelé geocast. En l'état, notre proposition est capable de faire du geocast vers des régions issues du partitionnement hiérarchique. Mais il est nécessaire d'enrichir la proposition pour permettre le geocast vers des zones géographiques quelconques.

### 3D

Les mécanismes développés dans cette thèse s'appliquent à un environnement en deux dimensions. Une extension du système d'adressage pour la prise en compte de la troisième dimension serait intéressante dans certains cas d'utilisation comme un déploiement à l'échelle d'un immeuble par exemple ou en ce qui concerne les FANETs, des réseaux ad hoc formés par des équipements volants.

### Réseau hétérogène

Les nœuds participant à un réseau sans fil ad hoc peuvent avoir des caractéristiques différentes donnant lieu à un réseau hétérogène. Par exemple, certains peuvent ne pas avoir accès à leurs positions géographiques. Grâce au concept de zone d'agrégation qui autorise les nœuds à partager une même adresse, les nœuds n'ayant pas à leur disposition leurs localisations géographiques pourraient quand même obtenir une adresse et exécuter le protocole de routage à condition d'avoir un nœud capable de se situer géographiquement comme voisin. De la même manière, un nœud ayant des ressources énergétiques limitées pourrait juste obtenir une adresse et être ainsi joignable sans participer au protocole de routage donc sans jouer un rôle de relais.

## Sécurité

La thématique de la sécurité n'a pas du tout été abordée dans cette thèse. Or, elle est primordiale pour certaines applications des réseaux sans fil ad hoc. Notre proposition est par conception très ouverte : l'acquisition des adresses ne nécessite pas d'échanges entre les nœuds dans le cas où les informations géographiques absolues sont utilisées. Un mécanisme à part entière doit donc être mis en place pour la sécurisation. Par exemple, les nœuds doivent être authentifiés avant leur participation éventuelle aux échanges liées au protocole de routage. En effet, les paquets de contrôle peuvent contenir des informations sensibles comme la localisation des nœuds.

# Publications

- [1] M. N. RANAIVO RAKOTONDRAVELONA, F. HARIVELO et P. ANELLI, « Hierarchical area-based address autoconfiguration protocol for self-organized networks », *AdHocNets 2015 7th International Conference on Ad Hoc Networks*, p. 173–184, 2015,  
DOI : [10.1007/978-3-319-25067-0\\_14](https://doi.org/10.1007/978-3-319-25067-0_14).
- [2] M. N. RANAIVO RAKOTONDRAVELONA, B. BAYNAT, F. HARIVELO et P. ANELLI, « Table-driven geographical routing using aggregate areas », *NextComp 2017 1st International Conference on Next Generation Computing Applications*, p. 41–46, 2017,  
DOI : [10.1109/NEXTCOMP.2017.8016174](https://doi.org/10.1109/NEXTCOMP.2017.8016174).



# Bibliographie

- [1] M. CONTI et S. GIORDANO, « Mobile ad hoc networking : milestones, challenges, and new research directions », *IEEE Communications Magazine*, t. 52, p. 85–96, 2014,  
DOI : [10.1109/MCOM.2014.6710069](https://doi.org/10.1109/MCOM.2014.6710069).
- [2] S.-H. YANG, « Principle of wireless sensor networks », *Wireless Sensor Networks : Principles, Design and Applications*, p. 7–47, 2014,  
DOI : [10.1007/978-1-4471-5505-8\\_2](https://doi.org/10.1007/978-1-4471-5505-8_2).
- [3] F. DOMINGOS DA CUNHA, A. BOUKERCHE, L. VILLAS, A. C. VIANA et A. A. F. LOUREIRO, « Data communication in VANETs : a survey, challenges and applications », INRIA Saclay ; INRIA, Research Report, 2014,  
URL : <https://hal.inria.fr/hal-00981126>.
- [4] İlker BEKMEZCI, O. K. SAHINGOZ et Şamil TEMEL, « Flying ad-hoc networks (FANETs) : a survey », *Ad Hoc Networks*, t. 11, p. 1254–1270, 2013,  
DOI : [10.1016/j.adhoc.2012.12.004](https://doi.org/10.1016/j.adhoc.2012.12.004).
- [5] D. BENYAMINA, A. HAFID et M. GENDREAU, « Wireless mesh networks design : a survey », *IEEE Communications Surveys Tutorials*, t. 14, p. 299–310, 2012,  
DOI : [10.1109/SURV.2011.042711.00007](https://doi.org/10.1109/SURV.2011.042711.00007).
- [6] M. CONTI, C. BOLDRINI, S. S. KANHERE, E. MINGOZZI, E. PAGANI, P. M. RUIZ et M. YOUNIS, « From MANET to people-centric networking : milestones and open research challenges », *Computer Communications*, t. 71, p. 1–21, 2015,  
DOI : [10.1016/j.comcom.2015.09.007](https://doi.org/10.1016/j.comcom.2015.09.007).
- [7] G. F. ELMASRY, « A comparative review of commercial vs. tactical wireless networks », *IEEE Communications Magazine*, t. 48, p. 54–59, 2010,  
DOI : [10.1109/MCOM.2010.5594677](https://doi.org/10.1109/MCOM.2010.5594677).
- [8] E. ROSAS, N. HIDALGO, V. GIL-COSTA, C. BONACIC, M. MARIN, H. SENGER, L. ARANTES, C. MARCONDES et O. MARIN, « Survey on simulation for mobile ad-hoc communication for disaster scenarios », *Journal of Computer Science and Technology*, t. 31, p. 326–349, 2016,  
DOI : [10.1007/s11390-016-1630-x](https://doi.org/10.1007/s11390-016-1630-x).
- [9] A. ur REHMAN, A. Z. ABBASI, N. ISLAM et Z. A. SHAIKH, « A review of wireless sensors and networks' applications in agriculture », *Computer Standards & Interfaces*, t. 36, p. 263–270, 2014,  
DOI : [10.1016/j.csi.2011.03.004](https://doi.org/10.1016/j.csi.2011.03.004).

- [10] J. DENG, H. ZENG, R. XU et X. WANG, « Dynamic and robust routing protocol for orbit access networks in lunar exploration », *MILCOM 2009 IEEE Military Communications Conference*, p. 1–7, 2009,  
DOI : [10.1109/MILCOM.2009.5379975](https://doi.org/10.1109/MILCOM.2009.5379975).
- [11] C. A. SANTIVANEZ, B. McDONALD, I. STAVRAKAKIS et R. RAMANATHAN, « On the scalability of ad hoc routing protocols », *Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*, 1688–1697 vol.3, 2002,  
DOI : [10.1109/INFCOM.2002.1019422](https://doi.org/10.1109/INFCOM.2002.1019422).
- [12] S. MOHSENI, R. HASSAN, A. PATEL et R. RAZALI, « Comparative review study of reactive and proactive routing protocols in MANETs », *4th IEEE International Conference on Digital Ecosystems and Technologies*, p. 304–309, 2010,  
DOI : [10.1109/DEST.2010.5610631](https://doi.org/10.1109/DEST.2010.5610631).
- [13] D. CHEN et P. K. VARSHNEY, « A survey of void handling techniques for geographic routing in wireless networks », *IEEE Communications Surveys Tutorials*, t. 9, p. 50–67, 2007,  
DOI : [10.1109/COMST.2007.358971](https://doi.org/10.1109/COMST.2007.358971).
- [14] F. VALOIS, « Self-organisation of multi-hop networks », Habilitation à diriger des recherches, Université Claude Bernard - Lyon I, 2007,  
URL : <https://tel.archives-ouvertes.fr/tel-00272361>.
- [15] P. MÜHLETHALER, « Routage dans les réseaux ad hoc », *Techniques de l'ingénieur Réseau Internet : protocoles, multicast, routage, MPLS et mobilité*, t. base documentaire : TIB289DUO. 2004,  
URL : <http://www.techniques-ingenieur.fr/base-documentaire/technologies-de-l-information-th9/reseau-internet-protocoles-multicast-routage-mpls-et-mobilite-42289210/routage-dans-les-reseaux-ad-hoc-te7520/>.
- [16] T. WATTEYNE, A. MOLINARO, M. G. RICHICHI et M. DOHLER, « From MANET to IETF ROLL standardization : a paradigm shift in WSN routing protocols », *IEEE Communications Surveys Tutorials*, t. 13, p. 688–707, 2011,  
DOI : [10.1109/SURV.2011.082710.00092](https://doi.org/10.1109/SURV.2011.082710.00092).
- [17] C. E. PERKINS et P. BHAGWAT, « Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers », *SIGCOMM '94 Conference on Communications Architectures, Protocols and Applications*, p. 234–244, 1994,  
DOI : [10.1145/190314.190336](https://doi.org/10.1145/190314.190336).
- [18] T.-W. CHEN et M. GERLA, « Global state routing : a new routing scheme for ad-hoc wireless networks », *ICC '98 IEEE International Conference on Communications*, p. 171–175, 1998,  
DOI : [10.1109/ICC.1998.682615](https://doi.org/10.1109/ICC.1998.682615).

- [19] G. PEI, M. GERLA et T.-W. CHEN, « Fisheye state routing : a routing scheme for ad hoc wireless networks », *ICC 2000 IEEE International Conference on Communications*, p. 70–74, 2000,  
DOI : [10.1109/ICC.2000.853066](https://doi.org/10.1109/ICC.2000.853066).
- [20] D. B. JOHNSON et D. A. MALTZ, « Dynamic source routing in ad hoc wireless networks », *Mobile Computing*, p. 153–181, 1996,  
DOI : [10.1007/978-0-585-29603-6\\_5](https://doi.org/10.1007/978-0-585-29603-6_5).
- [21] C. PERKINS, E. BELDING-ROYER, S. DAS et al., « RFC 3561-ad hoc on-demand distance vector (AODV) routing », *Internet RFCs*, p. 1–38, 2003,  
URL : <https://www.rfc-editor.org/rfc/rfc3561.txt>.
- [22] S. J. LEE et M. GERLA, « Aodv-br : backup routing in ad hoc networks », *WCNC 2000 IEEE Wireless Communications and Networking Conference*, p. 1311–1316, 2000,  
DOI : [10.1109/WCNC.2000.904822](https://doi.org/10.1109/WCNC.2000.904822).
- [23] C. PERKINS, S RATLIFF, J DOWDELL, L STEENBRINK et V MERCIECA, « Ad hoc on-demand distance vector version 2 (AODVv2) routing », Internet Draft (Standards Track), Mobile Ad hoc Networks Working Group, rapp. tech., 2016,  
URL : <https://tools.ietf.org/html/draft-ietf-manet-aodvv2-13>.
- [24] B. PATEL et S. SRIVASTAVA, « Performance analysis of zone routing protocols in mobile ad hoc networks », *NCC 2010 National Conference on Communications*, p. 1–5, 2010,  
DOI : [10.1109/NCC.2010.5430164](https://doi.org/10.1109/NCC.2010.5430164).
- [25] T. A. RAMREKHA et C. POLITIS, « A hybrid adaptive routing protocol for extreme emergency ad hoc communication », *19th International Conference on Computer Communications and Networks*, p. 1–6, 2010,  
DOI : [10.1109/ICCCN.2010.5560057](https://doi.org/10.1109/ICCCN.2010.5560057).
- [26] T. A. RAMREKHA, O. ADIGUN, A. LADAS, N. WEERASINGHE et C. POLITIS, « Towards a scalable routing approach for mobile ad-hoc networks », *IEEE 20th International Workshop on Computer Aided Modelling and Design of Communication Links and Networks*, p. 261–266, 2015,  
DOI : [10.1109/CAMAD.2015.7390521](https://doi.org/10.1109/CAMAD.2015.7390521).
- [27] J. SUCEC et I. MARSIC, « Hierarchical routing overhead in mobile ad hoc networks », *IEEE Transactions on Mobile Computing*, t. 3, p. 46–56, 2004,  
DOI : [10.1109/TMC.2004.1261816](https://doi.org/10.1109/TMC.2004.1261816).
- [28] G. PEI, M. GERLA, X. HONG et C. C. CHIANG, « A wireless hierarchical routing protocol with group mobility », *WCNC 1999 IEEE Wireless Communications and Networking Conference*, 1538–1542 vol.3, 1999,  
DOI : [10.1109/WCNC.1999.796996](https://doi.org/10.1109/WCNC.1999.796996).

- [29] M. SOOD et S. KANWAR, « Clustering in MANET and VANET : a survey », *CSCITA 2014 International Conference on Circuits, Systems, Communication and Information Technology Applications*, p. 375–380, 2014,  
DOI : [10.1109/CSCITA.2014.6839290](https://doi.org/10.1109/CSCITA.2014.6839290).
- [30] T.-F. SHIH et H.-C. YEN, « Core location-aided cluster-based routing protocol for mobile ad hoc networks », *10th WSEAS International Conference on Communications*, p. 223–228, 2006,  
URL : <http://dl.acm.org/citation.cfm?id=1981726.1981768>.
- [31] W. B. HEINZELMAN, A. P. CHANDRAKASAN et H. BALAKRISHNAN, « An application-specific protocol architecture for wireless microsensor networks », *IEEE Transactions on Wireless Communications*, t. 1, p. 660–670, 2002,  
DOI : [10.1109/TWC.2002.804190](https://doi.org/10.1109/TWC.2002.804190).
- [32] O. YOUNIS et S. FAHMY, « HEED : a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks », *IEEE Transactions on Mobile Computing*, t. 3, p. 366–379, 2004,  
DOI : [10.1109/TMC.2004.41](https://doi.org/10.1109/TMC.2004.41).
- [33] J. ERIKSSON, M. FALOUTSOS et S. KRISHNAMURTHY, « DART : dynamic address routing for scalable ad hoc and mesh networks », *IEEE/ACM Transactions on Networking*, t. 15, p. 119–132, 2007,  
DOI : [10.1109/TNET.2006.890092](https://doi.org/10.1109/TNET.2006.890092).
- [34] Y.-B. KO et N. H. VAIDYA, « Location-aided routing (LAR) in mobile ad hoc networks », *Wireless Networks*, t. 6, p. 307–321, 2000,  
DOI : [10.1023/A:1019106118419](https://doi.org/10.1023/A:1019106118419).
- [35] F. KUHN, R. WATTENHOFER et A. ZOLLINGER, « An algorithmic approach to geographic routing in ad hoc and sensor networks », *IEEE/ACM Trans. Netw.*, t. 16, p. 51–62, 2008,  
DOI : [10.1109/TNET.2007.900372](https://doi.org/10.1109/TNET.2007.900372).
- [36] B. KARP et H. T. KUNG, « GPSR : greedy perimeter stateless routing for wireless networks », *6th Annual International Conference on Mobile Computing and Networking*, p. 243–254, 2000,  
DOI : [10.1145/345910.345953](https://doi.org/10.1145/345910.345953).
- [37] Y.-J. KIM, R. GOVINDAN, B. KARP et S. SHENKER, « Geographic routing made practical », *2nd Conference on Symposium on Networked Systems Design & Implementation*, p. 217–230, 2005,  
URL : <http://dl.acm.org/citation.cfm?id=1251203.1251219>.
- [38] Y.-J. K. R. GOVINDAN, B. KARP et S. SHENKER, « Lazy cross-link removal for geographic routing », *4th International Conference on Embedded Networked Sensor Systems*, p. 112–124, 2006,  
DOI : [10.1145/1182807.1182819](https://doi.org/10.1145/1182807.1182819).

- [39] F. KUHN, R. WATTENHOFER et A. ZOLLINGER, « Worst-case optimal and average-case efficient geometric ad-hoc routing », *4th ACM International Symposium on Mobile Ad Hoc Networking & Computing*, p. 267–278, 2003,  
DOI : [10.1145/778415.778447](https://doi.org/10.1145/778415.778447).
- [40] J. NA et C. KWON KIM, « GLR : a novel geographic routing scheme for large wireless ad hoc networks », *Computer Networks*, t. 50, p. 3434–3448, 2006,  
DOI : <http://dx.doi.org/10.1016/j.comnet.2006.01.004>.
- [41] B. LEONG, B. LISKOV et R. MORRIS, « Geographic routing without planarization. », *NSDI 2006 3rd Symposium on Networked Systems Design and Implementation*, p. 25, 2006,  
URL : [http://static.usenix.org/events/nsdi06/tech/full\\_papers/leong/leong\\_html/](http://static.usenix.org/events/nsdi06/tech/full_papers/leong/leong_html/).
- [42] W.-H. LIAO, J.-P. SHEU et Y.-C. TSENG, « GRID : a fully location-aware routing protocol for mobile ad hoc networks », *Telecommunication Systems*, t. 18, p. 37–60, 2001,  
DOI : [10.1023/A:1016735301732](https://doi.org/10.1023/A:1016735301732).
- [43] D. M. SHRESTHA, C. KIM et Y.-B. KO, « A reliable multi-grid routing protocol for tactical manets », *ACM Symposium on Research in Applied Computation*, p. 164–169, 2011,  
DOI : [10.1145/2103380.2103414](https://doi.org/10.1145/2103380.2103414).
- [44] Q. CAO et T. ABDELZAHER, « Scalable logical coordinates framework for routing in wireless sensor networks », *ACM Trans. Sen. Netw.*, t. 2, p. 557–593, 2006,  
DOI : [10.1145/1218556.1218561](https://doi.org/10.1145/1218556.1218561).
- [45] R. FONSECA, S. RATNASAMY, J. ZHAO, C. T. EE, D. CULLER, S. SHENKER et I. STOICA, « Beacon vector routing : scalable point-to-point routing in wireless sensor networks », *2nd Conference on Symposium on Networked Systems Design & Implementation*, p. 329–342, 2005,  
URL : <http://dl.acm.org/citation.cfm?id=1251203.1251227>.
- [46] Y. ZHAO, Y. CHEN, B. LI et Q. ZHANG, « Hop ID : a virtual coordinate based routing for sparse mobile ad hoc networks », *IEEE Transactions on Mobile Computing*, t. 6, p. 1075–1089, 2007,  
DOI : [10.1109/TMC.2007.1042](https://doi.org/10.1109/TMC.2007.1042).
- [47] T. WINTER, P. THUBERT et al., « RFC 6550-RPL : ipv6 routing protocol for low-power and lossy networks », *Internet RFCs*, p. 1–158, 2012,  
URL : <https://tools.ietf.org/rfc/rfc6550.txt>.
- [48] J. LOO, J. MAURI et J. ORTIZ, *Mobile Ad Hoc Networks : Current Status and Future Trends*. CRC Press, 2016,  
URL : <https://books.google.com/books?id=k-zRBQAAQBAJ>.

- [49] S. DUROCHER, D. KIRKPATRICK et L. NARAYANAN, « On routing with guaranteed delivery in three-dimensional ad hoc wireless networks », *ICDCN 2008 Distributed Computing and Networking*, p. 546–557, 2008,  
DOI : [10.1007/978-3-540-77444-0\\_58](https://doi.org/10.1007/978-3-540-77444-0_58).
- [50] Y. WANG, C.-W. YI et F. LI, « Delivery guarantee of greedy routing in three dimensional wireless networks », *WASA 2008 Wireless Algorithms, Systems, and Applications*, p. 4–16, 2008,  
DOI : [10.1007/978-3-540-88582-5\\_4](https://doi.org/10.1007/978-3-540-88582-5_4).
- [51] H. HUANG, H. YIN, Y. LUO, X. ZHANG, G. MIN et Q. FAN, « Three-dimensional geographic routing in wireless mobile ad hoc and sensor networks », *IEEE Network*, t. 30, p. 82–90, 2016,  
DOI : [10.1109/MNET.2016.7437029](https://doi.org/10.1109/MNET.2016.7437029).
- [52] Q. CHEN, S. S. KANHERE et M. HASSAN, « Adaptive position update for geographic routing in mobile ad hoc networks », *IEEE Transactions on Mobile Computing*, t. 12, p. 489–501, 2013,  
DOI : [10.1109/TMC.2012.20](https://doi.org/10.1109/TMC.2012.20).
- [53] J. A. SANCHEZ, P. M. RUIZ et R. MARIN-PEREZ, « Beacon-less geographic routing made practical : challenges, design guidelines, and protocols », *IEEE Communications Magazine*, t. 47, p. 85–91, 2009,  
DOI : [10.1109/MCOM.2009.5181897](https://doi.org/10.1109/MCOM.2009.5181897).
- [54] F. CADGER, K. CURRAN, J. SANTOS et S. MOFFETT, « A survey of geographical routing in wireless ad-hoc networks », *IEEE Communications Surveys Tutorials*, t. 15, p. 621–653, 2013,  
DOI : [10.1109/SURV.2012.062612.00109](https://doi.org/10.1109/SURV.2012.062612.00109).
- [55] K. H. LI, J. S. LEU et J. HOEK, « Ant-based on-demand clustering routing protocol for mobile ad-hoc networks », *2013 Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, p. 354–359, 2013,  
DOI : [10.1109/IMIS.2013.66](https://doi.org/10.1109/IMIS.2013.66).
- [56] A. C. VIANA, M. D. de AMORIM, S. FDIDA et J. F. de REZENDE, « An underlay strategy for indirect routing », *Wireless Networks*, t. 10, p. 747–758, 2004,  
DOI : [10.1023/B:WINE.0000044033.02880.b7](https://doi.org/10.1023/B:WINE.0000044033.02880.b7).
- [57] C. TSCHUDIN, R. GOLD, O. RENSFELT et O. WIBLING, « LUNAR - a light-weight underlay network ad-hoc routing protocol and implementation », *NEW2AN 2004 Next Generation Teletraffic and Wired/Wireless Advanced Networking*, 2004,  
URL : <https://www.gta.ufrj.br/ensino/CPE825/2006/2004-new2an-lunar.pdf>.
- [58] D. E. PROZOROV, A. P. METELYOV, A. V. CHISTYAKOV et S. V. ROMANOV, « Simulation of a hierarchical routing protocol for manet », *EWDTs 2016 IEEE East-West Design Test Symposium*, p. 1–4, 2016,  
DOI : [10.1109/EWDTs.2016.7807681](https://doi.org/10.1109/EWDTs.2016.7807681).

- [59] M. TRANSIER, H. FÜSSLER, J. WIDMER, M. MAUVE et W. EFFELSBERG, « A hierarchical approach to position-based multicast for mobile ad-hoc networks », *Wireless Networks*, t. 13, p. 447–460, 2007,  
DOI : [10.1007/s11276-006-9199-x](https://doi.org/10.1007/s11276-006-9199-x).
- [60] L. A. HERNANDO et U. ARRONATEGUI, « Hierarchical geographic routing for wireless ad-hoc networks », *WSAN 2007 First International Conference on Wireless Sensor and Actor Networks*, p. 203–214, 2007,  
DOI : [10.1007/978-0-387-74899-3\\_18](https://doi.org/10.1007/978-0-387-74899-3_18).
- [61] N. WANGI, R. PRASAD, M. JACOBSSON et I. NIEMEGEREERS, « Address auto-configuration in wireless ad hoc networks : protocols and techniques », *IEEE Wireless Communications*, t. 15, p. 70–80, 2008,  
DOI : [10.1109/MWC.2008.4454707](https://doi.org/10.1109/MWC.2008.4454707).
- [62] S. NESARGI et R. PRAKASH, « MANETconf : configuration of hosts in a mobile ad hoc network », *Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*, 1059–1068 vol.2, 2002,  
DOI : [10.1109/INFCOM.2002.1019354](https://doi.org/10.1109/INFCOM.2002.1019354).
- [63] L. J. G. VILLALBA, J. G. MATESANZ, A. L. S. OROZCO et J. D. M. DÍAZ, « Distributed dynamic host configuration protocol (D2HCP) », *Sensors*, t. 11, 4438—4461, 2011,  
DOI : [10.3390/s110404438](https://doi.org/10.3390/s110404438).
- [64] L. J. G. VILLALBA, A. L. SANDOVAL OROZCO, J. G. MATESANZ et T.-H. KIM, « E-D2HCP : enhanced distributed dynamic host configuration protocol », *Computing*, t. 96, p. 777–791, 2014,  
DOI : [10.1007/s00607-013-0307-3](https://doi.org/10.1007/s00607-013-0307-3).
- [65] H. ZHOU, L. NI et M. MUTKA, « Prophet address allocation for large scale MANETs », *IEEE INFOCOM*, p. 1304–1311, 2003,  
DOI : [10.1109/INFCOM.2003.1208966](https://doi.org/10.1109/INFCOM.2003.1208966).
- [66] Y.-Y. HSU et C.-C. TSENG, « Prime DHCP : a prime numbering address allocation mechanism for manets », *IEEE Communications Letters*, t. 9, p. 712–714, 2005,  
DOI : [10.1109/LCOMM.2005.1496591](https://doi.org/10.1109/LCOMM.2005.1496591).
- [67] E. TALIPOV, H. SHIN, S. HAN et H. CHA, « A lightweight stateful address auto-configuration for 6lowpan », *Wireless Networks*, t. 17, p. 183–197, 2011,  
DOI : [10.1007/s11276-010-0272-0](https://doi.org/10.1007/s11276-010-0272-0).
- [68] Z. SHELBY et C. BORMANN, *6LoWPAN : The wireless embedded Internet*. John Wiley & Sons, 2011, t. 43,  
URL : <https://books.google.com/books?id=3Nm7ZCxscMQC>.
- [69] C. PERKINS, J. T. MALINEN et R. WAKIKAWA, « IP address autoconfiguration for ad-hoc networks », *IETF draft-ietf-manet-autoconf-01*, 2001.

- [70] M. FAZIO, M. VILLARI et A. PULIAFITO, « AIPAC : automatic IP address configuration in mobile ad-hoc networks », *Computer Communications*, t. 29, 2006, DOI : [10.1016/j.comcom.2005.07.006](https://doi.org/10.1016/j.comcom.2005.07.006).
- [71] K. WENIGER et M. ZITTERBART, « IPv6 autoconfiguration in large scale mobile ad-hoc networks », *European Wireless*, p. 142–148, 2002, URL : <http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.19.6458>.
- [72] N. KIM, S. AHN et Y. LEE, « AROD : an address autoconfiguration with address reservation and optimistic duplicated address detection for mobile ad hoc networks », *Computer Communications*, t. 30, p. 1913–1925, 2007, DOI : [10.1016/j.comcom.2007.03.002](https://doi.org/10.1016/j.comcom.2007.03.002).
- [73] A. YOUSEF, H. AL-MAHDI, M. A. r. KALIL et A. MITSCHLE-THIEL, « LHA : logical hierarchical addressing protocol for mobile ad-hoc networks », *ACM Workshop on Performance Monitoring and Measurement of Heterogeneous Wireless and Wired Networks*, p. 96–99, 2007, DOI : [10.1145/1298275.1298296](https://doi.org/10.1145/1298275.1298296).
- [74] X. WANG et H. QIAN, « A tree-based address configuration for a MANET », *Pervasive and Mobile Computing*, t. 12, p. 123–137, 2014, DOI : <http://dx.doi.org/10.1016/j.pmcj.2013.05.002>.
- [75] A. IWATA, C.-C. CHIANG, G. PEI, M. GERLA et T.-W. CHEN, « Scalable routing strategies for ad hoc wireless networks », *IEEE Journal on Selected Areas in Communications*, t. 17, p. 1369–1379, 1999, DOI : [10.1109/49.779920](https://doi.org/10.1109/49.779920).
- [76] E. ALOTAIBI et B. MUKHERJEE, « A survey on routing algorithms for wireless ad-hoc and mesh networks », *Computer Networks*, t. 56, p. 940–965, 2012, DOI : [10.1016/j.comnet.2011.10.011](https://doi.org/10.1016/j.comnet.2011.10.011).
- [77] F. DRESSLER, « A study of self-organization mechanisms in ad hoc and sensor networks », *Computer Communications*, t. 31, p. 3018–3029, 2008, DOI : [10.1016/j.comcom.2008.02.001](https://doi.org/10.1016/j.comcom.2008.02.001).
- [78] P.-H. HSIAO, « Geographical region summary service for geographical routing », *SIGMOBILE Mobile Computing and Communications Review*, t. 5, p. 25–39, 2001, DOI : [10.1145/509506.509515](https://doi.org/10.1145/509506.509515).
- [79] L. BLAŽEVIĆ, S. GIORDANO et J.-Y. LE BOUDEC, « Self organized terminode routing », *Cluster Computing*, t. 5, p. 205–218, 2002, DOI : [10.1023/A:1013998030317](https://doi.org/10.1023/A:1013998030317).

- [80] D. D. COUTO et R. MORRIS, « Location proxies and intermediate node forwarding for practical geographic forwarding », MIT Laboratory for Computer Science, rapp. tech., 2001,  
URL : <http://publications.csail.mit.edu/lcs/pubs/pdf/MIT-LCS-TR-824.pdf>.
- [81] S. DAS, H. PUCHA et Y. HU, « Performance comparison of scalable location services for geographic ad hoc routing », *INFOCOM 2005 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, p. 1228–1239, 2005,  
DOI : [10.1109/INFCOM.2005.1498349](https://doi.org/10.1109/INFCOM.2005.1498349).
- [82] *Ns-3 : a discrete event simulator*,  
URL : <https://www.nsnam.org/>.
- [83] T. H. CLAUSEN et E. BACCELLI, « A simple address autoconfiguration mechanism for OLSR [manet routing protocol] », *IEEE International Symposium on Circuits and Systems*, 2971–2974 Vol. 3, 2005,  
DOI : [10.1109/ISCAS.2005.1465251](https://doi.org/10.1109/ISCAS.2005.1465251).
- [84] F. DRESSLER, « Self-Organization in Ad Hoc Networks : Overview and Classification », University of Erlangen, Dept. of Computer Science 7, Technical Report, 2006.
- [85] X. HONG, K. XU et M. GERLA, « Scalable routing protocols for mobile ad hoc networks », *IEEE Network*, t. 16, p. 11–21, 2002,  
DOI : [10.1109/MNET.2002.1020231](https://doi.org/10.1109/MNET.2002.1020231).
- [86] S. A. ZEKAVAT, S. KANSAL et A. H. LEVESQUE, « Wireless positioning systems : operation, application, and comparison », *Handbook of Position Location*, p. 3–23, 2011,  
DOI : [10.1002/9781118104750.ch1](https://doi.org/10.1002/9781118104750.ch1).
- [87] I. GUVENC et C. C. CHONG, « A survey on TOA based wireless localization and nlos mitigation techniques », *IEEE Communications Surveys Tutorials*, t. 11, p. 107–124, 2009,  
DOI : [10.1109/SURV.2009.090308](https://doi.org/10.1109/SURV.2009.090308).
- [88] P. KUŁAKOWSKI, J. VALES-ALONSO, E. EGEA-LÓPEZ, W. LUDWIN et J. GARCÍA-HARO, « Angle-of-arrival localization based on antenna arrays for wireless sensor networks », *Computers & Electrical Engineering*, t. 36, p. 1181–1186, 2010,  
DOI : [10.1016/j.compeleceng.2010.03.007](https://doi.org/10.1016/j.compeleceng.2010.03.007).
- [89] G. DING, Z. TAN, L. ZHANG, Z. ZHANG et J. ZHANG, « Hybrid TOA/AOA cooperative localization in non-line-of-sight environments », *IEEE 75th Vehicular Technology Conference (VTC Spring)*, p. 1–5, 2012,  
DOI : [10.1109/VETECS.2012.6239884](https://doi.org/10.1109/VETECS.2012.6239884).

- [90] F. MOURAD, H. SNOUSSI et C. RICHARD, « Interval-based localization using RSSI comparison in MANETs », *IEEE Transactions on Aerospace and Electronic Systems*, t. 47, p. 2897–2910, 2011,  
DOI : [10.1109/TAES.2011.6034672](https://doi.org/10.1109/TAES.2011.6034672).
- [91] T. M. C. TRAN, B. SCHEUERMANN et M. MAUVE, « Lightweight detection of node presence in MANETs », *Ad Hoc Networks*, t. 7, p. 1386–1399, 2009,  
DOI : [10.1016/j.adhoc.2009.02.002](https://doi.org/10.1016/j.adhoc.2009.02.002).
- [92] S. Y. HAN et D. LEE, « An adaptive hello messaging scheme for neighbor discovery in on-demand manet routing protocols », *IEEE Communications Letters*, t. 17, p. 1040–1043, 2013,  
DOI : [10.1109/LCOMM.2013.040213.130076](https://doi.org/10.1109/LCOMM.2013.040213.130076).
- [93] E. LE MERRER, A.-M. KERMARREC et L. MASSOULIE, « Peer to peer size estimation in large and dynamic networks : a comparative study », *IEEE Symposium on International High Performance Distributed Computing*, p. 7–17, 2006,  
DOI : [10.1109/HPDC.2006.1652131](https://doi.org/10.1109/HPDC.2006.1652131).
- [94] V. GIRUKA et M. SINGHAL, « Hello protocols for ad-hoc networks : overhead and accuracy tradeoffs », *IEEE WoWMoM*, p. 354–361, 2005,  
DOI : [10.1109/WOWMOM.2005.50](https://doi.org/10.1109/WOWMOM.2005.50).
- [95] J. LI, J. JANNOTTI, D. DE COUTO, D. KARGER et R. MORRIS, « A scalable location service for geographic ad hoc routing », *International Conference on Mobile Computing and Networking*, p. 120–130, 2000,  
DOI : [10.1145/345910.345931](https://doi.org/10.1145/345910.345931).

# Table des figures

2.1	Maximum local. . . . .	14
2.2	Exemple de planarisation et de traversée de face. . . . .	15
3.1	Zone locale. . . . .	32
3.2	Exemple de zones d'agrégation. . . . .	33
3.3	Zones d'agrégation de N. . . . .	36
3.4	Exemple de mise à jour de table. . . . .	37
3.5	Exemple de sélection de prochain saut. . . . .	39
4.1	Partitionnement suivant un modèle de <i>quadtree</i> . . . . .	46
4.2	Partitionnement hiérarchique du plan ( <i>top-down</i> ). . . . .	47
4.3	Partitionnement hiérarchique du plan ( <i>bottom-up</i> ). . . . .	50
4.4	Croissance en spirale du partitionnement. . . . .	51
4.5	Exemple d'adressage par positionnement absolu. . . . .	52
4.6	Configuration du premier nœud et d'un nœud standard. . . . .	56
4.7	Configuration d'un nœud en AA. . . . .	57
4.8	Temps de convergence. . . . .	61
4.9	Paquets de contrôle. . . . .	61
5.1	Rappel sur le partitionnement et l'adressage. . . . .	66
5.2	Choix de la taille d'une région élémentaire. . . . .	67
5.3	Zone locale. . . . .	68
5.4	Zones d'agrégation. . . . .	69
5.5	Ensemble des zones d'agrégation de tous les niveaux en fonction de $m$ . . . . .	70
5.6	Exemple de mise à jour de table. . . . .	77
5.7	Exemple de choix du prochain saut. . . . .	80
5.8	Taille des tables et de région élémentaire. . . . .	83
5.9	Trafic de contrôle selon la taille de région élémentaire. . . . .	84
5.10	Optimalité selon la taille de région élémentaire. . . . .	85
5.11	Temps de convergence selon différentes valeurs de $k$ et $m$ . . . . .	86
5.12	Taille des tables de routage selon différentes valeurs de $k$ et $m$ . . . . .	87
5.13	Trafic de contrôle selon différentes valeurs de $k$ et $m$ . . . . .	88
5.14	Taux de succès d'acheminement selon différentes valeurs de $k$ et $m$ . . . . .	89
5.15	<i>Stretch factor</i> selon différentes valeurs de $k$ et $m$ . . . . .	91
5.16	Taille moyenne d'une table de routage. . . . .	93
5.17	Trafic de contrôle en fonction de la mobilité. . . . .	94

5.18 Taux de succès d'acheminement par rapport au routage glouton. . . .	95
5.19 Comparaison des taux de succès d'acheminement. . . . .	96
5.20 Comparaison de l'optimalité par rapport au plus court chemin. . . . .	97

## LETTRE D'ENGAGEMENT DE NON-PLAGIAT

Je, soussigné(e) .....,  
en ma qualité de doctorant(e) de l'Université de La Réunion, déclare être conscient(e) que le plagiat est un acte délictueux passible de sanctions disciplinaires. Aussi, dans le respect de la propriété intellectuelle et du droit d'auteur, je m'engage à systématiquement citer mes sources, quelle qu'en soit la forme (textes, images, audiovisuel, internet), dans le cadre de la rédaction de ma thèse et de toute autre production scientifique, sachant que l'établissement est susceptible de soumettre le texte de ma thèse à un logiciel anti-plagiat.

Fait à \_\_\_\_\_ le :

Signature :



**Extrait du Règlement intérieur de l'Université de La Réunion**  
(validé par le Conseil d'Administration en date du 11 décembre 2014)

### **Article 9. Protection de la propriété intellectuelle – Faux et usage de faux, contrefaçon, plagiat**

L'utilisation des ressources informatiques de l'Université implique le respect de ses droits de propriété intellectuelle ainsi que ceux de ses partenaires et plus généralement, de tous tiers titulaires de tels droits.

En conséquence, chaque utilisateur doit :

- utiliser les logiciels dans les conditions de licences souscrites ;
- ne pas reproduire, copier, diffuser, modifier ou utiliser des logiciels, bases de données, pages Web, textes, images, photographies ou autres créations protégées par le droit d'auteur ou un droit privatif, sans avoir obtenu préalablement l'autorisation des titulaires de ces droits.

#### **La contrefaçon et le faux**

Conformément aux dispositions du code de la propriété intellectuelle, toute représentation ou reproduction intégrale ou partielle d'une œuvre de l'esprit faite sans le consentement de son auteur est illicite et constitue un délit pénal.

L'article 444-1 du code pénal dispose : « Constitue un faux toute altération frauduleuse de la vérité, de nature à causer un préjudice et accomplie par quelque moyen que ce soit, dans un écrit ou tout autre support d'expression de la pensée qui a pour objet ou qui peut avoir pour effet d'établir la preuve d'un droit ou d'un fait ayant des conséquences juridiques ».

L'article L335\_3 du code de la propriété intellectuelle précise que : « Est également un délit de contrefaçon toute reproduction, représentation ou diffusion, par quelque moyen que ce soit, d'une œuvre de l'esprit en violation des droits de l'auteur, tels qu'ils sont définis et réglementés par la loi. Est également un délit de contrefaçon la violation de l'un des droits de l'auteur d'un logiciel (...) ».

**Le plagiat** est constitué par la copie, totale ou partielle d'un travail réalisé par autrui, lorsque la source empruntée n'est pas citée, quel que soit le moyen utilisé. Le plagiat constitue une violation du droit d'auteur (au sens des articles L 335-2 et L 335-3 du code de la propriété intellectuelle). Il peut être assimilé à un délit de contrefaçon. C'est aussi une faute disciplinaire, susceptible d'entraîner une sanction.

Les sources et les références utilisées dans le cadre des travaux (préparations, devoirs, mémoires, thèses, rapports de stage...) doivent être clairement citées. Des citations intégrales peuvent figurer dans les documents rendus, si elles sont assorties de leur référence (nom d'auteur, publication, date, éditeur...) et identifiées comme telles par des guillemets ou des italiques.

Les délits de contrefaçon, de plagiat et d'usage de faux peuvent donner lieu à une sanction disciplinaire indépendante de la mise en œuvre de poursuites pénales.