



HAL
open science

Vers une intégration des comportements communautaires dans les réseaux mobiles

Leïla Harfouche

► **To cite this version:**

Leïla Harfouche. Vers une intégration des comportements communautaires dans les réseaux mobiles. Informatique mobile. Conservatoire national des arts et métiers - CNAM, 2011. Français. NNT : 2011CNAM0790 . tel-00682193

HAL Id: tel-00682193

<https://theses.hal.science/tel-00682193>

Submitted on 23 Mar 2012

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.

École Doctorale du Conservatoire National des Arts et Métiers, ED 130

Centre de Recherche en Informatique et Communication
Laboratoire CEDRIC, EA 1395

THÈSE DE DOCTORAT

présentée par : **Leïla HARFOUCHE**

soutenue le : **27 septembre 2011**

pour obtenir le grade de : **Docteur du Conservatoire National des Arts et Métiers**

Discipline / Spécialité : **Informatique**

Vers une intégration des comportements communautaires dans les réseaux mobiles

THÈSE DIRIGÉE PAR

Mme BOUMERDASSI Selma

MCF / HDR, CNAM - CEDRIC

RAPPORTEURS

Pr MILOCCO Ruben

Professeur, Universidad Nacional del Comahue

Pr SENOUCI Sidi Mohammed

Professeur, ISAT

EXAMINATEURS

Pr BARKAOUI Kamel

Professeur, CNAM

Pr BEYLOT André-Luc

Professeur, INPT/ENSEEIH

Mme BOUZEFANE Samia

MCF / HDR, CNAM

M. GHAMRI-DOUDANE Yacine

MCF / HDR, ENSIE

M. RENAULT Éric

MCF, Télécom SudParis

À mon père.

*"Sache que tu ne peux acquérir une partie de la science
que lorsque tu te donnes entièrement à son acquisition"*

*"...et qu'un individu est qualifié de savant tant qu'il recherche la science,
le jour où il pense qu'il a tout acquis, c'est là qu'il devient ignorant"*

Dr. Ratib NABULSI

Remerciements

Mes remerciements vont en premier lieu au Pr Ruben MILOCCO ainsi qu'au Pr Sidi Mohammed SENOUCI pour m'avoir fait l'honneur d'accepter d'être les rapporteurs de ma thèse. Merci pour l'intérêt que vous avez accordé à mon travail.

Mes vifs remerciements vont à Mme Samia BOUZEFRANE, au Pr Kamel BARKAOUI, au Pr André-Luc BEYLOT et à MM. Yacine GHAMRI-DOUDANE et Éric RENAULT pour avoir accepté d'encadrer ma soutenance.

Je ne remercierai jamais assez Mme Selma BOUMERDASSI, ma directrice et bien plus, pour ses efforts pendant toutes les années de cette thèse. Il est difficile de la remercier à la hauteur de ses qualités tant personnelles que professionnelles, du cœur qu'elle y a mis et du temps qu'elle y a consacré. Merci pour TOUT.

Je remercie aussi le Pr Eric GRESSIER-SOUDAN qui m'a permis d'intégrer l'équipe.

Je tiens à remercier le Pr Jean Pierre ARNAUD pour m'avoir permis de découvrir le monde de l'enseignement au sein de la chaire de réseau et M. Joël BERTHELIN pour m'avoir toujours accompagnée et soutenue.

Je voudrai aussi exprimer ma gratitude à M. Éric RENAULT. Ses collaborations, ses conseils et ses remarques ont souvent influencés mon travail.

Un certain nombre de personnes du CEDRIC, ont de part leurs questions, remarques, bonne humeur et soutien contribué à cette thèse : Patrick, Amélie, Hélène, Julien, Rodrigo, Pedro, Guillaume, Christophe, Marc, José, Fatima, Mme Sider, Mme Gal, M. Pugliese et toutes les personnes que j'ai côtoyées au CEDRIC... Merci à toi Yahia.

Hervé méritera bien plus qu'un paragraphe pour lui exprimer ma gratitude.

Enfin, il n'existe pas de mots assez forts pour remercier mes parents, ma famille.

Résumé

À l'aire du *Multimedia Mobile*, l'essor des réseaux sans fil est fulgurant et la mobilité est devenue un sujet primordial exacerbé par l'augmentation significative du nombre d'utilisateurs mobiles.

Un nœud évoluant dans un *réseau mobile de base* se comporte de la même manière qu'un aveugle évoluant dans notre univers en élaborant sa propre représentation à l'aide de son bâton, mécanisme dénommé dans la littérature : *mobilité terminale*. Pour réduire cet aveuglement, plusieurs méthodes ont été élaborées qui prennent appui sur les services de localisation ou les modèles de mobilité.

Un modèle de mobilité est donc destiné à décrire en termes d'environnement, le mode de circulation des nœuds mobiles avec pour défi de trouver des modèles fidèles aux comportements des utilisateurs. Les modèles aléatoires sont biaisés car les appareils mobiles sont portés par des êtres sociaux. Cela nous a amené à inclure des éléments sociaux dans notre modélisation.

Nous présentons les modèles de mobilité existants et les classifions. Nous définissons nos modèles, les implémentons et mesurons leur impact sur le test des réseaux.

Enfin nous élargissons notre spectre en montrant comment le fait d'octroyer la perception de regroupement social à un protocole de routage réseau, peut en améliorer les performances.

Mots clés : Réseaux sans fil, modèle de mobilité pour l'évaluation des performances, protocoles de routage.

Abstract

In our area of *Mobile Multimedia*, the expansion of wireless networks is dazzling and mobility has become a major issue exacerbated by the significant increase in the number of mobile users.

A node operating in a *basic mobile network* behaves the same way a blind person moving in our universe by developing its own representation with his stick, a mechanism known in the literature as *terminal mobility*. To reduce this blindness, several methods have been developed that are based on location services and mobility models.

A mobility model is then intended to describe in terms of environment, the motion criteria of mobile nodes with the challenge to find models faithful to user behavior. Random models are biased because mobile devices are supported by social beings. This led us to include social elements in our models.

We present the existing mobility models and classify them. We define our models, implement them and measure their impact on the network testing.

Finally we expand our spectrum by showing that granting social grouping perception to a network routing protocol, can improve its performance.

Keywords : Wireless networks, mobility model for performance evaluation, routing protocols.

Table des matières

Remerciements	i
Résumé	iii
Abstract	v
1 Introduction	1
2 État de l’art	5
2.1 Introduction	5
2.2 Modèles Synthétiques	7
2.2.1 Modèles d’entités	8
2.2.2 Modèles de groupes	10
2.2.3 Conclusion	11
2.3 Modèles d’observation	12
2.4 Modèles à traces	12
2.5 Modèles de trafic	14
2.6 Classification sociale	14
2.7 Conclusion	16
3 La mobilité sociale	17
3.1 Introduction	17

TABLE DES MATIÈRES

3.2	Réseaux sociaux	19
3.3	Points d'Attraction (PA)	20
3.4	Modélisation	21
3.5	Application : Social Manhattan (SMN)	22
3.6	Implémentation	24
3.7	Test	26
3.7.1	Protocoles de routage réactifs	28
3.7.2	Protocoles de routage proactifs	28
3.7.3	Configurations de test	29
3.8	Résultats	30
3.8.1	Perspective réseau	40
3.9	Conclusion	42
4	Les réseaux pondérés	43
4.1	Introduction	43
4.2	Réseaux pondérés	44
4.2.1	Théorie des graphes	44
4.2.2	Exemples	45
4.3	Points d'Attraction pondérés	46
4.4	Modélisation	47
4.5	Maximum Acceptable Factor ($f(t)$) : exemples	49
4.5.1	Fonction constante	49
4.5.2	Fonction parabole	49
4.5.3	Fonction hybride mi-parabolique et mi-constante	51
4.6	Application : Weighted Manhattan (WMN)	51
4.7	Implémentation et paramètres du modèle	52

4.7.1	Implémentation	52
4.7.2	Paramètres du modèle	52
4.8	Résultats	54
4.8.1	Introduction	54
4.8.2	Configurations de test	54
4.9	Conclusion	67
5	Les systèmes de colonies de fourmis	69
5.1	Introduction	69
5.2	Le problème du voyageur de commerce	71
5.3	Modélisation	72
5.4	Application : Ant Manhattan (AMN)	73
5.5	Synthèse de nos modèles de mobilité	73
5.6	Résultats	75
5.6.1	Introduction	75
5.6.2	Configurations de test	75
5.7	Conclusion	88
6	Les protocoles de routage sociaux	91
6.1	Introduction	91
6.2	État de l'art des protocoles sociaux	92
6.3	Rappel sur le protocole <i>OLSR</i>	94
6.4	Une approche énergétique du protocole <i>OLSR</i> : e-OLSR	95
6.5	Une approche sociale du protocole <i>OLSR</i> : OLSR-S	96
6.6	Application	97
6.7	Implémentation	98
6.8	Résultats	103

TABLE DES MATIÈRES

6.9	Conclusion	105
7	Publications	107
8	Conclusions et Perspectives	109
8.1	Conclusion	109
8.2	Perspectives	111
	Bibliographie	115
	Annexes	123
A	Choix relatifs à la simulation et implémentation de nos modèles	123
A.1	Introduction	123
A.2	Présentation d' <i>Opnet Modeler</i>	124
A.3	Étude de l'existant	126
A.4	Nos premiers développements	141
A.5	Le problème des intersections manquées	141
A.6	Les 3 systèmes de coordonnées d' <i>Opnet Modeler</i>	142
A.6.1	Système de coordonnées global	142
A.6.2	Système de coordonnées relatif au sous-réseau	143
A.6.3	Système de coordonnées géocentrique	144
A.7	La cause du problème	145
A.8	La solution du problème	146
A.9	Création d'objet	146
A.9.1	Par duplication	147
A.9.2	Par dérivation	150

A.9.3	Création d'icône	153
A.9.4	Intégration d'icône et désignation standard	155
A.10	Attribut d'objet	157
A.10.1	Nom d'attribut	157
A.10.2	Modification	158
A.10.3	Création, suppression	164
A.11	Attribut étendu et XML ou limites de l'outil	166
B	Implémentation d'OLSR-S	171

Liste des tableaux

2.1	Classification Sociale	15
3.1	SMN	24
3.2	Simulation	31
4.1	WMN	52
4.2	Simulation	56
5.1	Simulation	77

Table des figures

2.1	Classification de Fiore	7
2.2	Classification de Bai	7
2.3	Classification de Haerri	7
2.4	Mouvement d'un nœud selon RWP et RD	9
2.5	Réinjections des nœuds dans le Boundless Area Mobility Model	10
2.6	Mouvement d'un nœud selon MANHATTAN	10
2.7	Mouvement des noeuds selon Colum Model et RPGM	11
2.8	Classification Sociale	14
3.1	Points d'attractions	21
3.2	Schéma de principe de Manhattan	23
3.3	Mouvement d'un nœud selon Social Manhattan : Schéma fonctionnel	25
3.4	Attributs Mobility Config Ext	27
3.5	Nouveaux objets visuels <i>Opnet Modeler</i>	28
3.6	Attributs Sociaux	28
3.7	SMN piéton	33
3.8	SMN bicyclette	35
3.9	SMN véhicule urbain	37
3.10	SMN - Variables statistiques réseau	39

TABLE DES FIGURES

4.1	Exemple de graphe pondéré	45
4.2	Un exemple de Point d'Attraction	47
4.3	Maximum Acceptable Factor $f(t)$ pour $D = 50$ et $M = 7$	51
4.4	Weighted Manhattan (WMN) : schéma de principe	53
4.5	WMN piéton	58
4.6	WMN bicyclette	60
4.7	WMN véhicule urbain	62
4.8	WMN - Variables statistiques réseau	64
5.1	Ant Colony Systems (ACS)	70
5.2	Algorithme des colonies de fourmis pour le TSP	72
5.3	Mouvement d'un nœud selon Ant Manhattan : Schéma fonctionnel	74
5.4	AMN piéton	79
5.5	AMN bicyclette	81
5.6	AMN véhicule urbain	83
5.7	AMN - Variables statistiques réseau	85
6.1	Gain de retransmission 50 % : 12 au lieu de 24 : (a) sans MPR, (b) avec MPR	95
6.2	2 mécanismes d'optimisation énergétique utilisant OLSR	96
A.1	OPNET Modeler	124
A.2	Notre version d'OPNET Modeler	125
A.3	Mentions légales OPNET Modeler à destination des Académies	125
A.4	L'Interface Homme-Machine (IHM) d' <i>OPNET Modeler</i>	126
A.5	Projet Exemple Random_mobility d'Opnet Modeler	128
A.6	Editeur d'attributs de l'objet Mobility Config	129
A.7	Éditeur d'attributs avancé de l'objet Mobility Config	133

A.8 Mise en garde concernant les modifications de modèle d'objet <i>Opnet</i>	134
A.9 Implementation de Manhattan dans Opnet Modeler	141
A.10 Le système de coordonnées global d'Opnet Modeler	143
A.11 Le système de coordonnées relatif au sous-réseau d'Opnet Modeler	144
A.12 Le système de coordonnées géocentrique d'Opnet Modeler	145
A.13 Erreur de conversion dans Opnet Modeler X 1000	146
A.14 Boîte de dialogue Icon View d'Opnet Modeler	148
A.15 Boîte de dialogue Configure Palette d'Opnet Modeler	148
A.16 Boîte de dialogue Select Included Entries d'Opnet Modeler	149
A.17 Configure Palette après ajout de l'objet <code>mobility_config_ext</code>	150
A.18 Configure Palette après ajout de l'objet <code>mobility_config_ext</code>	151
A.19 Boîte de dialogue Derive New Node Model d'Opnet Modeler	151
A.20 Boîte de dialogue Edit Derived Model d'Opnet Modeler	152
A.21 Boîte de dialogue Icon Palette d'Opnet Modeler	152
A.22 Boîte de dialogue Image Editor d'Opnet Modeler	154
A.23 Boîte de dialogue Color Palette Editor d'Opnet Modeler	154
A.24 Icône distinctive de notre nouvel objet <i>Mobility Config Ext</i>	155
A.25 Fenêtre de l'éditeur de modèle de nœud d'Opnet	159
A.26 Fenêtre de l'éditeur de processus d'Opnet	160
A.27 Boîte de dialogue Model Attributes Initiale de <code>mobility_config_ext</code>	160
A.28 Boîte de dialogue Model Attributes de <i>Mobility Config Ext</i>	161
A.29 Modification du code de l'état Init du processus <i>Mobility Config Ext</i>	161
A.30 Boîte de dialogue de compilation réussie	162
A.31 Boîte de dialogue Rename/Merge Attributes : <i>Mobility Config Ext</i>	163
A.32 Boîte de dialogue Random Mobility Definition Properties	164

TABLE DES FIGURES

A.33 Boîte de dialogue Random Mobility Definition Properties 165

Chapitre 1

Introduction

Cette thèse présente de nouveaux modèles de mobilité sociaux, et montre les apports par rapport aux modèles existants, en soulignant le fait qu'en se rapprochant des comportements sociaux au travers de l'utilisation de la notion de Points d'Attraction, cela change considérablement les résultats des tests de performances des protocoles de routage des réseaux mobiles. Cela nous a amené à proposer une orientation sociale au protocole *OLSR*.

À leurs débuts, les réseaux de télécommunications étaient pensés de manière statique. En effet, à cette époque, les besoins des utilisateurs en termes de mobilité n'avaient pas du tout été envisagés. À l'aire du *Multimedia Mobile*, l'essor des réseaux sans fil, comme réponse des opérateurs de télécommunications aux besoins de mobilité des utilisateurs est fulgurant. Les différentes générations de réseaux cellulaires se sont succédées dans l'unique objectif d'offrir, d'une part, une couverture toujours plus étendue avec l'introduction de l'itinérance (*roaming*) et, d'autre part, des débits toujours plus rapides. La mobilité est aujourd'hui devenue un sujet primordial. Un exemple des thèmes de recherche le plus couramment cité est la prise en compte des déconnexions intempestives dues aux déplacements des appareils portables. Par ailleurs, l'expression de besoin des applicatifs couvre peu ou prou les aspects mobiles. Cela est symptomatique du malaise des industriels vis à vis d'un domaine qui manque cruellement d'un langage communément admis, sur l'ensemble du cycle de vie des infrastructures concernées. De plus l'augmentation significative du nombre d'utilisateurs mobiles a considérablement changé la donne en matière de gestion et de mainte-

nance des réseaux.

Dans la littérature, le terme *mobilité terminale* est très présent. L'idée sous-jacente peut s'exprimer de façon métaphorique en se représentant un aveugle évoluant dans notre univers en en élaborant sa propre représentation, à l'aide de son bâton. Un nœud évoluant dans un *réseau mobile de base* se comporte de la même manière. La transmission périodique par les autres nœuds, de messages balises (*beacons*), permet à cet aveugle de découvrir de proche en proche, l'« implantation » des informations dans le réseau. Dès lors, plusieurs méthodes ont été élaborées dans le but de réduire cet aveuglement et, plus généralement, l'impact de la mobilité sur la Qualité de Service. Ces méthodes peuvent prendre appui sur les services de localisation ou les modèles de mobilité qui, les uns comme les autres, octroient à un nœud mobile des indications cruciales concernant son environnement.

Un modèle de mobilité est donc destiné à décrire, en termes d'environnement, le mode de circulation des nœuds mobiles. Le défi est de trouver des modèles, le plus possible, fidèles aux comportements réels des utilisateurs. Par exemple, il n'est pas envisageable de modéliser le déplacement d'usagers d'un bus scolaire et d'utiliser le même modèle pour le déplacement de businessmen dans un centre d'affaires. Les modèles aléatoires encore trop souvent utilisés dans le testing des protocoles réseau sont un pis-aller pour « testeurs en manque de temps ». En effet, il n'y a rien à préciser dans le paramétrage d'un tel modèle et aucune connaissance préalable n'y est requise. Mais, les mesures de performances obtenues avec ces méthodes sont biaisées. Car, les appareils mobiles sont portés par des utilisateurs qui ont un comportement social, des habitudes etc.. Cela nous a amené tout naturellement à l'idée d'inclure des éléments de cette composante sociale dans la modélisation de la mobilité.

Dans cette thèse, nous traitons tout d'abord des limites des modèles de mobilité existants et les classifions. Ensuite, nous définissons nos propres modèles sociaux et les implémentons afin d'en mesurer l'impact sur les performances d'un réseau mobile simulé. Nous montrons l'influence de ces critères et l'importance d'un choix adéquat du modèle de mobilité utilisé dans l'environnement de test des réseaux de communication. Pour revenir à l'exemple de l'aveugle, il s'agit de lui donner les moyens de connaître les uses et usages de son environnement *a priori* au lieu de le laisser les découvrir à tâtons *a posteriori* à

grands frais. Enfin nous avons souhaité élargir notre spectre en montrant comment le fait d’octroyer la perception de regroupement social à un protocole de routage réseau peut en améliorer les performances.

La suite de cette thèse est organisée comme suit. Dans le chapitre 2, « *État de l’art* », nous dressons un état de l’art des modèles de mobilité. Nous commençons par les méthodes de classifications les plus courantes. Ensuite, nous décrivons les modèles de mobilités les plus utilisés. Enfin, nous introduisons notre classification sociale des modèles de mobilité.

Dans le chapitre 3, « *La mobilité sociale* », nous présentons notre première classe de modèles de mobilité sociale, *Social Mobility Models*, qui s’articule autour des notions de socialisation et de Points d’Attraction sociaux. Nous discutons les réseaux sociaux et leur importance dans la modélisation de la mobilité des usagers d’un réseau mobile. Nous présentons la notion de Points d’Attraction sociaux sur laquelle repose notre classe de modèles de mobilité, ensuite nous présentons sa modélisation. Comme exemple d’application pratique, nous déclinons notre modèle *Social Manhattan Mobility Model (SMN)* et décrivons son implémentation sous *Opnet Modeler*. Enfin nous présentons nos configurations de test et discutons les résultats associés.

Dans le chapitre 4, « *Les réseaux pondérés* », nous présentons notre nouvelle classe de modèles de mobilité sociale pondérée, *Weighted Social Mobility Models*. Nous introduisons d’autres aspects de la socialisation résultant de l’appartenance à une communauté et des centres d’intérêts de ses membres. Dans cette nouvelle proposition, les valeurs sociales affectées aux Points d’Attraction changent au fil du temps afin de rendre compte de ces nouvelles caractéristiques sociales. Nous commençons par la description des réseaux pondérés et leurs applications dans le domaine des réseaux mobiles et nous en citons quelques exemples. Ensuite, nous présentons les Points d’Attraction sociaux propres à notre nouvelle classe de modèles de mobilité ainsi que sa modélisation. Comme exemple d’application pratique, nous déclinons notre modèle *Weighted Social Manhattan Mobility Model (WMN)* et décrivons les différentes configurations de test ainsi que les résultats obtenus.

Dans le chapitre 5, « *Les systèmes de colonies de fourmis* », nous introduisons les systèmes de colonies de fourmis, *Ant Colony Systems (ACS)*. Nous présentons notre nouvelle classe de modèles de mobilité sociale basée sur les systèmes de colonies de fourmis, *Ant*

Mobility Models. Nous dressons un état de l'art des *ACS* et du modèle mathématique associé. Ensuite, nous présentons notre nouvelle classe de modèles de mobilité sociale basée sur les colonies de fourmis *Ant Mobility Models* ainsi que sa modélisation. Un exemple d'application *Ant Manhattan Mobility Model (AMN)* est proposé. Enfin, nous synthétisons nos modèles dans une vue d'ensemble afin de rendre plus claire l'évolution de notre démarche. Les configurations de test et les résultats associés seront présentés et discutés.

Dans le chapitre 6, « *Les protocoles de routage sociaux* », nous abordons les protocoles de routage en tant que nouvel espace prospectif alternatif à l'utilisation des comportements communautaires dans les réseaux mobiles. Après un état de l'art, nous signalons des résultats intéressants concernant un sujet d'actualité : la gestion de l'énergie dans le protocole de routage *Optimized Link State Routing Protocol : OLSR, e-OLSR*. Ensuite, nous passons au cœur de ce chapitre qui est une approche sociale du protocole *OLSR* et proposons notre version sociale de *OLSR* : *OLSR-S*. Nous testons cette approche par la simulation d'un réseau mobile mettant en œuvre ce protocole et relatons les résultats des mesures réalisées sur les différents organes de notre réseau mobile simulé.

Les publications afférentes à cette thèse sont listées dans le chapitre 7, « *Publications* ».

Enfin, en « *Conclusions et perspectives* », nous concluons en résumant l'ensemble de ce travail et en donnant les perspectives de recherche qu'il offre.

Nous joignons en annexes les procédures associées à l'implémentation de nos différents modèles et de notre protocole de routage :

Annexe A : Choix relatifs à la simulation et implémentation de nos modèles sous *Opnet Modeler*

Annexe B : Implémentation de *OLSR-S* sous *Opnet Modeler*.

Chapitre 2

État de l'art

2.1 Introduction

Avec les avancées récentes des technologies de communication sans fil, les réseaux mobiles sont aujourd'hui omniprésents. Un réseau mobile est un ensemble de nœuds mobiles (NM) reliés par ondes hertziennes. L'ensemble de ces nœuds peut se déplacer et forme une topologie arbitraire qui peut changer rapidement. Si les nœuds mobiles représentent des humains, ces déplacements sont souvent répétitifs et peuvent être corrélés à certaines caractéristiques propres à l'humain et d'une certaine manière reproductibles en laboratoire.

Or, cette reproductivité est capitale dans l'élaboration des environnements de test de nouveaux protocoles ou de nouvelles applications pour réseaux mobiles. Elle est formulée formellement en *modèle de mobilité*. Nous en trouvons chez Mousavi et al. [2007] une définition concise et précise : « *Un modèle de mobilité est destiné à décrire le mode de circulation des utilisateurs mobiles et la façon dont leurs localisation, vitesse et accélération évoluent dans le temps* ».

Un même modèle ne pouvant a priori s'appliquer à toutes les situations (déplacement d'un autobus scolaire, déplacement d'individus dans un centre d'affaires, etc.), de nombreuses propositions ont été publiées, afin de fournir des solutions à des classes de déplacements spécifiques (transport motorisé, pédestre etc.). Néanmoins, une difficulté demeure, à savoir celle d'associer le bon modèle ou la bonne classe de déplacement à un cas d'espèce qui est souvent une combinaison : une même personne pouvant très bien (et c'est en fait notre quotidien) se déplacer à pied puis en voiture puis de nouveau à pied puis prendre le

bus, le tram, le métro, le train puis à nouveau se déplacer à pied.

Pour vaincre cette difficulté, la fourniture d'un méta-modèle, c'est-à-dire un modèle de modèles, capable d'établir cette adéquation serait l'idéal. Mais, cela est à l'heure actuelle hors de notre portée.

Bien que cela mène à des mesures de performances sans grande signification, cette difficulté a conduit la communauté scientifique à se cantonner aux modèles de mobilité aléatoires. Lesquels ont fait l'objet de plusieurs tentatives de classification.

Fiore et al. [2007] distingue la *micro-mobilité* qui traite du lieu simulé (rues, intersections, feux) et la *macro-mobilité* qui traite du nœud mobile (déplacement aléatoire, prédéfini ou calculé; comportement individuel, de groupe, social). D'un autre côté, il propose de découper les modèles de mobilité en *modèles stochastiques* qui contraignent un nœud (ou un groupe) selon un graphe prédéfini représentant le lieu simulé, *modèles de trafic* qui sont assimilés à un phénomène hydrodynamique (fonction des vitesses, densité, débit, espace et temps), *modèles de poursuite* où le déplacement d'un nœud est fonction de son prédécesseur, *modèles à files d'attente* qui représentent les voies de circulation, *modèles comportementaux* où le mouvement est déterminé par des règles sociales ou suit un processus stimulus-réaction (Fig. 2.1).

Camp et al. [2002] considère les modèles *d'entités* où un nœud se déplace indépendamment des autres et *de groupe* où les nœuds se déplacent en groupe.

Bai and Helmy [2007], Fig. 2.2, voit des modèles *aléatoires* où un nœud est totalement libre, à *dépendances temporelles* où la vitesse courante d'un nœud est fonction de ses vitesses précédentes, à *dépendances géographiques* où l'environnement joue un rôle déterminant (véhicules sur autoroutes, routes nationales ou en milieu urbain; piétons bloqués par les bâtiments) ou à *dépendances spatiales* où un nœud peut être influencé par ses voisins (collaboration au sein d'une équipe, leadership).

Enfin, la classification de Haerri et al. [2005], Fig. 2.3, distingue les modèles *synthétiques*, *d'observation*, à *traces* et *de trafic*.

Dans ce qui suit, après avoir cité les méthodes de classifications les plus usuelles, nous allons décrire plus en détail les modèles de mobilités les plus courants selon la classification

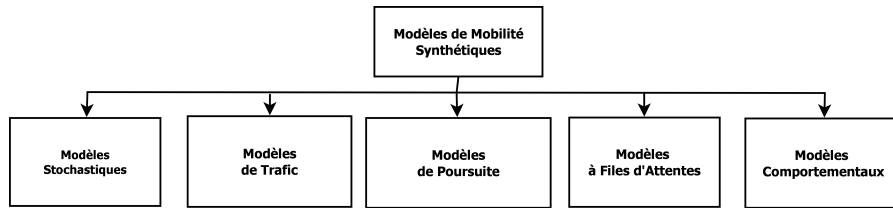


FIGURE 2.1 – Classification de Fiore

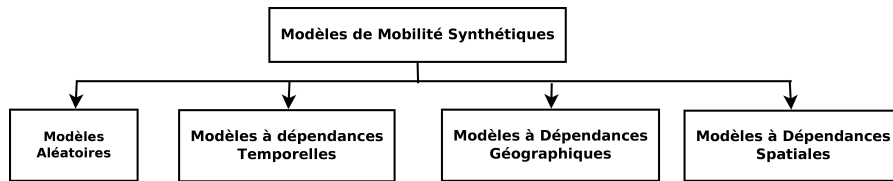


FIGURE 2.2 – Classification de Bai

de Haerri Fig. 2.3 qui nous paraît assez exhaustive. Dans la section 2.2 les *modèles de mobilités synthétiques* les plus usuels (d'entités et de groupe) sont présentés, *les modèles d'observation* sont décrits en section 2.3, *les modèles à traces* sont exposés en section 2.4, ensuite *les modèles de trafic* sont décrits en section 2.5. Dans la section 2.6 nous introduisons notre *classification sociale* des modèles de mobilité. Enfin la section 2.7 conclue ce chapitre.

2.2 Modèles Synthétiques

La première classe de modèles de mobilité, et celle qui est la plus connue et étudiée, car aisément reproductible en laboratoire, est la classe des modèles synthétiques. Elle inclut

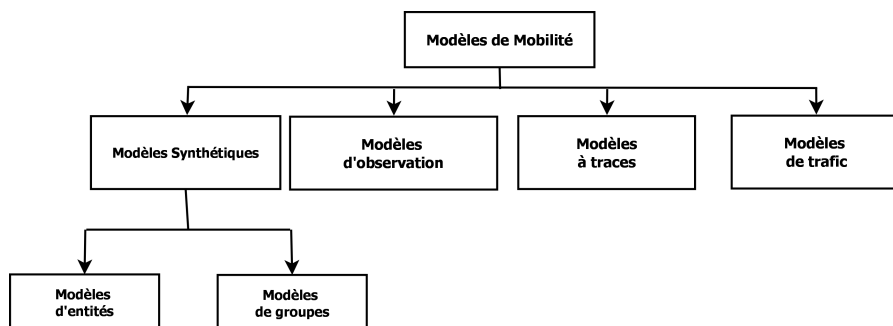


FIGURE 2.3 – Classification de Haerri

les modèles mathématiques où le déplacement d'un nœud est une fonction mathématique. Le comportement des nœuds mobiles peut être *individuel* où le déplacement d'un nœud du réseau est défini indépendamment des autres ou *collectif* dans le cas contraire. On parle respectivement de modèles *d'entités* ou *de groupe*.

2.2.1 Modèles d'entités

Les deux premiers modèles d'entités cités ci-dessus sont les plus utilisés car simples à mettre en œuvre.

Dans *Random Walk Mobility Model (RW)*, les vitesse $v \in [v_{min}; v_{max}]$ et direction θ sont aléatoires. Au bout d'un temps t_{max} ou d'une distance d_{max} , un nouveau tirage aléatoire des vitesse et direction est fait. Quand un nœud mobile arrive aux bornes de la zone de simulation, il rebrousse chemin selon un angle qui dépend de l'angle d'arrivée. Si t_{max} et/ou d_{max} sont choisis trop petits, le mouvement des nœuds mobiles reste confiné à une petite partie de la zone de simulation située autour du point initial. De plus, les arrêts et changements de direction sont trop brusques pour être naturels.

Random Waypoint Mobility Model (RWP), Fig. 2.4a, tente d'y remédier en ajoutant un *temps de pause* entre les changements de vitesse et de direction. Camp et al. [2002] recommande de partir d'une configuration issue d'une simulation antérieure ou de ne pas prendre en compte les premières minutes de simulation, car dans ce modèle la distribution initiale des nœuds est cruciale. De plus, on remarque une concentration des nœuds au centre de la zone de mobilité

Pour y pallier, *Random Direction Mobility Model (RD)*, Fig. 2.4b, ajoute l'interdiction à un nœud de changer ses vitesse et direction si ce n'est aux bords de la zone de simulation.

Dans *Boundless Simulation Area Mobility Model*, les bords opposés de la zone de simulation rectangulaire se touchent virtuellement. De sorte que, arrivé à un bord, un nœud « sort » de la zone de simulation et au même moment y « pénètre » par le bord opposé. Il s'agit d'un modèle à *mémoire*, i.e. les direction et vitesse choisies prennent en compte les direction et vitesse précédentes. Fig. 2.5.

Gauss-Markov Mobility Model (GM) utilise les chaînes de Markov pour faire varier le

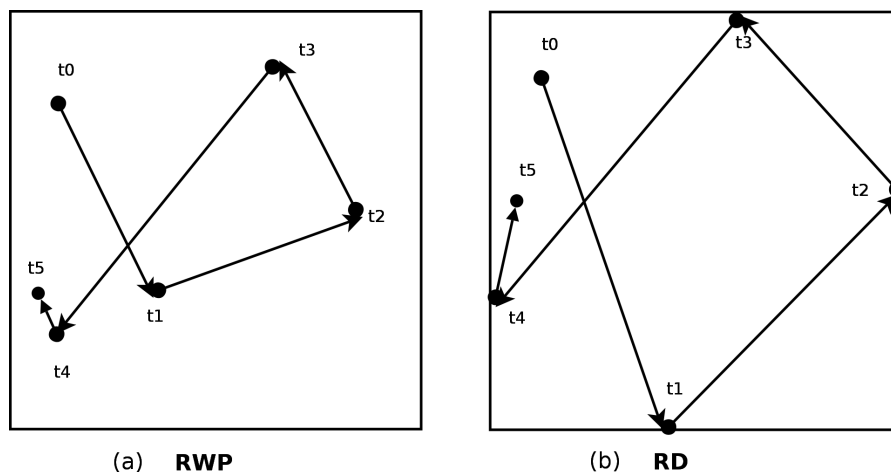


FIGURE 2.4 – Mouvement d'un nœud selon RWP et RD

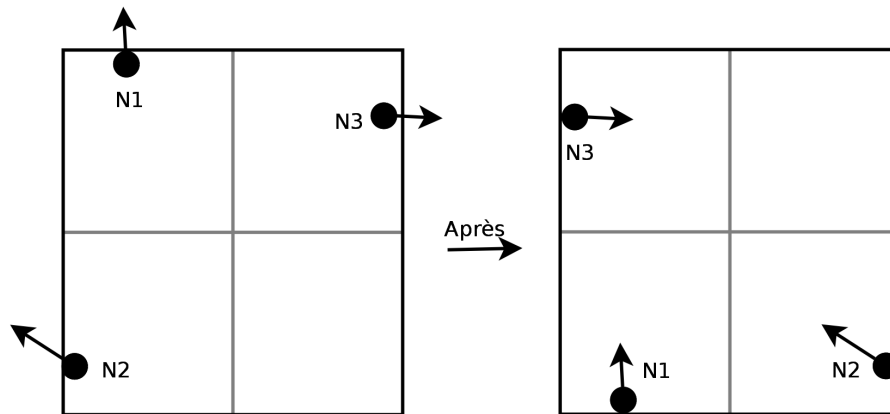
tirage aléatoire des vitesse et direction du modèle *Random Walk* tout en tenant compte des vitesse et direction précédentes en les faisant évoluer suivant un processus markovien. Ceci évite les changements brusques de vitesse et de direction.

Probabilistic Version of the Random Walk Mobility Model dérive du *Random Walk Model* et utilise les probabilités pour définir la position suivante.

Chiang and Shenoy [2001] définit le *Chiang Model*. C'est un modèle à *mémoire*. Un nœud mobile mémorise 3 positions : $(pos_{t-1}, pos_t, pos_{t+1})$. La probabilité $P(pos_a, pos_b)$ prédéfinie entre chaque paire de positions est la probabilité qu'un nœud passe de la position p_a à la position p_b . Ces probabilités sont utilisées pour calculer la nouvelle position d'un nœud. Ce modèle supprime les changements soudains. Son point faible est la difficulté de choix des probabilités.

Chiang and Shenoy [2004] définit aussi le *City Section Mobility Model* qui représente un quartier avec ses rues. Il se rapproche de la modélisation des déplacements urbains de véhicules. La zone de simulation comprend les rues, les obstacles, les limitations de vitesse, les feux... Très proche de la réalité, son paramétrage ainsi que l'implémentation des contraintes restent lourds.

Dans *MANHATTAN Mobility Model*, la zone de simulation représente des rues bidirectionnelles. Les nœuds se déplacent en lignes et colonnes. À une intersection, un nœud choisit



Boundless Area

FIGURE 2.5 – Réinjections des nœuds dans le Boundless Area Mobility Model

de continuer tout droit avec une probabilité $1/2$ ou tourner à gauche ou à droite avec une probabilité $1/4$, Fig. 2.6.

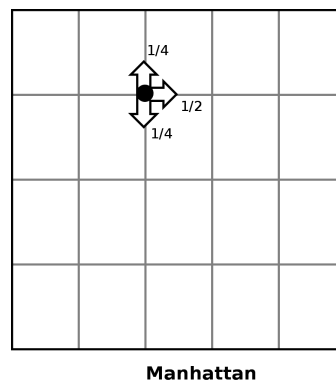


FIGURE 2.6 – Mouvement d'un nœud selon MANHATTAN

2.2.2 Modèles de groupes

Au lieu de modéliser le mouvement de chaque nœud mobile séparément, il est parfois judicieux est plus réaliste de modéliser des groupes qui suivent un même comportement selon une stratégie donnée.

Exponential Correlated Random Mobility Model est l'un des premiers modèles de groupes. Le groupe est vu comme un bloc en déplacement, étant donnée une position du groupe à un temps t , une fonction est utilisée pour définir la position à $t + 1$.

Column Mobility Model représente une ligne/colonne de nœuds qui suivent la même direction, par exemple des soldats en file indienne. Chaque nœud a son propre point de référence autour duquel il a un certain degré de liberté. Ces points de référence sont alignés et mis à jour régulièrement. Pour le référencement, la zone de simulation est découpée en une grille.

Nomadic Community Mobility Model offre un plus grand degré de liberté de mouvement que *Column Mobility Model*.

Pursure Mobility Model représente un groupe qui suit une cible : course poursuite. La nouvelle position est fonction de l'ancienne position et de l'accélération de la cible.

Reference Point Group Mobility Model (RPGM) représente le mouvement aléatoire d'un groupe autour d'un centre logique. Un nœud a un mouvement aléatoire mais confiné au groupe. C'est le modèle le plus général dans la catégorie des modèles de groupes. Le mouvement est effectué sur un chemin qui mène au centre logique du groupe, Fig. 2.7.

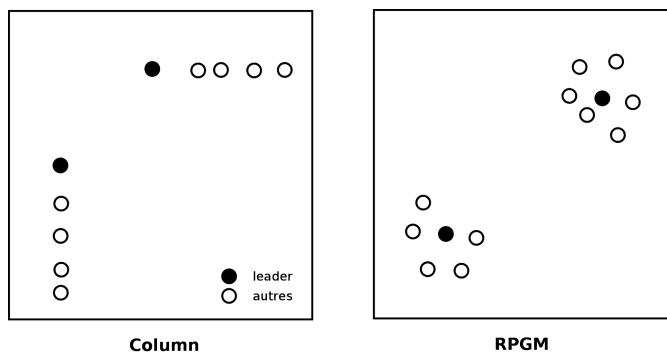


FIGURE 2.7 – Mouvement des noeuds selon Colum Model et RPGM

2.2.3 Conclusion

Les modèles mathématiques sont aléatoires ou quasi-aléatoires. Le choix du modèle et de ses paramètres reste un point crucial. Les modèles individuels *Random Waypoint* et *Random Walk* restent les plus simples et les plus utilisés. Cependant, pour des perfor-

mances spécifiques, on peut passer à leurs dérivés ou aux autres modèles de cette classe. Pour un modèle de groupe, Camp et al. [2002] conseille le *Reference Point Group Mobility Model (RPGM)* avec un paramétrage soigné. Ces modèles sont encore loin de modéliser des déplacements réels d'individus dans un espace donné.

2.3 Modèles d'observation

Une alternative aux modèles synthétiques, est le calibrage par comparaison aux situations réelles au lieu de se contenter de comparer des modèles mathématiques entre eux. On développe ainsi un modèle de mobilité générique qui n'est pas aléatoire mais basé sur des mesures réelles. Dans cette catégorie, le projet Udel [Ude 2011] décrit un modèle générique, *Udel Mobility Model*, basé sur l'observation du comportement de travailleurs durant la journée. Zheng et al. [2006] définit *Agenda-based Mobility Model* qui est lui plus fin car il associe les aspects individuels et sociaux à l'individu.

2.4 Modèles à traces

Un des inconvénients des modèles synthétiques, est que, dans les faits, il existe peu de modèles (même très compliqués), qui puissent se rapprocher de la réalité. Une autre approche est alors née : au lieu de développer des modèles complexes puis les valider en utilisant des traces ou des observations, un temps crucial peut être économisé en prenant directement des traces réelles de mouvement au lieu de modéliser la mobilité.

Cette approche se développe de plus en plus, [Gonzalez et al. 2009]. En effet, malgré son importance dans l'urbanisme, le trafic urbain et la propagation des virus biologiques et mobiles, notre compréhension des lois fondamentales régissant les déplacements humains est très limitée. Cet état de fait est consécutif au manque d'outils de suivi des individus. En étudiant les déplacements de 100000 utilisateurs de téléphones cellulaires durant une période de 6 mois, Gonzalez et al. [2009] trouve que, contrairement aux chemins aléatoires calculés par les modèles de mobilité les plus répandus tels que *Random Walk*, les humains suivent des trajectoires d'un haut degré de régularité temporelle et spatiale. En effet, tout individu se caractérise par une distance moyenne de déplacement et un nombre limité de

lieux qu'il visite fréquemment. Après correction des différences relatives aux distances et à celles dues à l'anisotropie¹ des trajectoires, les formes de trajet individuel se réduisent à une unique distribution de probabilité spatiale. Cela montre que, malgré leur diversité, les déplacements humains suivent des formes simples et récurrentes. Cette similitude dans les modes inhérents aux déplacements pourrait influencer sur tous les phénomènes liés à la mobilité humaine, de la prévention des épidémies aux interventions d'urgence, en passant par l'urbanisme et la modélisation multi-agents.

Citons dans cette catégorie, le projet *SUMATRA* pour *Stanford University Mobile Activity TRAcés*². *Sumatra* est un générateur de traces qui englobe plusieurs modèles d'appels et de mobilité. L'aspect unique de *SUMATRA* est qu'il est validé par des données réelles : traces d'appels et de mobilité. En mettant ces modèles d'appels et de mobilité à la disposition de la communauté sans fil, le groupe de recherche espérait fournir une « référence » commune aux chercheurs.

Une nouvelle source de traces de mobilité réalistes pour la simulation de réseaux inter-véhiculaires est l'utilisation d'un simulateur de trafic multi-agent microscopique (*Multi-agent Microscopic Traffic Simulator (MMTS)*) développé par *K. Nagel* à l'*EPF de Zurich*. Ce simulateur est capable de simuler le trafic public et privé sur les cartes routières régionales réelles de *Suisse* avec un haut niveau de réalisme. Les traces générées par *MMTS* fournissent un point de départ intéressant pour étudier les performances des protocoles de routage dans les réseaux *VANET*. Environ 260000 véhicules sont impliqués dans la simulation avec plus de 25000000 changements de direction ou de vitesse dans une zone d'environ $250km \times 260km$.

Le projet *Crawdad* pour *Community Resource for Archiving Wireless Data At Dartmouth*³ est un réseau sans fil d'archivage de données pour la recherche. La multiplicité des lieux d'archivage contribue à l'essor de ce réseau et une équipe développe des outils pour les collecte, préservation de l'anonymat, et analyse des données.

Enfin, rappelons également que Liao et al. [2004] inspiré par Heck and Ghosh [2000]

1. Qualité d'un milieu transparent dont les propriétés optiques varient selon la direction de la lumière, *Dictionnaire Larousse 2011*. Ici les contingences qui faussent les résultats pour des détails sans importance.

2. <http://infolab.stanford.edu/sumatra/>

3. <http://rawdad.cs.dartmouth.edu/index.php>

définit un modèle qui génère des traces au lieu d'utiliser des modèles de mobilité mathématiques pour engendrer le mouvement des fourmis.

2.5 Modèles de trafic

À nouveau par souci d'économiser l'étape de validation, des simulateurs de trafics réels ont été proposés. PARAMICS Par [2011], CORSIM cor [2011], VISSIM vis [2011] et TRANSMIS tra [2011] utilisent des trafics réels en milieu urbain pour définir des nouveaux trafics très proches des réels. Ceux-ci sont liés aux lieux d'où sont extraites les traces de trafic initiales. À nouveau, il n'y a pas de généralisation ou d'extension possible. Ils sont complexes à paramétrer. De plus, ses modèles sont plutôt destinés à l'industrie car ils ne sont pas libres d'accès (licences) ce qui est une limite pour la recherche. Enfin, il n'existe pas encore (ou peu) d'analyseurs (*parsers*) qui permettent d'intégrer ces modèles dans les simulateurs.

2.6 Classification sociale

La composante sociale est de plus en plus présente et influente dans les modèles de mobilité récents. Nous procédons à la classification des modèles de mobilité par rapport à leurs caractéristiques sociales, Fig. 2.8. L'intérêt de cette classification est qu'elle nous permet de définir si un modèle de mobilité est social et, dans l'affirmative, de savoir précisément la partie concernée par le comportement social et ainsi de pouvoir mieux l'appréhender.

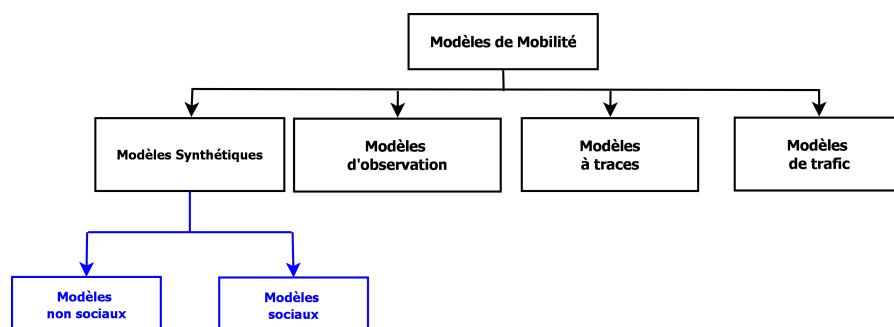


FIGURE 2.8 – Classification Sociale

TABLE 2.1 – Classification Sociale

Modèle	Random Way Point (RWP)	Gauss Markov (GM)	Reference Point Group Mobility (RPGM)	Manhattan
Région	Aléatoire	Aléatoire	Aléatoire	Grille
Nœuds	≥ 2	≥ 2	≥ 2	≥ 2
Communauté	<i>non social</i> → chaque nœud représente une communauté.	<i>non social</i>	<i>social</i> → un groupe de nœuds représente une communauté.	<i>non social</i>
Comportement	<i>non social</i>	<i>non social</i>	<i>social</i> → un groupe de nœuds avec son leader.	<i>non social</i>
Déplacement	<i>non social</i> → tout droit, avec une vitesse aléatoire; → vers une direction aléatoire; → avec des pauses.	<i>non social</i> → la vitesse des nœuds suit un processus markovien.	<i>non social</i> → le leader se déplace selon RWP; <i>social</i> → les autres nœuds se déplacent en suivant le leader.	<i>non social</i> → vitesse des nœuds aléatoire; → se déplacer suivant les lignes/colonnes; → à une intersection, continuer dans la même direction avec la probabilité 1/2 ou de tourner à droite ou à gauche avec la probabilité 1/4.

D'après la table 2.1. Nous remarquons la pauvreté de ces modèles en termes de composantes sociales. En définitive, seul RPGM en dispose.

2.7 Conclusion

Dans ce chapitre, nous avons introduit les modèles de mobilité et la nécessité de leur utilisation dans le domaine des réseaux mobiles.

Malgré l'existence de nombreuses catégories de modèles de mobilité comme décrit dans ce chapitre, deux seules ont perdurées : *les modèles synthétiques* et *les traces réelles*.

Toutefois, la possession de traces réelles est un atout majeur pour celui qui en dispose. Celles-ci ont une réelle valeur marchande. On connaît les récentes tribulations de *Apple Corporation*, concernant la capitalisation des traces de tous ses utilisateurs après que ceux-là ont accepté de leur propre chef d'activer la géolocalisation. C'est la raison pour laquelle la plupart des recherches et la nôtre également portent sur les modèles de mobilité synthétiques qui sont bien éloignés des enjeux financiers des traces.

Enfin, nous avons proposé une nouvelle classification que nous avons qualifiée de sociale dans la mesure où elle tient compte des aspects sociaux introduit dans les modèles. Cette nouvelle classification nous a permis de mettre en perspective les modèles actuels sous un éclairage social et constater leur grande pauvreté en la matière. Cette classification facilite l'élaboration de nouveaux modèles sociaux car elle précise quelle partie du modèle est concernée par quelle caractéristique sociale. Bien entendu, nous nous basons sur cette nouvelle classification pour définir nos propres modèles.

Chapitre 3

La mobilité sociale

3.1 Introduction

La mobilité sociale peut revêtir une multitude de significations.

Alors que nous mettons la dernière touche à notre thèse, nous découvrons Myer [2011] qui nous informe du fait que *AVG*, *Google*, *Nokia* et *Sony Ericsson* viennent de s'associer à la création du *MobiSocial Lab* de l'*Université de Stanford*. Ce laboratoire présente déjà une de ses inventions sous la forme d'une scène sous *Youtube*, où 2 étudiants munis de téléphones intelligents ou *smartphones*, participent à l'élaboration d'un croquis qui non seulement s'affiche simultanément sur les 2 téléphones mais en sus sur un écran de télévision situé en face d'eux. À la lecture de cette brève, nous sommes restés un peu dubitatif au sens donné ici à la mobilité sociale. Mais, c'est donc officiellement un des sens possibles !

Dans nos travaux, c'est bien de la théorie des réseaux sociaux que nous partons laquelle a été largement appliquée à la modélisation de la mobilité [Heck and Ghosh 2000] : un réseau social a été utilisé pour générer un modèle de mobilité pour les réseaux *ad-hoc* pour la mobilité des groupes [Musolesi and Mascolo 2009] ; d'autres modèles de mobilité font déplacer les nœuds suivant des Points d'Attraction [Jardosh et al. 2005]. Haerri et al. [2005] et Ghosh et al. [2005] décrivent les fondements mathématiques, les progrès et les perspectives dans l'analyse de réseaux sociaux complexes. En écrivant ces lignes, nous ne pouvons pas réprimer une pensée pour les habitants de Fukushima et Sendai, qui n'ont que leurs réseaux sociaux pour communiquer, à la suite des tremblements de terre, tsunami et catastrophes nucléaires, dont ils sont l'objet.

Comme le mouvement des appareils mobiles est principalement basé sur les décisions et le comportement des personnes qui les portent, il est important de prendre le comportement social des êtres humains dans la modélisation de la mobilité de ces appareils. On peut remarquer que les mouvements d'un homme peuvent être représentés comme un compromis entre son fort besoin de socialiser avec les autres et ses habitudes et/ou intérêts. Ainsi, les comportements sociaux des humains peuvent être caractérisés par :

- *des Points d'Attraction*, c'est-à-dire endroits qui sont considérés comme importants pour le groupe et qui doivent être visités ;
- *des communautés de voyageurs*, à savoir des personnes dans une communauté donnée qui visitent les mêmes lieux : plus un lieu est visité, plus il est attrayant.

Certains services de localisation divisent la zone de réseau en partitions selon les emplacements physiques des groupes. Toutefois, ceci ne parvient pas à modéliser le comportement social des individus et des groupes. Pour saisir ce type de comportement au sein d'un réseau, il est important de définir des groupes selon les relations entre les personnes qui le composent et les relations entre les gens et leur environnement. Cela permet de se rapprocher d'un mouvement plus réaliste. En conséquence, *la théorie des réseaux sociaux* est de plus en plus largement utilisée dans le domaine des réseaux mobiles. Les fondements mathématiques et les progrès réalisés dans l'analyse des réseaux sociaux complexes peuvent être trouvés dans Eason et al. [1955] et Maxwell [1873]. Les modèles mathématiques de ces réseaux ont été empiriquement démontré être utiles pour décrire de nombreux types de relations, y compris les rapports sociaux réels [Jacobs and Bean 1963].

Dans ce chapitre nous présentons une classe de modèles de mobilité sociale, *Social Mobility Models*, basées sur les notions de socialisation et de Points d'Attraction. La suite de ce chapitre est organisée comme suit ; dans la section 3.2 nous discutons les réseaux sociaux et leur importance dans la modélisation de la mobilité des usagers d'un réseau, en section 3.3 nous présentons la notion de Points d'Attraction sociaux sur laquelle repose notre classe de modèles de mobilité, le modèle mathématique associé est présenté en section 3.4. Comme exemple d'application pratique, nous décrivons notre modèle *Social Manhattan Mobility Model (SMN)* dans la section 3.5. Son implémentation sous *Opnet Modeler* est décrite en section 3.6. Dans la section 3.7 nous présentons nos configurations de test dont les résultats

sont décrits en section 3.8, Enfin la section 3.9 conclue ce chapitre.

3.2 Réseaux sociaux

Un réseau social peut être défini comme l'ensemble des relations entre les personnes à l'intérieur d'un groupe. Les relations entre les gens peuvent survenir pour différentes raisons : ils peuvent partager les mêmes intérêts, pour un long terme ou occasionnellement, ils peuvent se déplacer dans la même direction, ils peuvent avoir les mêmes références, et ainsi de suite. Dans un tel groupe, les gens peuvent être ou ne pas être au courant de l'appartenance à une même communauté, ou le réseau social peut être trop grand pour permettre à tous les membres de se connaître.

Le comportement social des utilisateurs dans un réseau mobile peut être classé selon les caractéristiques et/ou orientations spécifiques suivantes :

- Les utilisateurs partagent la même application, comme un jeu distribué, le comportement est axé sur l'application. Dans ce cas, il n'y a pas de contrainte liée à la localisation et les mouvements des utilisateurs ne sont pas nécessairement liés ;
- Les gens suivent un individu du groupe comme un guide touristique. Encore une fois, les contraintes de localisation et les mouvements de mobiles sont liés ;
- Les mobiles en question sont localisés dans la même région et/ou sont attirés par un ou plusieurs endroits, comme une station de métro ou un centre de conférence. Dans ce cas, le comportement est dit *orienté points-d'attraction-sociaux*. Ici, les contraintes de localisation et les mouvements de mobiles sont généralement liés ;
- Les gens visitent les lieux les plus connus ou visités (les plus attractifs). La popularité des lieux et du mouvement des nœuds sont étroitement liés.

Il est très difficile de modéliser des comportements basés sur l'application comme le comportement des utilisateurs dépend en grande partie de l'application car il y a généralement très peu de points communs entre deux applications. Ce chapitre se concentre principalement sur les comportements *orientés points-attraction-sociaux*.

Les comportements *orientés points-attraction-sociaux* reposent sur la notion des Points d'Attraction. Dans le cas non social, les Points d'Attraction qui sont dans le domaine des

utilisateurs sont censés les attirés « un » à la fois en fonction de leurs valeurs d'*attractivité* respectives. Cette *attractivité* est affectée aléatoirement aux Points d'Attraction. D'autre part, dans le cas social, les individus suivent des Points d'Attraction, dont les valeurs d'*attractivité* dépendent de leurs popularités respectives. Si un point est visité, c'est qu'il a une « grande » *attractivité*.

Ce comportement (popularité d'un lieu) doit, à notre sens, impérativement être pris en compte dans la modélisation de la mobilité.

3.3 Points d'Attraction (PA)

En effet, on peut remarquer par exemple que lors d'achats sur *Internet*, un produit classé comme le plus acheté, sera certainement plus attirant et plus de personnes vont se précipiter pour l'acheter. Cela est également vrai au moment de choisir une destination de vacances, ou d'un lieu à visiter : nous pensons que si tant de gens y vont, il devrait être intéressant. Un autre exemple plus frappant, c'est au cours de la période de soldes : plus un magasin est peuplé, plus les gens y entrent parce qu'ils pensent (consciemment ou non) que si il est si encombré, c'est qu'il doit y avoir beaucoup de bonnes affaires.

Dans la littérature, les Points d'Attraction peuvent modéliser ce genre de centres d'intérêt [Jardosh et al. 2005]. Nous utilisons donc cette notion de Points d'Attraction et y introduisons des comportements sociaux.

Pour modéliser le comportement cités ci-dessus et l'appliquer au domaine de la modélisation de la mobilité, la première étape consiste à identifier les *Points d'Attraction (PA)* du réseau.

Chaque PA sera ensuite doté d'une valeur sociale (*attractivité*) qui représente sa popularité relative aux autres points. Cette valeur sociale est proportionnelle à la réputation du PA et elle prend en compte le comportement grégaire des utilisateurs qui pensent généralement que quelque chose est intéressante si les gens dans leur réseau social pensent qu'elle l'est.

Chaque individu du réseau va se diriger vers le Points d'Attraction le plus proche de lui et ayant la plus grande valeur d'*attractivité*.

3.4 Modélisation

Les personnes appartenant à un même groupe social partagent en général des intérêts communs. Ces « centres d'intérêt » sont plus visités que tous les autres endroits. Nous décrivons ici le modèle mathématique lié à notre première classe de modèles de mobilité sociale, *Social Mobility Models*. Dans nos modèles de mobilité sociale, nous considérons des Points d'Attraction avec des valeurs d'attractivité fixes, ces valeurs d'attractivité représentent leurs popularités respectives. Un nœud mobile (NM) peut visiter un sous-ensemble de Points d'Attraction qui correspond à ses centres d'intérêt. Il visite toujours le Point d'Attraction le plus proche et le plus intéressant pour lui (i.e. avec la plus grande valeur d'attractivité). Chaque nœud mobile visite chaque Point d'Attraction une et une seule fois puis se dirige vers un autre point (plus proche, plus attractif). La Fig. 3.1 illustre ce comportement.

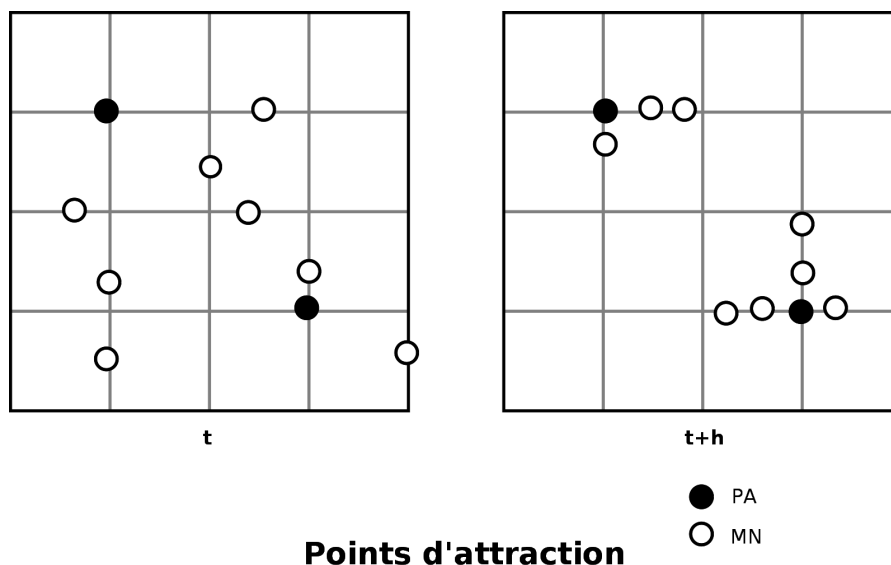


FIGURE 3.1 – Points d'attractions

Nous modélisons cela mathématiquement comme suit. Soit N , l'ensemble des nœuds d'un réseau sans fil. Soit A , un ensemble de Points d'Attraction. Soit T , l'ensemble des points du temps, tels que $t \in T \iff t = b\Delta$ avec $b \in \mathbb{N}$, $\Delta \in \mathbb{R}$ et $t \leq D$, i.e. la durée de simulation. Soit $r : A \rightarrow [0; 1]$, la normalisation de la valeur d'attractivité $u(a)$ avec $a \in A$, telle que :

$$r(a) = \frac{u(a)}{\sum_{a' \in A} u(a')}$$

avec :

$$u(a) = \begin{cases} user_value & \text{si défini par l'utilisateur} \\ \frac{1}{|A|} & \text{sinon} \end{cases} \quad (3.1)$$

Soit $d : A \times N \times T \rightarrow \mathbb{R}$, la distance euclidienne du Point d'Attraction a au nœud n à l'instant t . Soit $v : A \times N \times T \rightarrow \{V, F\}$ où V et F signifient VRAI et FAUX, l'application qui définit l'ensemble des Points d'Attraction à visiter à l'instant t par le nœud n :

$$v(a, n, t) = \begin{cases} T & \text{si non défini par l'utilisateur} \\ F & \text{si visité} \end{cases}$$

Finalement : n attiré par $a \iff$

$$a = \arg \max_{a' | v(a', n, t) = V} \frac{r(a')}{d(a', n, t)} \quad (3.2)$$

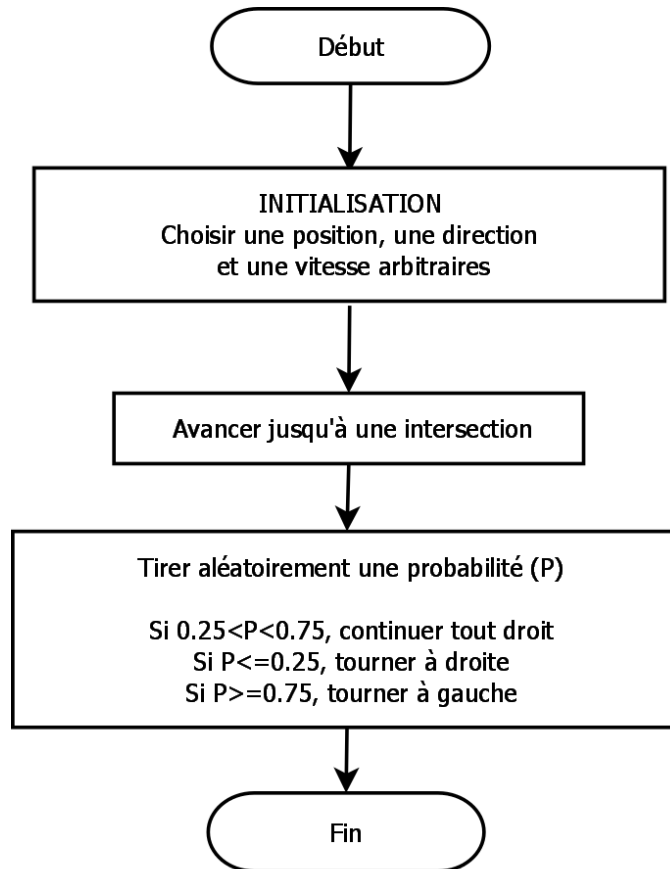
3.5 Application : Social Manhattan (SMN)

Notre première classe de modèles de mobilité sociale, *Social Mobility Models*, regroupe tous les modèles de mobilités *classiques* en y introduisant nos Points d'Attraction comme décrit dans le modèle ci-dessus. Comme exemple d'application pour notre première classe de modèles de mobilité sociale et dans le cadre de simulation, nous implémentons nos Points d'Attraction sociaux sur le modèle de mobilité *Manhattan Grid Mobility Model* de Jayakumar and Gopinath [2008]. Ce modèle décrit le mouvement des nœuds dans une ville à l'aide d'une grille représentant des routes à double sens qui se croisent à angle droit. Ce modèle est très utilisé parce que la ville est une zone de mobilité par excellence en ce sens où, de par le monde, elle est bien un lieu très fréquenté par opposition aux hameaux. Ce choix de *Manhattan* pour cadre de travail, n'est cependant pas le seul possible. La notion de Points d'Attraction sociaux peut s'appliquer à tous les modèles de mobilité.

La grille de Manhattan est définie par ses coordonnées *min* et *max* sur le terrain. Les lignes et les colonnes sont séparées d'une taille en mètres. À une intersection, un nœud

mobile continue tout droit avec une probabilité $1/2$, tourne à gauche ou à droite avec une probabilité $1/4$ pour chaque direction comme indiqué dans Jayakumar and Gopinath [2008].

La Fig. 3.2 présente un schéma de principe de Manhattan.



MANHATTAN

FIGURE 3.2 – Schéma de principe de Manhattan

Notre modèle *Social Manhattan Mobility Model (SMN)* [Harfouche et al. 2009] introduit le modèle mathématique décrit dans la section précédente dans le modèle Manhattan. Se référer au paragraphe 3.3, pour les motivations.

La Table 3.1 décrit *SMN* en les termes de notre classification des modèles de mobilité et la Fig. 3.3 présente son schéma de principe.

TABLE 3.1 – SMN

Région	Topologie en Grille
Nœuds	$N \geq 2$
Communauté	<i>social</i> : → avoir les même centres d'intérêts ;
Comportement	<i>social</i> : → attirés par les mêmes Points d'Attraction (PA) ;
Déplacement	<i>social</i> : suivant Harfouche et al. [2009] → vitesse aléatoire des nœuds ; → se déplacer suivant une ligne/colonne ; → à une intersection, choisir de se déplacer vers le plus proche ; plus intéressant Point d'Attraction ; → PA avec une <i>attractivité</i> fixe .

3.6 Implémentation

Cette section est un résumé du travail d'implémentation décrit en détail à l'annexe A.

Afin d'évaluer nos modèles, nous devons choisir :

1. un cadre de travail parmi ceux existant en matière de mobilité ;
2. un outil de simulation.

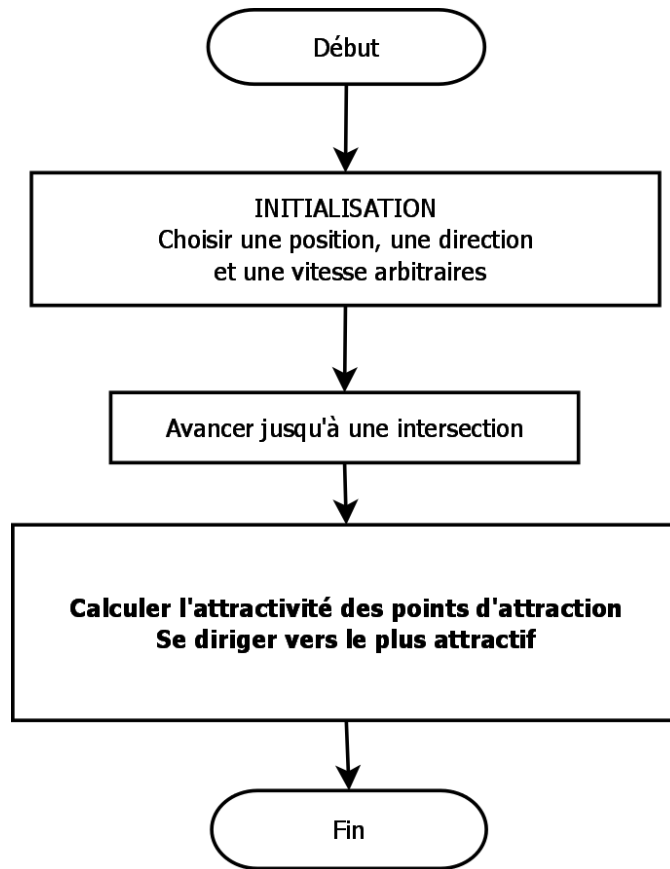
Le cadre de travail que nous choisissons est celui du modèle de mobilité *Manhattan* car de très loin le plus répandu dans la littérature.

L'outil de simulation que nous choisissons est *Opnet Modeler* car il s'agit d'un outil commercial abouti tant du point de vue du fond que de la forme. Sur le fond, il suit de près les dernières évolutions des réseaux d'entreprise. Sur la forme, il propose un certain nombre d'éditeurs et notamment une interface homme-machine visuelle d'élaboration des modèles de simulation.

Dans la version d'*Opnet Modeler* destinée aux académies, seul le modèle *Random Way-point (RWP)* est disponible. Il nous faut donc développer nous-même le modèle *Manhattan*.

Initialement, nous pensons pouvoir prendre comme point de départ les principes de développement utilisés dans *RWP*.

Toutefois, nous sommes confrontés au *problème des intersections manquées* ce qui représente un verrou technique majeur.



Social Manhattan

FIGURE 3.3 – Mouvement d’un nœud selon Social Manhattan : Schéma fonctionnel

Un exposé détaillé du problème et de son solutionnement est présent à l’annexe A.

Brièvement, *Opnet Modeler* traduit toutes les coordonnées dans son *système de coordonnées géographiques*.

Pour tous les modèles de mobilité aléatoires sans exception, les erreurs relatives aux conversions du *système de coordonnées cartésiennes* vers celui-ci passent tout à fait inaperçues.

En revanche, dès qu’on passe à un modèle de mobilité où les décisions sont prises aux intersections (continuer tout droit ou tourner ; se diriger vers un Point d’Attraction ou un autre...), le fait de « manquer » une intersection fausse tous les résultats ! C’est exactement

le problème auquel nous faisons face.

Le solutionnement proposé consiste à ne pas s'appuyer sur l'infrastructure prévue par *Opnet* pour les modèles de mobilité, mais de créer nous-même une nouvelle infrastructure autour du *système de coordonnées cartésiennes*.

Nous commençons l'implémentation du modèle de mobilité *Freeway* [Divecha et al.], Fig. 3.4c.

Le modèle *Freeway* est un modèle uni-directionnel (chaque ligne/rue ayant une *Orientation*). Périodiquement, un nœud choisit une vitesse et se déplace d'une étape correspondant à *Move Step* mètres. Sa vitesse est aléatoire, sauf quand il rentre dans la distance de sécurité de son prédécesseur, *Safety Distance*. Dans ce cas, la vitesse est divisée par 2. Aucun nœud ne peut dépasser un autre nœud.

Sur cette base, nous développons le modèle *Manhattan*, Fig. 3.4d. Il suppose une grille représentant des routes bi-directionnelles (plus besoin d'*Orientation* de la ligne/colonne). L'aire de simulation est définie par ses coordonnées *min* et *max* sur le terrain (x_{min}, y_{min}), (x_{max}, y_{max}). Les lignes et les colonnes sont séparées par la longueur d'un pâté de maison, *Block Length*, en mètres. À une intersection, un nœud continue tout droit avec une probabilité 0,5, tourne à gauche ou à droite avec une probabilité de 0,25 pour chaque coté comme indiqué dans Jayakumar and Gopinath [2008].

Puis vient l'implémentation de notre modèle *Social Manhattan*, voir Fig. 3.4e, 3.5b et 3.6a.

Il ajoute des Points d'Attraction avec des *attractivités* fixes. Ceci est modélisé dans un nouveau composant *Attraction Point*, voir Fig. 3.5b.

3.7 Test

Le caractère socio-culturel des relations humaines est-il un élément déterminant dans la modélisation du déplacement des humains ? Dans les sections précédentes, nous avons apporté des éléments de réponse tant du point de vue de l'observation que de celui des mathématiques, qui nous ont permis de mesurer l'impact que certains caractères sociaux, représentés par nos Points d'Attraction, ont sur la modélisation du déplacement.

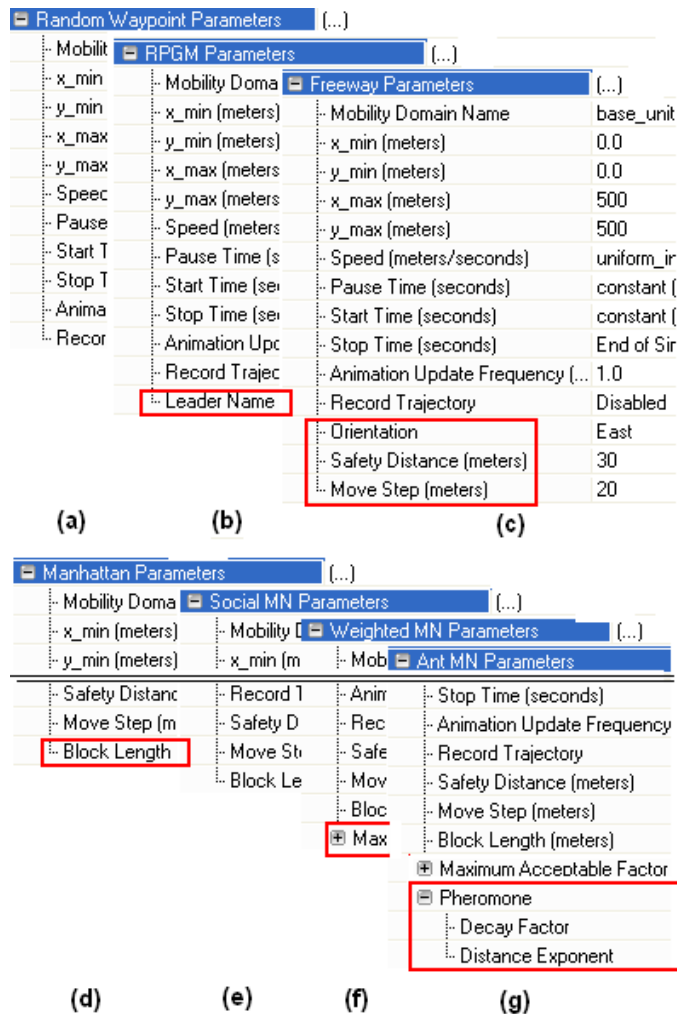


FIGURE 3.4 – Attributs Mobility Config Ext

Notre premier modèle *Social Manhattan Mobility Model (SMN)* établi, a-t-il un réel intérêt pratique ? Sera-t-il à même de remettre en cause tout ou partie des résultats obtenus par la démarche usuelle, majoritairement aléatoire (*MN* ou *RWP*) ? Notre intuition nous porte à penser que la réponse est également affirmative. Pour confirmer cette intuition, ou l'infirmier, nous procédons à une série de tests qui nous permettent de répondre de proche en proche à nos questionnements issus directement ou indirectement de notre motivation première.

Notre objectif dans cette section est d'évaluer l'influence de notre modèle *SMN* sur les performances des protocoles réseaux parmi lesquels les protocoles de routage constituent

FIGURE 3.5 – Nouveaux objets visuels *Opnet Modeler*

name	Attraction point 01	name	Station 05
model	manet_station_attraction_point	model	manet_station_social
x position	200	x position	200
y position	200	y position	320
Attractivity	5.0	Weight	5.0

FIGURE 3.6 – Attributs Sociaux

une branche très importante. Nous étudions ici les protocoles de routage les plus utilisés à savoir *Optimized Link State Routing Protocol (OLSR)* pour le cas proactif [Group 2003] et *Ad hoc On Demand Distance Vector (AODV)* pour le cas réactif [Perkins et al. 2003]. Rappelons ce que signifie les adjectifs réactif et proactif qui peuvent qualifier un protocole de routage réseau.

3.7.1 Protocoles de routage réactifs

Ces protocoles réagissent aux anomalies, par exemple, en cas de routes non fonctionnelles. Ils sont moins exigeants en temps processeur. L'exemple le plus courant est le protocole de routage *AODV, Ad hoc On Demand Distance Vector*. Ce protocole est conçu pour les réseaux mobiles Ad Hoc. Il permet un routage dynamique, automatique et « multi-sauts » entre les ordinateurs. Il est défini dans la *RFC 3561*. Le projet *AODV@IETF* est rendu possible par la collaboration des laboratoires *MOMENT* et *NMSL* de *Santa Barbara* et de *Intel R&D*.

3.7.2 Protocoles de routage proactifs

Ces protocoles vérifient l'état des liens et mettent à jour leurs tableaux en anticipant toute demande de route. C'est pourquoi il est proactif : il agit par anticipation. Il est complexe et exigeant en temps processeur. Il est en revanche performant. L'exemple le plus

courant est *OLSR*, *Optimized Link State Routing Protocol* qui fait l'objet de la *RFC 3626*. Il est conçu pour les réseaux mobiles Ad Hoc. Il est dit « dirigé par les tableaux » et utilise une technique appelée *MRP*, *multipoint relaying* pour acheminer les messages. C'est l'un des protocoles les plus stables et prometteurs.

3.7.3 Configurations de test

Nous déclinons nos tests selon les critères suivants :

- modèle de mobilité ;
- paramètres du modèle ;
- densité du réseau mobile ;
- nombre de Points d'Attraction.

Nous rapprochons notre modèle social (*SMN*) du modèle de mobilité le plus courant dans la communauté scientifique, à savoir Manhattan (*MN* ou *MAN*) pour mettre en perspective les apports de notre modèle.

Il va de soi que ce rapprochement nécessite un *cadre de test* commun. C'est la raison pour laquelle nous décidons de fixer certains paramètres de simulation. Il s'agit des paramètres suivants, nous utilisons ci-dessous leurs équivalents dans l'implémentation :

- *Safety Distance* : distance de sécurité routière ; valeur : 30 m, valeur moyenne en zone urbaine ;
- *Move Step* : unité de mouvement d'un nœud mobile ; valeur : 25 m, pour ne pas alourdir les temps de simulation ; diviseur du *Block Length* afin que les nœuds « passent » par les intersections ;
- *Block Length* : distance entre 2 lignes ou 2 colonnes de la grille ; valeur : 50 m, largeur moyenne d'un pâté de maisons ;
- $x_{min}, y_{min}, x_{max}, y_{max}$: aire de simulation ; valeur : 1 km^2 , superficie moyenne d'un quartier.

Le paramètre *Attractivity*, popularité d'un Point d'Attraction n'est relatif qu'à notre modèle social *SMN*. Pour nos tests sa valeur est fixée à 5. Elle est issue des tests préliminaires qui ont mis en évidence les contraintes suivantes. Si la valeur est trop grande, la distance devient négligeable. Inversement, si elle est trop petite, c'est elle-même qui devient

totalelement négligeable par rapport à la distance. D'autres séries de tests seront nécessaires afin de déterminer comment faire en sorte que les Points d'Attraction aient des attractivités initiales différentes les unes des autres, en tenant compte de ces contraintes.

Les paramètres de simulation variables sont ceux qui nous permettent la mise en perspective de nos modèles. Les valeurs choisies nous apparaissent suffisantes dans un premier temps pour connaître l'influence des paramètres correspondants sur les modèles étudiés. Il s'agit de :

- protocole de routage ; valeurs : OLSR et AODV, les plus utilisés dans les réseaux sans fil et les plus représentatifs des grands ensembles que sont les protocoles réactifs et les protocoles proactifs, respectivement ;
 - débit ; valeurs : 11, 24 et 54 Mbit/s, un débit faible, un autre moyen, un débit plus important ;
 - vitesse *maximale* des nœuds ; valeurs : 2, 3 et 15 m/s soit 7 km/h pour les piétons, 11 km/h pour les bicyclettes et 52 km/h pour les véhicules urbains ;
 - nombre de nœuds ; valeurs : 20 nœuds, un groupe moyen (classe d'école...) et 40 nœuds, un groupe plus important (groupe de touristes, conférence...);
 - nombre de Points d'Attraction ; valeurs : 4, quartier périphérique et 8, centre ville ;
- Les différentes valeurs de nos paramètres de simulation sont résumés en Table 3.2.

3.8 Résultats

Dans cette section, nous relatons les tests de simulation effectués selon les prérogatives de la section précédente. Les modèles *Manhattan (MN ou MAN)* et *Social Manhattan (SMN)* sont rapprochés afin de mettre en évidence leurs différences.

Les mesures reportées sont :

- *Load (Mbps)* : moyenne des *Mbit/s* envoyés par les couches hautes vers la couche *MAC* pour l'ensemble du réseau ;
- *Delay (s) ou Dlay* : moyenne en secondes des délais d'acheminement des paquets depuis la couche *MAC* du nœud émetteur jusqu'à la couche *MAC* du nœud récepteur ;
- *Throughput (Mbps) ou Thru* : moyenne des *Mbit/s* reçus par la couche *MAC* des

TABLE 3.2 – Simulation

Simulateur	OPNET Modeler 16.0
Couche MAC	IEEE 802.11
Taux de transmission	250 m
Type de trafic	UDP
Destination	aléatoire
Caractéristiques physiques	Extended Rate PHY (802.11g)
Protocoles de routage	OLSR, AODV
Débit (Mbit/s)	11, 24, 54
Vitesse des nœuds (m/s)	2 (piéton), 3 (bicyclette), 15 (véhicule)
Nombre de nœuds	20, 40
Aire de simulation	1000 m × 1000 m
Block length	50 m
Safety distance	30 m
Move step	25 m
Nombre de Points d'Attraction	4, 8
Attractivity	5
Modèle de mobilité	Manhattan (MN), Social Manhattan (SMN)

nœuds en provenance des couches basses ;

- *Data dropped (Retry Threshold Exceeded) (Mbps) ou DrpR* : moyenne des *Mbit/s* abandonnés par la couche *MAC* pour cause de brouillage à l'émission (médium radio non disponible).

Les résultats sont exposés selon 2 angles de vue.

En premier lieu, ils suivent la perspective *social* : piéton, bicyclette et véhicule urbain.

En second lieu, ils suivent la perspective *réseau* : Load, Throughput (Thru) et Delay (Dlay), Data dropped (Retry Threshold Exceeded, DrpR).

Pour ce qui est de la perspective *social*, nous proposons une figure par *Vitesse de nœud* : piéton, bicyclette et véhicule urbain. Chacune de ces figures comporte 2 colonnes et 3 lignes qui renferment de gauche à droite et de haut en bas, les résultats suivants :

- *AODV 40* ;
- *AODV 20* ;
- *OLSR 40* ;
- *OLSR 20* ;

3.8. RÉSULTATS

- *OLSR 20 8AP*;
- *OLSR 20 ASOC*.

Où :

- 20 et 40 représentent la densité du réseau ;
- 8AP représente le doublement des Points d'Attraction (*Attraction Points, AP*) de 4 à 8 et
- ASOC représente la « rébellion » d'une proportion de nœuds qui sont totalement asociaux en ce sens où ils ne respectent aucunement les modèles de mobilité de l'ensemble.

Chacun de ces résultats renferme les paramètres suivants :

- les 3 débits les plus usités : 11, 24 et 54 Mbit/s et
- les 4 mesures réseau classiques : Load, Throughput (Thru), Delay (Dlay) et Data Dropped (Retry Threshold exceeded) (DrpR).

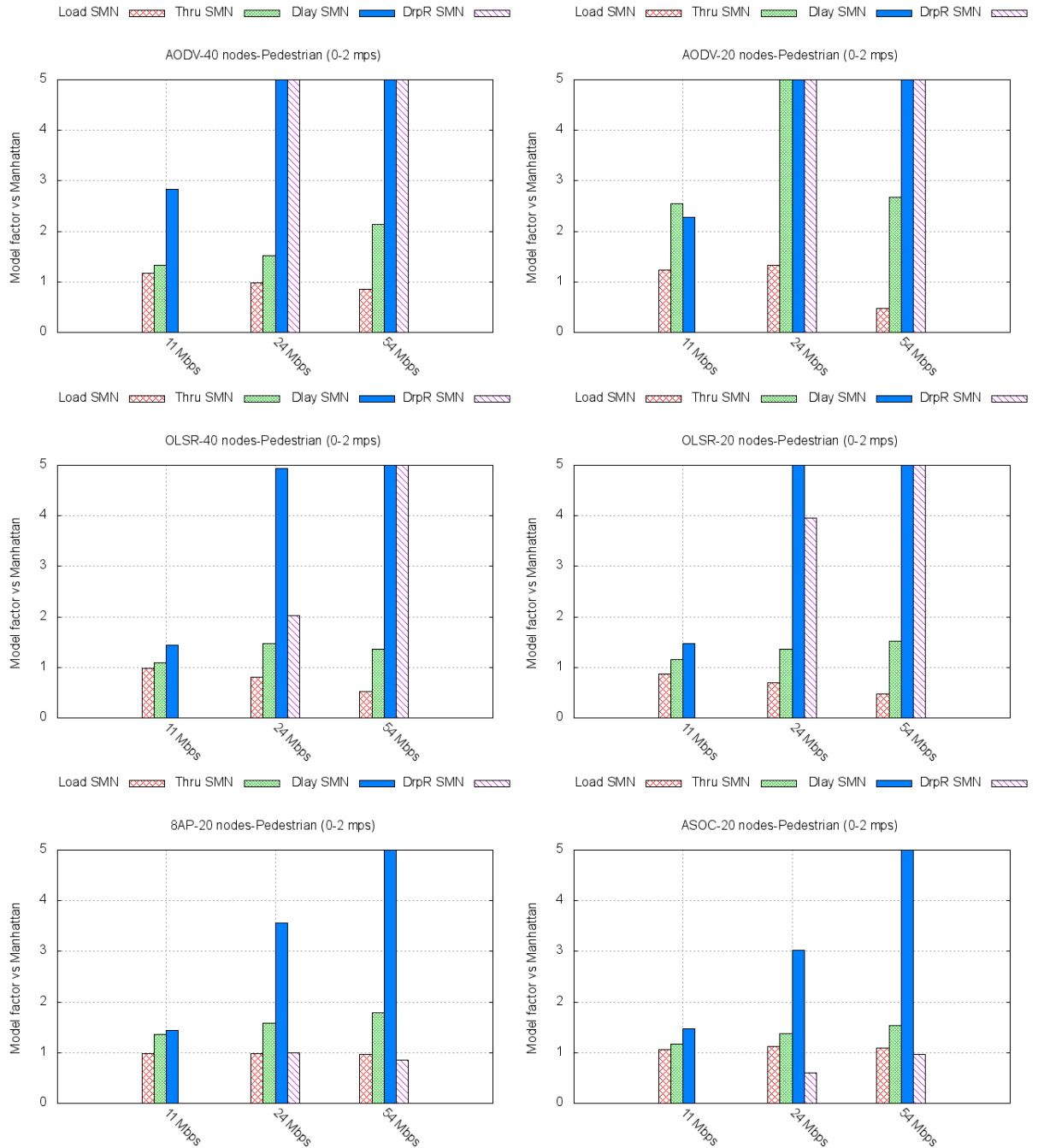


FIGURE 3.7 – SMN piéton

Résultats Piéton (*Pedestrian*) La Fig. 3.7 expose les résultats des simulations concernant la vitesse de déplacement des nœuds 0 à 2 m/s à savoir celle d'un piéton (*pedestrian*), en indiquant le facteur de variation de *Social Manhattan (SMN)* par rapport à *Manhattan (MAN)*.

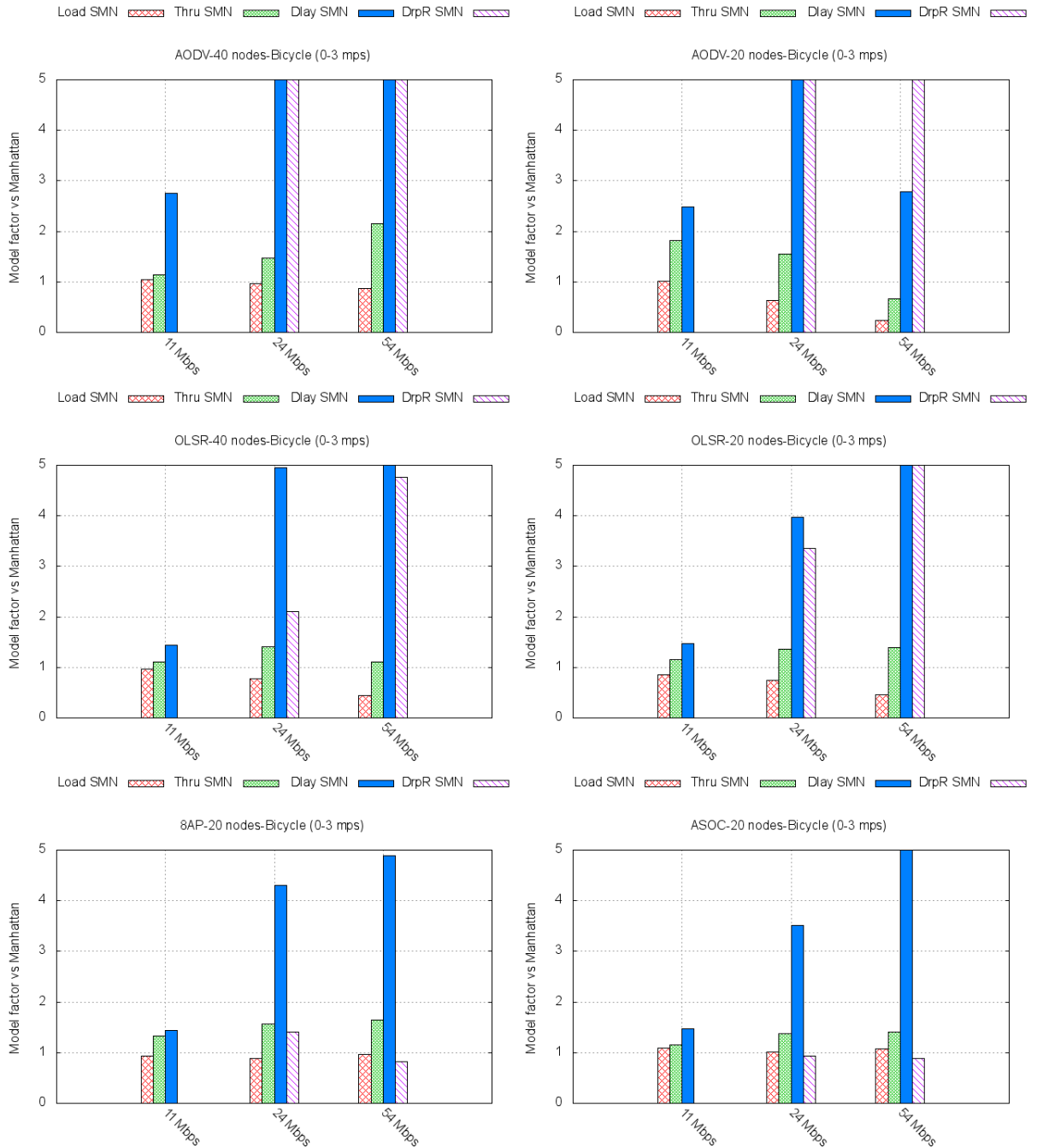


FIGURE 3.8 – SMN bicylette

Résultats Bicyclette (*Bicycle*) La Fig. 3.8 expose les résultats des simulations concernant la vitesse de déplacement des nœuds 0 à 3 *m/s* à savoir celle d'une bicyclette (*bicycle*), en indiquant le facteur de variation de *Social Manhattan (SMN)* par rapport à *Manhattan (MAN)*.

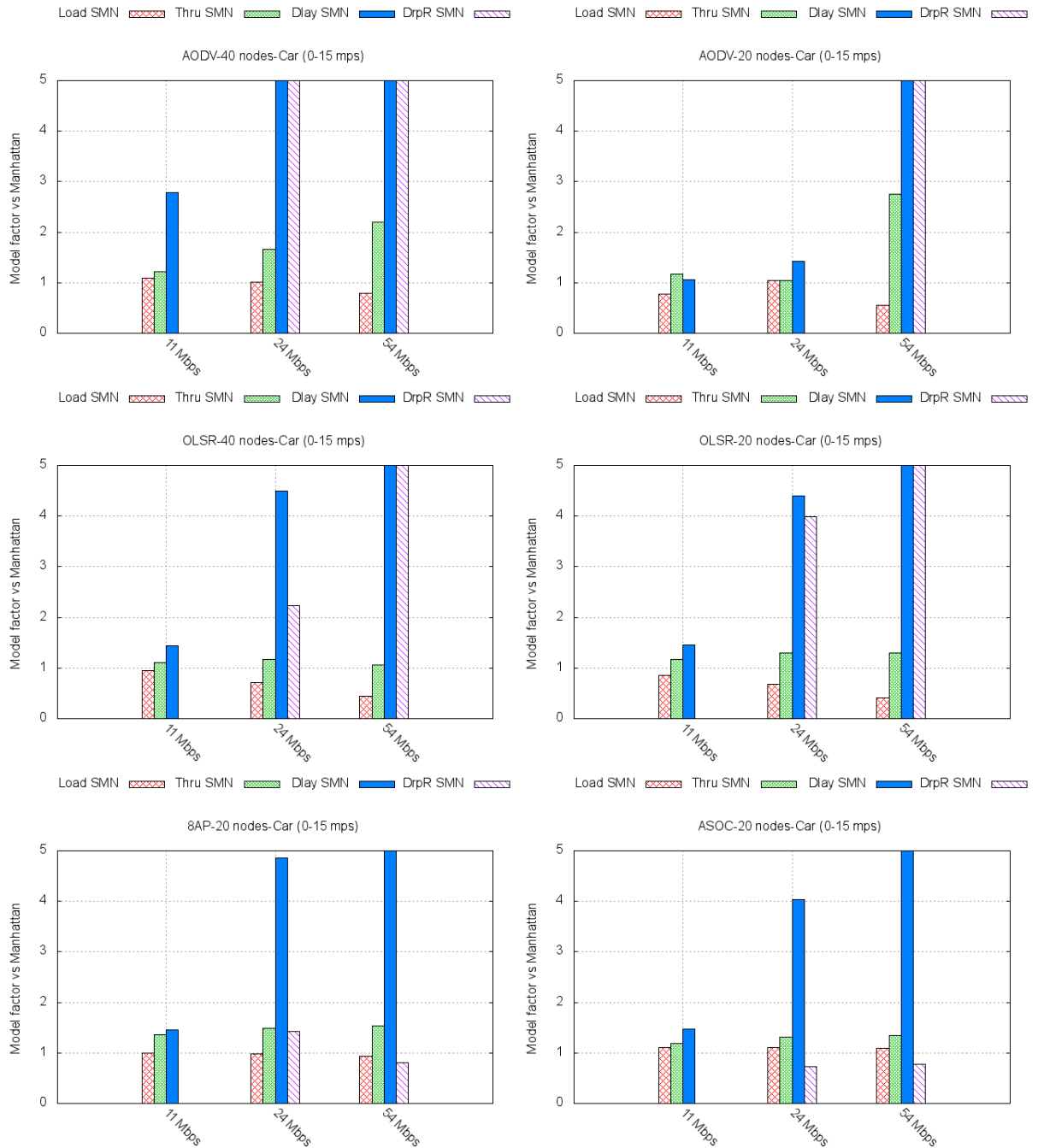


FIGURE 3.9 – SMN véhicule urbain

Résultats véhicule urbain (*Car*) La Fig. 3.9 expose les résultats des simulations concernant la vitesse de déplacement des nœuds 0 à 15 *m/s* à savoir celle d'une Automobile (*car*), en indiquant le facteur de variation de *Social Manhattan (SMN)* par rapport à *Manhattan (MAN)*.

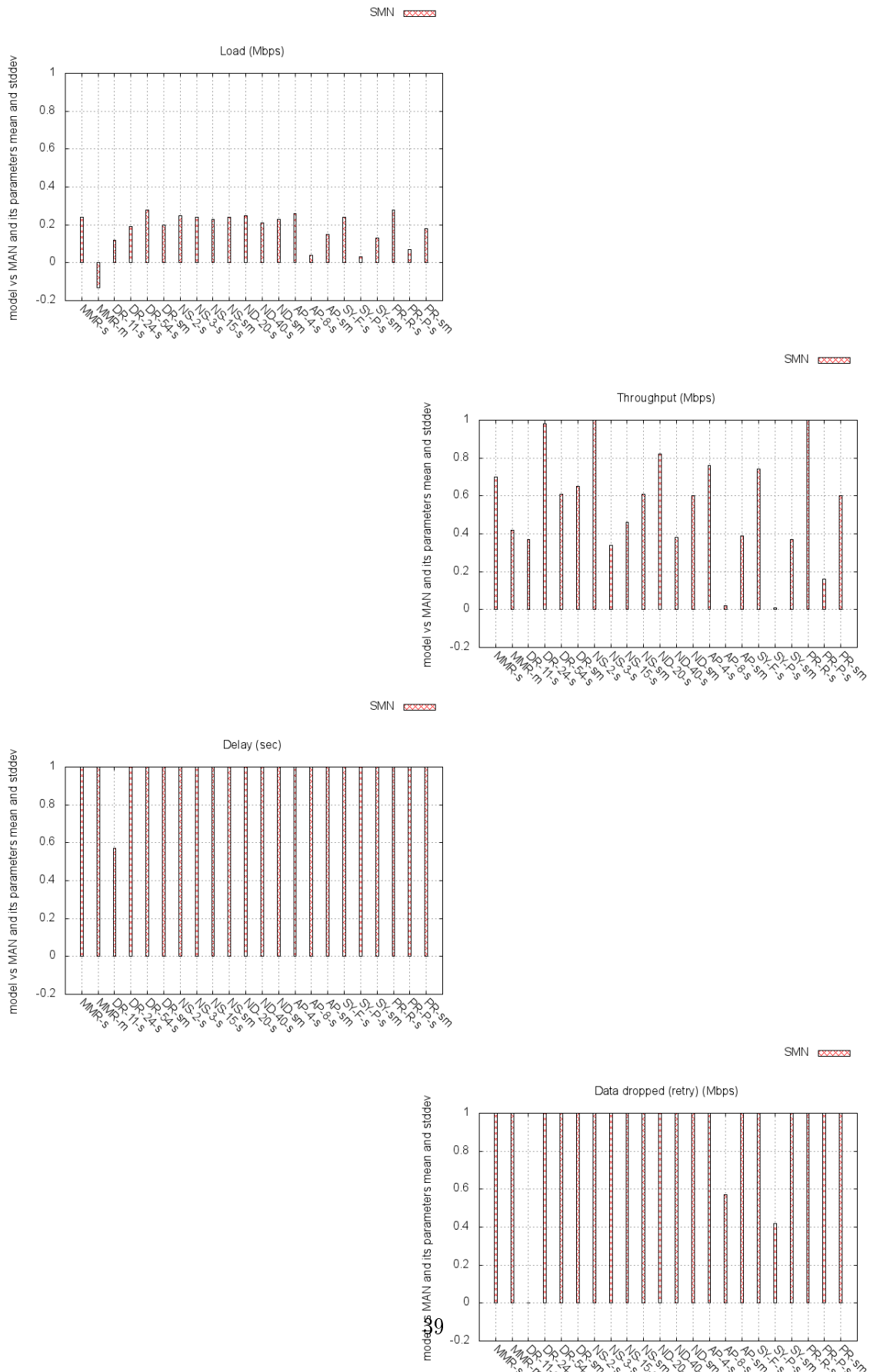


FIGURE 3.10 – SMN - Variables statistiques réseau

3.8.1 Perspective réseau

La perspective *réseau* présente une synthèse de l'influence de la prise en compte de l'aspect social sur les variables statistiques mesurées. La Fig. 3.10 rassemble les graphiques suivants : Load, Throughput (Thru), Delay (Dlay), Data dropped (Retry Threshold Exceeded, DrpR).

La Table suivante en donne la légende.

MMR	"Model me mean" to "MAN me mean" Ratio with Model in {SMN,WMN,AM1,AM2,AM3} and me in {Load, Thru, Dlay, DrpR}
MMR-s	stddev of MMR of ALL scenarios
MMR-m	mean of MMR of ALL scenarios
DR-n	Data Rate of n Mbps with n in {11,24,54}
NS-n	Node Speed in range 0-n mps with n in {2,3,15}
ND-n	Network Density of n nodes with n in {20,40}
AP-n	Number of Attraction Points with n in {4,8}
SY-t	Sociability Type with t in {(F)ull,(P)artial}
PR-t	Protocol Type with t in {(R)eactive,(P)roactive}
X	MMR of All scenarios covered by X with X in {DR,NS,ND,AP,SY,PR}
X-n	with n in {relative parameter value, see above}
X-n-s	stddev of X-n
X-sm	mean of all X-n-s

Un graphique comporte 22 histogrammes. Notons tout d'abord *MMR*, le rapport ou *ratio* du modèle considéré sur le modèle Manhattan de la variable statistique considérée. Un histogramme donne les moyenne et écarts des MMR, d'une part, de façon globale à tous les scénarios de test, puis, d'autre part, par paramètre, pour chacune de ses valeurs, pour les scénarios concernés. C'est cette dernière « moyenne d'écarts » qui dénote l' « influence » du paramètre sur l'ensemble de la mesure. Une précision concernant la Sociabilité s'impose : le caractère entièrement social (*(F)ull*) signifie que tous les nœuds sont respectueux du modèle de mobilité en vigueur ; tandis que la présence d'éléments en « rébellion » ou asociaux (*(P)artial*) est modélisée par environ 15 % des nœuds qui manquent totalement de respect à cet ordre préétabli.

Notons en premier lieu que :

La moyenne globale de Load lors du passage de Manhattan (MAN) à Social Manhattan (SMN) est de : -0,13. Cela signifie que Manhattan est surévalué en l'absence de prise en compte de l'aspect social.

La Table suivante résume l'« influence » des paramètres selon les prérogatives suivantes concernant l'écart global d'un paramètre (dénoté par X-s, conférer la Table légende ci-dessus et l'axe des abscisses des graphiques) :

- inférieur à 0,2 : influence nulle ;
- entre 0,2 et 0,6 : influence faible ;
- entre 0,6 et 1 : influence moyenne ;
- supérieur à 1 influence forte.

	Load	Throughput	Delay	Data dropped
Debit	nulle	moyenne	forte	forte
Vitesse	faible	moyenne	forte	forte
Densité	faible	moyenne	forte	forte
Nombre de Points d'Attraction	nulle	faible	forte	forte
Sociabilité	nulle	faible	forte	forte
Réactif / Proactif	nulle	moyenne	forte	forte

À noter que le cas extrême où il n'y a pas de Point d'Attraction, à savoir Manhattan (MN ou MAN) peut être assimilé au cas où il y a une infinité de Points d'Attraction. En effet, dans ce cas, les nœuds sont distribués aléatoirement sans prise en compte de leurs éventuels liens sociaux.

Quand le nombre de Points d'Attraction diminue, la concentration des nœuds augmente. Cela a pour effet de réduire le délai de transmission de bout en bout. En effet, il n'y a pratiquement plus besoin de nœuds intermédiaires. En revanche, nombre de données sont perdues pour cause de bourrage du médium air (Data dropped). L'explication est la suivante : la bande de fréquence étant fixe, l'augmentation de la densité géographique des nœuds résulte en une augmentation du nombre de collisions.

3.9 Conclusion

Nous avons élaboré dans ce chapitre, un premier modèle de mobilité à Points d'Attraction à attractivité fixe.

Nous avons mis en évidence que le manque de prise en compte de l'aspect social résulte en une surestimation des performances.

Les variables statistiques réseau *Delay* et *Data dropped* qui représentent le délai entre couches MAC des émetteur et récepteur, d'une part, et, d'autre part, les données abandonnées au niveau de la couche MAC à l'émission pour cause de bourrage du médium-air, ont rendu, pour la totalité des paramètres (débit, vitesse, densité, nombre de Points d'Attraction, Sociabilité et Type de protocole réactif ou proactif), des écarts considérables. Pour les autres variables statistiques, *Load* et *Throughput*, la tendance de leurs influences est « faible » voire « moyenne », plus rarement « nulle ». Cela nous permet de conclure que la prise en compte de l'aspect social des communications dans les réseaux sans fil devient une nécessité tant pour les protocoles réactifs que les protocoles proactifs, d'une part, et, d'autre part, tant du point de vue débit (cruciaux pour les applications multimédia en plein essor) que des point de vue vitesse de déplacement (la mobilité étant LE challenge d'aujourd'hui) et la densité laquelle fait l'objet de nombreuses études qui s'appuient souvent sur des traces réelles afin d'anticiper les goulots d'étranglement.

Chapitre 4

Les réseaux pondérés

4.1 Introduction

Dans le chapitre précédent, nous avons proposé une classe de modèles de mobilité sociale, *Social Mobility Models*, dont les valeurs sociales des Points d'Attraction, à savoir l'attraction d'un lieu, sont choisies initialement par rapport à la réputation de celui-ci et fixées une fois pour toute. Or, une valeur d'attractivité fixe ne peut pas modéliser l'évolution de la popularité d'un lieu. Dans ce chapitre, nous proposons une nouvelle classe de modèles de mobilité sociale pondérée, *Weighted Social Mobility Models*, dans lesquels nous introduisons d'autres aspects de la socialisation liés aux déplacements des membres d'une communauté et leurs centres d'intérêts communs.

Cette nouvelle classe de modèles de mobilité sociale pondérée sera basée sur les notions de Points d'Attraction sociaux et de réseaux pondérés afin de tenir compte de plus de caractéristiques sociales. Les valeurs sociales affectées aux Points d'Attraction changent au fil du temps. Elles sont étroitement liées à la fois au nombre de visites et la distance qui les séparent des nœuds mobiles de façon à modéliser un *effet boule de neige* ou *de bouche à oreille*. Dans ce qui suit nous introduisons donc la notion de « poids social », *weight*, d'un nœud du réseau et faisons varier l'attractivité d'un Points d'Attraction (augmenter l'attractivité) en fonction du poids du nœud mobile à chaque fois qu'un nœud mobile visite un lieu. En effet, un lieu visité par une personnalité connue (avec un plus grand poids social) devient subitement plus attractif.

Pour expliciter notre démarche, la suite de ce chapitre est organisée comme suit, dans la

section 4.2 nous présentons les réseaux pondérés et leurs applications dans le domaine des réseaux mobiles, quelques exemples sont cités. En section 4.3 nous présentons nos Points d'Attraction sociaux sur lesquels repose notre nouvelle classe de modèles de mobilité pondérée *Weighted Social Mobility Model*, le modèle mathématique associé et quelques exemples de fonctions sont respectivement présentés en sections 4.4 et 4.5. Comme exemple d'application pratique, nous décrivons notre modèle *Weighted Social Manhattan Mobility Model (WMN)* dans la section 4.6, Son implémentation sous Opnet Modeler et les configurations de test sont décrites en section 4.7. Les résultats associés sont présentés et discutés en section 4.8, Enfin nous concluons le chapitre en section 4.9.

4.2 Réseaux pondérés

4.2.1 Théorie des graphes

De nombreux systèmes peuvent être représentés comme des réseaux ou des graphes, i.e. un ensemble de sommets reliés par des arcs. Parmi ces exemples on peut citer *Internet*, le *World Wide Web*, les réseaux de référencement, les réseaux sociaux et les réseaux biologiques ou biochimiques de toutes sortes. Bien qu'une ancienne et solide filière d'études soit établie pour ce genre de réseaux en mathématiques et en sociologie, la recherche sur les réseaux n'attire que ces dernières années l'attention des membres de la communauté des réseaux mobiles. Dans ces réseaux les liens entre les différentes entités peuvent être représentés par $(0, 1)$ ou des matrices binaires. Un réseau à n sommets est représenté par une matrice d'adjacence A de taille $n \times n$ comme suit :

$$A_{i,j} = \begin{cases} 1 & \text{si } i \text{ et } j \text{ sont connectés} \\ 0 & \text{sinon} \end{cases}$$

Néanmoins, les connexions dans de nombreux réseaux ne se limitent pas à des entités binaires, présence de connexion ou non, mais peuvent avoir des poids associés qui enregistrent leurs forces par rapport aux autres [Newman 2004]. Les études des réseaux mobiles ont, en général, tenus à l'écart ce genre de réseaux pondérés, qui sont souvent perçus comme étant plus difficiles à analyser que les réseaux non pondérés.

Un réseau pondéré peut être représenté mathématiquement par une matrice d'adjacence avec des éléments qui ne sont pas tout simplement zéro ou un (existence ou non d'arcs), mais représentant les poids sur les arcs :

A_{ij} = poids de la connexion à partir de i à j .

Un exemple de graphe pondéré est donné dans la figure. 4.1.

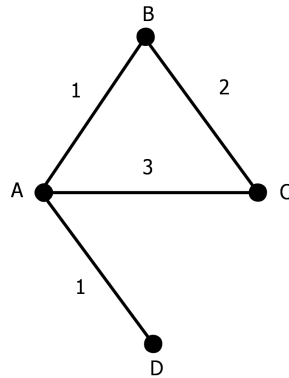


FIGURE 4.1 – Exemple de graphe pondéré

Dans ce qui suit, nous utiliserons la notion de réseaux pondérés pour apporter des améliorations à nos modèles de mobilité et modéliser nos modèles de mobilité pondérée.

4.2.2 Exemples

La notion de réseaux pondérés a été utilisée dans les réseaux mobiles. Par exemple, Musolesi and Mascolo [2007] propose un modèle de mobilité fondé sur la théorie des réseaux sociaux. Le modèle permet un regroupement de machines en conformité avec les relations sociales existant entre les individus. Ce regroupement est alors mis en correspondance avec un espace topographique dont les mouvements suivent les variations de l'intensité des liens sociaux et montrent que les modèles de mobilité synthétiques sont une bonne approximation des modes de circulation de l'homme. Alzoubi and Ayyash [2006] définit un arbre de routage de la qualité de service (*Quality of Service Routing Tree, QoSRT*), et ce en passant d'un graphe pondéré à un sous-graphe ayant la structure d'un arbre et où l'ensemble des arêtes intermédiaires du chemin entre n'importe quelle paire de nœuds est un sous-ensemble de l'ensemble des arêtes initial. Le poids d'une arête représente sa

bande passante. En outre, le chemin choisi sera celui dont le poids de l'arête de poids le plus faible (goulot d'étranglement) est le plus grand et proposent un algorithme distribué de construction d'un QoSRT qui s'adapte rapidement à des environnements mobiles.

4.3 Points d'Attraction pondérés

La classe de modèles de mobilité sociale pondérée *Weighted Social Mobility Model* va introduire de nouvelles contraintes pour rendre compte du comportement humain réel. Ces modèles prennent en compte le fait que les gens vont visiter, non seulement, le lieu le plus proche en premier, de plus, plus un endroit sera visité, plus il deviendra intéressant (effet bouche-à-oreille).

En rapport avec notre première classe de modèles de mobilité sociale *Social Mobility Models*, nous reprenons l'idée des Points d'Attraction, qui rappellent les ont des valeurs d'attractivités fixes jusque là.

Nous introduisons les caractéristiques suivantes dans notre nouvelle classe de modèles de mobilité sociale pondérée *Weighted Social Mobility Models* :

- des *Points d'Attraction (PA)*, avec des attractivités changeantes $r(t)$;
- des *nœuds mobiles (NM)*, avec des poids w .

Un réseau pondéré va correspondre à la figure 4.2. Un nœud mobile va se diriger vers le Point d'Attraction envers lequel il est le plus attiré (i.e avec la plus grande valeur $r_i(t)$). En arrivant il va faire augmenter la popularité de ce dernier (i.e. $r(t)$) d'une valeur w (poids social du visiteur).

Afin de limiter l'impact d'un Point d'Attraction sur les autres pour que tous puissent survivre, nous introduisons une borne supérieure à l'attractivité. Pour représenter certains comportements sociaux comme l'enthousiasme et la désaffection du public vis-à-vis d'un lieu, le coefficient d'attraction peut augmenter en fonction des valeurs d'attraction jusqu'à un seuil donné pour ensuite diminuer.

Le modèle mathématique correspondant est décrit dans la section suivante.

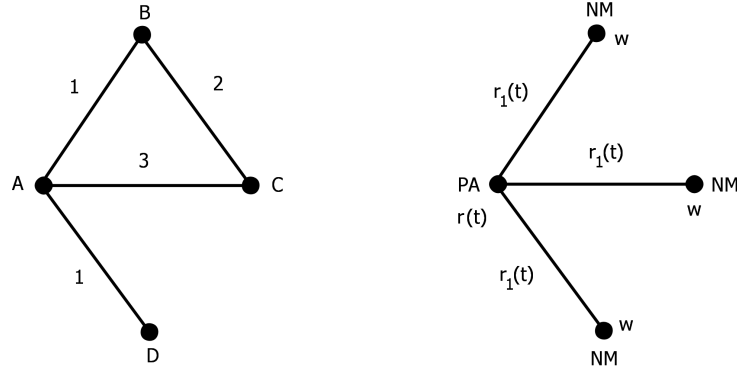


FIGURE 4.2 – Un exemple de Point d'Attraction

4.4 Modélisation

Nous reprenons notre modèle mathématique décrit dans le chapitre précédent en section 3.4, p. 21 et y introduisons les changements nécessaires pour l'adapter à nos nouveaux modèles de mobilité sociale pondérée.

Pour notre nouveau modèle mathématique, un poids aléatoire $w(n)$ est affecté à chaque nœud mobile n . Chaque fois qu'un nœud arrive à un Point d'Attraction, le poids de ce point est augmenté. Les nœuds choisissent leur nouvelle destination comme le Point d'Attraction ayant le coefficient social le plus fort.

Soit l'application $w : N \rightarrow [0; 1]$ avec $\sum_{n \in N} w(n) = 1$, le poids social (i.e. la classe sociale) des nœuds. En introduisant w , l'attractivité de $a \in A$ à l'instant $t \in T$ devient l'application $r : A \times T \rightarrow [0; 1]$, telle que :

$$r(a, t) = \begin{cases} \frac{u(a)}{\sum_{a' \in A} u(a')} & t = 0 \\ \frac{\sum_{n' \in N} w(n') \min(r_1(a, n', t - \Delta), r_{max}(t))}{\sum_{a' \in A} \sum_{n' \in N} w(n') \min(r_1(a', n', t - \Delta), r_{max}(t))} & t \neq 0 \end{cases}$$

avec $u(a)$ tel que défini à l'Eq. 3.1, p. 22 du chapitre 3 où

$$r_1(a, n, t) = \frac{\frac{r(a, t)}{d(a, n, t)}}{\sum_{a' \in A} \frac{r(a', t)}{d(a', n, t)}} \quad (4.1)$$

Nous introduisons une borne supérieure à l'impact d'un Point d'Attraction sur les autres, afin que tous puissent survivre. Nous limitons l'impact des Points d'Attraction par une fonction r_{max} fonction de $f(t)$ et de la valeur sociale moyenne normalisée, à savoir : $\frac{1}{|A|}$ où $|A|$ est le nombre des Points d'Attraction dans le réseau.

ainsi

$$r_{max}(t) = f(t) \times \frac{1}{|A|} \quad (4.2)$$

qui représente une limite consécutive au comportement social modélisé, $f(t)$ étant le *maximum acceptable factor* de Harfouche et al. [2010] qui décrit la manière dont la popularité d'un lieu (PA) croit ou décroît dans le temps. Pour expliquer son fonctionnement, nous donnons quelques exemples de la fonction $f(t)$ dans la section suivante.

La fonction *min* joue le rôle d'un filtre passe-bas qui garantit la survie des Points d'Attraction (i.e. qu'ils soient visités) en évitant que les nœuds ne soient attirés que par un sous-ensemble des Points d'Attraction ayant des valeurs de l'attractivité trop grande.

Lors de ses déplacements, un nœud mobile, ne vas pas forcément visiter tous les Points d'Attraction présents dans son réseau, et ce selon ses affinités mais aussi selon le moment de la journée où on se trouve (aller au bureau le matin, aller manger à midi, activités en fin de journée, rentrer chez soi le soir). Nous pouvons choisir les Points d'Attraction à visiter (ou qui restent à visiter) par un nœud mobile à un instant donné en définissant la fonction v suivante.

Soit $v : A \times N \times T \rightarrow \{V, F\}$ où V et F signifient VRAI et FAUX, l'application qui définit l'ensemble des Points d'Attraction à visiter à l'instant t par le nœud n :

$$v(a, n, t) = \begin{cases} T & \text{si non défini par l'utilisateur} \\ F & \text{si visité} \end{cases}$$

Finalement le nœud mobile va se diriger vers le Point d'Attraction envers lequel il a la plus grande attraction : n attiré par $a \iff$

$$a = \arg \max_{a' | v(a', n, t) = T} \frac{r(a', t)}{d(a', n, t)} \quad (4.3)$$

4.5 Maximum Acceptable Factor ($f(t)$) : exemples

La fonction $f(t)$ définit la manière dont la valeur de l'attractivité d'un Point d'Attraction change dans le temps (la manière dont les gens se ruent, ou pas, vers un lieu).

Nous donnons ci-après quelques fonctions pouvant jouer le rôle de $f(t)$, le *maximum acceptable factor* :

- une constante k ;
- une fonction qui croît puis décroît de façon parabolique ;
- une fonction qui croît puis est constante.

Ces fonctions sont données à titre d'exemple, mais ne sont pas les seules valeurs possibles pour $f(t)$. $f(t)$ peut prendre n'importe quelle valeur, fonction ou distribution selon le phénomène qu'on veut modéliser.

4.5.1 Fonction constante

Une valeur constante du *Maximum Acceptable Factor* ($f(t)$) peut représenter un intérêt constant vers un lieu donné, comme un monument important (tour Eiffel, arc de triomphe,...) dont la popularité est stable depuis des années.

L'équation 4.2, p. 48 de la section 4.4 devient :

$$r_{max}(t) = k \times \frac{1}{|A|}$$

4.5.2 Fonction parabole

Une autre valeur intéressante du *Maximum Acceptable Factor* ($f(t)$) est la fonction parabole, qui peut représenter un évènement vers lesquels les gens sont très enthousiastes au début, puis se désaffectent dans le temps : exemple un nouveau film au cinéma.

4.5. MAXIMUM ACCEPTABLE FACTOR ($F(T)$) : EXEMPLES

Si nous modélisons maintenant un événement dont la durée est D et dont l'attractivité croît jusqu'à un maximum noté M et qui décroît par la suite selon les conditions suivantes :

1. au début et à la fin de l'événement, $r_{max}(0) = r_{max}(D) = \frac{1}{|A|} \iff f(0) = f(D) = 1$;
2. M est atteint à l'instant c de l'événement, i.e. $0 \leq c \leq D, f(c) = M$.

Par exemple, supposons que $c = \frac{D}{2}$. Alors le *maximum acceptable factor* peut être modélisé par une équation paramétrique du second degré de la forme : $f(t) = m_1 t^2 + m_2 t + m_3$.

Laquelle se résout en écrivant m_1, m_2 and m_3 en termes de M et E , le calcul est le suivant :

$$\begin{cases}
 m_1(0)^2 + m_2(0) + m_3 = 1 & (1) : f(0) = 1 \\
 m_1(D)^2 + m_2(D) + m_3 = 1 & (2) : f(D) = 1 \\
 m_1(\frac{D}{2})^2 + m_2(\frac{D}{2}) + m_3 = M & (3) : f(\frac{D}{2}) = M
 \end{cases}$$

$$\begin{cases}
 m_1 D^2 + m_2 D + m_3 = 1 & (1) \\
 m_1(\frac{D}{2})^2 + m_2 \frac{D}{2} + m_3 = 0 & (2) \\
 m_1(\frac{D}{2})^2 + m_2 \frac{D}{2} + m_3 = M - 1 & (3)
 \end{cases}$$

$$\begin{cases}
 m_1 D^2 + m_2 D = 0 & (1) : (2) \\
 \frac{m_1}{2} D^2 + m_2 D = 2(M - 1) & (2) : (3)
 \end{cases}$$

$$\begin{cases}
 m_1 D^2 + m_2 D = 0 & (1) \\
 -m_2 D = -4(M - 1) & (2) : (1) - 2 \times (2)
 \end{cases}$$

$$\begin{cases}
 m_1 D^2 + m_2 D = 0 & (1) \\
 m_2 = 4 \frac{M-1}{D} & (2) : (2) \times -\frac{1}{D}
 \end{cases}$$

$$\begin{aligned}
 m_1 D^2 + 4 \frac{M-1}{D} D &= 0 & (1) : (1) \text{ using } (2) \\
 m_1 &= \frac{4}{D^2} (1 - M)
 \end{aligned}$$

$$\begin{cases}
 m_1 = \frac{4}{D^2} (1 - M) \\
 m_2 = -\frac{4}{D} (1 - M) \\
 m_3 = 1
 \end{cases}$$

$$\mathbf{f(t)} = \frac{4}{D^2} (1 - M) t^2 - \frac{4}{D} (1 - M) t + 1$$

Fig. 4.3 donne représentation graphique du *maximum acceptable factor* $f(t)$ pour $D = 50$ et $M = 7$:

$$\mathbf{f(t)} = -0.0096t^2 + 0.48t + 1$$

courbe issue de Anonymous [2007].

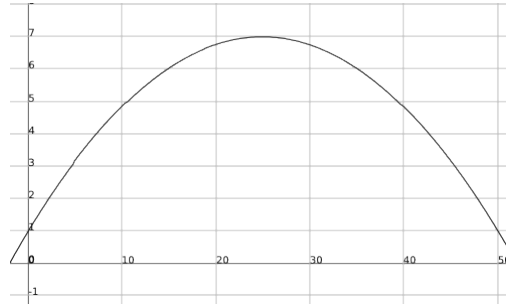


FIGURE 4.3 – Maximum Acceptable Factor $f(t)$ pour $D = 50$ et $M = 7$

4.5.3 Fonction hybride mi-parabolique et mi-constante

Supposons que notre fonction croisse de façon parabolique comme à la section 4.5.2 jusqu'à un maximum, puis qu'elle soit constante comme à la section 4.5.1. Ceci représente un grand engouement vers un lieu donné au départ, puis une popularité stable (exemple : un nouveau monument vers lesquels les gens sont très enthousiastes au début puis la fréquentation se voit stabilisée). Nous obtenons la nouvelle équation suivante :

$$f(t) = \begin{cases} \frac{4}{D^2}(1 - M)t^2 - \frac{4}{D}(1 - M)t + 1 & t \leq c = \frac{D}{2} \\ k = M & t > c \end{cases}$$

4.6 Application : Weighted Manhattan (WMN)

Comme pour notre première classe de modèles de mobilité sociale, nous choisissons le modèle *Manhattan* comme exemple d'application pour notre classe de modèles de mobilité sociale pondérée. Le modèle *Weighted Social Manhattan (WMN)* [Harfouche et al. 2010] introduit le modèle mathématique précédent dans *Manhattan*.

La Table 4.1 décrit *WMN* selon notre classification des modèles de mobilité.

Le schéma de principe de *WMN* est donné en figure Fig. 4.4.

TABLE 4.1 – WMN

Région	Topologie en Grille
Nœuds	$N \geq 2$
Communauté	<i>social</i> : → avoir les mêmes centres d'intérêts
Comportement	<i>social</i> : → attirés par les mêmes Points d'Attraction (PA)
Déplacements	<i>social</i> : suivant Harfouche et al. [2010] → vitesse aléatoire des nœuds ; → se déplacer suivant une ligne/colonne ; → à une intersection, choisir de se déplacer vers le plus proche, plus intéressant Point d'Attraction ; → l' <i>attractivité</i> des PA n'est pas constante : plus un PA est visité, plus il devient attractif.

4.7 Implémentation et paramètres du modèle

4.7.1 Implémentation

Nous développons le modèle *Weighted Social Manhattan Mobility Model (WMN)* sur la base de notre modèle *Social Manhattan Mobility Model (SMN)*. Le nouveau modèle *WMN* ajoute :

- Le *weight*, i.e. poids social de l'utilisateur, au nœud mobile, ce poids est implémenté par le composant *Social Manet Station* qui est une extension du composant *Opnet Manet Station* ;
- Le *Maximum Acceptable Factor*, qui limite l'accroissement de l'attractivité d'un Point d'Attraction.

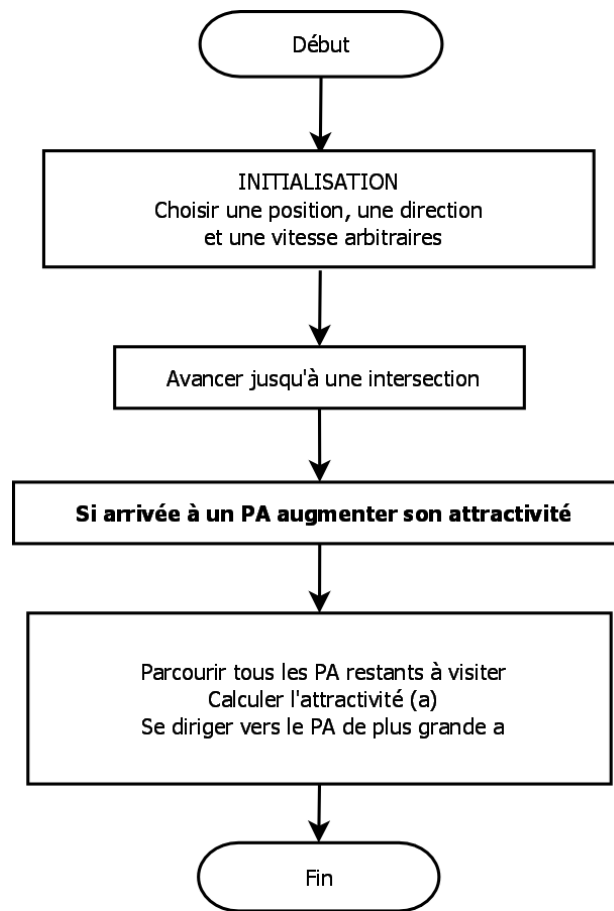
4.7.2 Paramètres du modèle

Notre objectif dans ce qui suit est d'évaluer l'influence de notre modèle *WMN* sur les performances des protocoles de routage et le comparer avec *SMN* et *MN* de base.

Les déclinaisons de nos tests sont les même que ceux décrit en section 3.7, p. 26 :

Dans ce *cadre de test* commun, nous avons les paramètres suivants qui sont spécifiques à notre modèle *WMN* :

- *weight* : poids social ; influence du nœud mobile sur l'attractivité ;



Weighted Manhattan

FIGURE 4.4 – Weighted Manhattan (WMN) : schéma de principe

- *Attractivity* : popularité d'un Point d'Attraction ; devient changeante en fonction de la fréquentation ;
- *Maximum Acceptable Factor* : fonction $f(t)$; limite de la croissance de l'attractivité ; valeur *constante* ;

4.8 Résultats

4.8.1 Introduction

Dans cette section, nous relatons les tests de simulation effectués selon les prérogatives de la section précédente. Les modèles *Manhattan* (*MN/MAN*) et *Weighted Manhattan* (*WMN*) sont rapprochés afin de mettre en évidence leurs différences. Le(s) rapprochement(s) présenté(s) au(x) chapitre(s) précédent(s) est(ont) également conservé(s) afin de le(s) mettre en perspective avec le présent rapprochement.

4.8.2 Configurations de test

Nous déclinons nos tests selon les critères suivants :

- modèle de mobilité ;
- paramètres du modèle ;
- densité du réseau mobile ;
- nombre de Points d’Attraction.

Nous rapprochons notre modèle *Weighted Manhattan* (*WMN*) du modèle de mobilité de base, à savoir *Manhattan* (*MN* ou *MAN*) pour mettre en perspective les apports de notre modèle.

Il va de soi que ce rapprochement nécessite un *cadre de test* commun. C’est la raison pour laquelle nous décidons de fixer certains paramètres de simulation. (cf. Table 4.2 pour les paramètres de simulation). Il s’agit des paramètres suivants, nous utilisons ci-dessous leurs équivalents dans l’implémentation :

- *Safety Distance* : distance de sécurité routière ; valeur : 30 m, valeur moyenne en zone urbaine ;
- *Move Step* : unité de mouvement d’un nœud mobile ; valeur : 25 m, pour ne pas alourdir les temps de simulation ; diviseur du *Block Length* afin que les nœuds « passent » par les intersections ;
- *Block Length* : distance entre 2 lignes ou 2 colonnes de la grille ; valeur : 50 m, largeur moyenne d’un pâté de maisons ;

- $x_min, y_min, x_max, y_max$: aire de simulation ; valeur : $1 km^2$, superficie moyenne d'un quartier.

Pour les paramètres de simulation variables, les valeurs choisies nous apparaissent suffisantes dans un premier temps pour connaître l'influence des paramètres correspondants sur les modèles étudiés. Il s'agit de :

- protocole de routage ; valeurs : OLSR et AODV, les plus utilisés dans les réseaux sans fil et les plus représentatifs des grands ensembles que sont les protocoles réactifs et les protocoles proactifs, respectivement ;
 - débit ; valeurs : 11, 24 et 54 Mbit/s, un débit faible, un autre moyen, un débit plus important ;
 - vitesse *maximale* des nœuds ; valeurs : 2, 3 et 15 m/s soit 7 km/h pour les piétons, 11 km/h pour les bicyclettes et 52 km/h pour les véhicules urbains ;
 - nombre de nœuds ; valeurs : 20 nœuds, un groupe moyen (classe d'école...) et 40 nœuds, un groupe plus important (groupe de touristes, conférence...);
 - nombre de Points d'Attraction ; valeurs : 4, quartier périphérique et 8, centre ville ;
- Les différentes valeurs de nos paramètres de simulation sont résumés en Table 4.2.

Les résultats sont exposés selon 2 angles de vue.

En premier lieu, ils suivent la perspective *social* : piéton, bicyclette et véhicule urbain.

En second lieu, ils suivent la perspective *réseau* : Load, Throughput (Thru), Delay (Dlay) et Data dropped (Retry Threshold Exceeded, DrpR).

Pour ce qui est de la perspective *social*, nous proposons une figure par *Vitesse de nœud* : piéton, bicyclette et véhicule urbain. Chacune de ces figures comporte 2 colonnes et 3 lignes qui renferment de gauche à droite et de haut en bas, les résultats suivants :

- *AODV 40* ;
- *AODV 20* ;
- *OLSR 40* ;
- *OLSR 20* ;
- *OLSR 20 8AP* ;
- *OLSR 20 ASOC*.

Où :

TABLE 4.2 – Simulation

Simulateur	OPNET Modeler 16.0
Couche MAC	IEEE 802.11
Taux de transmission	250 m
Type de trafic	UDP
Destination	aléatoire
Caractéristiques physiques	Extended Rate PHY (802.11g)
Protocoles de routage	OLSR, AODV
Débit (Mbit/s)	11, 24, 54
Vitesse des nœuds (m/s)	2 (7 km/h, piéton, <i>pedestrian</i> , <i>ped</i>) 3 (11 km/h, bicyclette, <i>bicycle</i> , <i>bic</i>) 15 (52 km/h, véhicule, <i>car</i>)
Nombre de nœuds	20, 40
Aire de simulation	1000 m × 1000 m
Block length	50 m
Safety distance	30 m
Move step	25 m
Nombre de Points d'Attraction	4, 8
Attractivity	5
Modèle de mobilité	Manhattan (MN) Social Manhattan (SMN) Weighted Social Manhattan (WMN)

- 20 et 40 représentent la densité du réseau ;
- 8AP représente le doublement des Points d'Attraction (*Attraction Points, AP*) de 4 à 8 et
- ASOC représente la « rébellion » d'une proportion de nœuds qui sont totalement asociaux en ce sens où ils ne respectent aucunement les modèles de mobilité de l'ensemble.

Chacun de ces résultats renferme les paramètres suivants :

- les 3 débits les plus usités : 11, 24 et 54 Mbit/s et
- les 4 mesures réseau classiques : Load, Throughput (Thru), Delay (Dlay) et Data Dropped (Retry Threshold exceeded) (DrpR).

4.8. RÉSULTATS

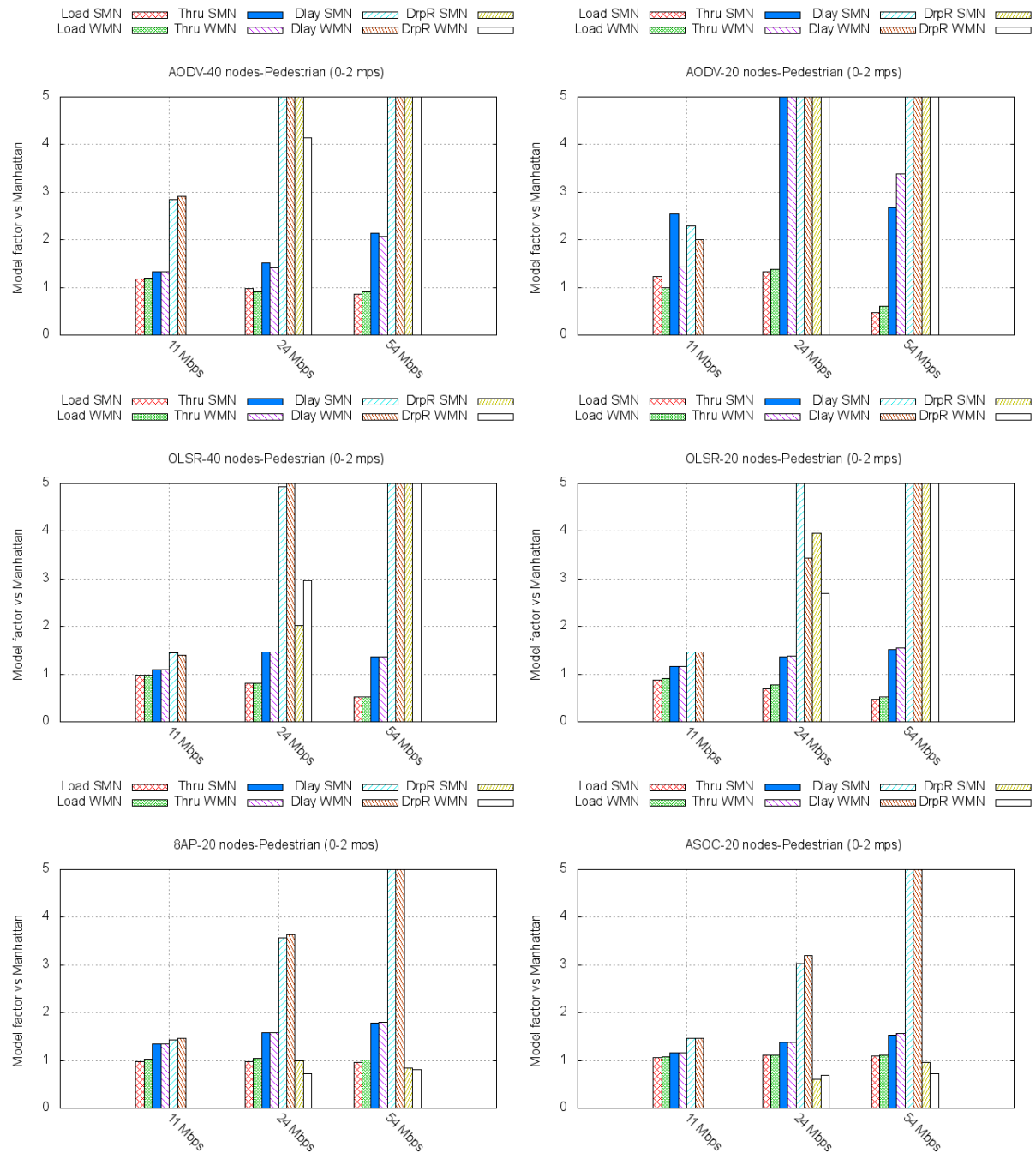


FIGURE 4.5 – WMN piéton

Résultats Piéton (*Pedestrian*) La Fig. 4.5 expose les résultats des simulations concernant la vitesse de déplacement des nœuds 0 à 2 *m/s* à savoir celle d'un piéton (*pedestrian*), en indiquant le facteur de variation de *Weighted Manhattan (WMN)* par rapport à *Manhattan (MAN)*.

4.8. RÉSULTATS

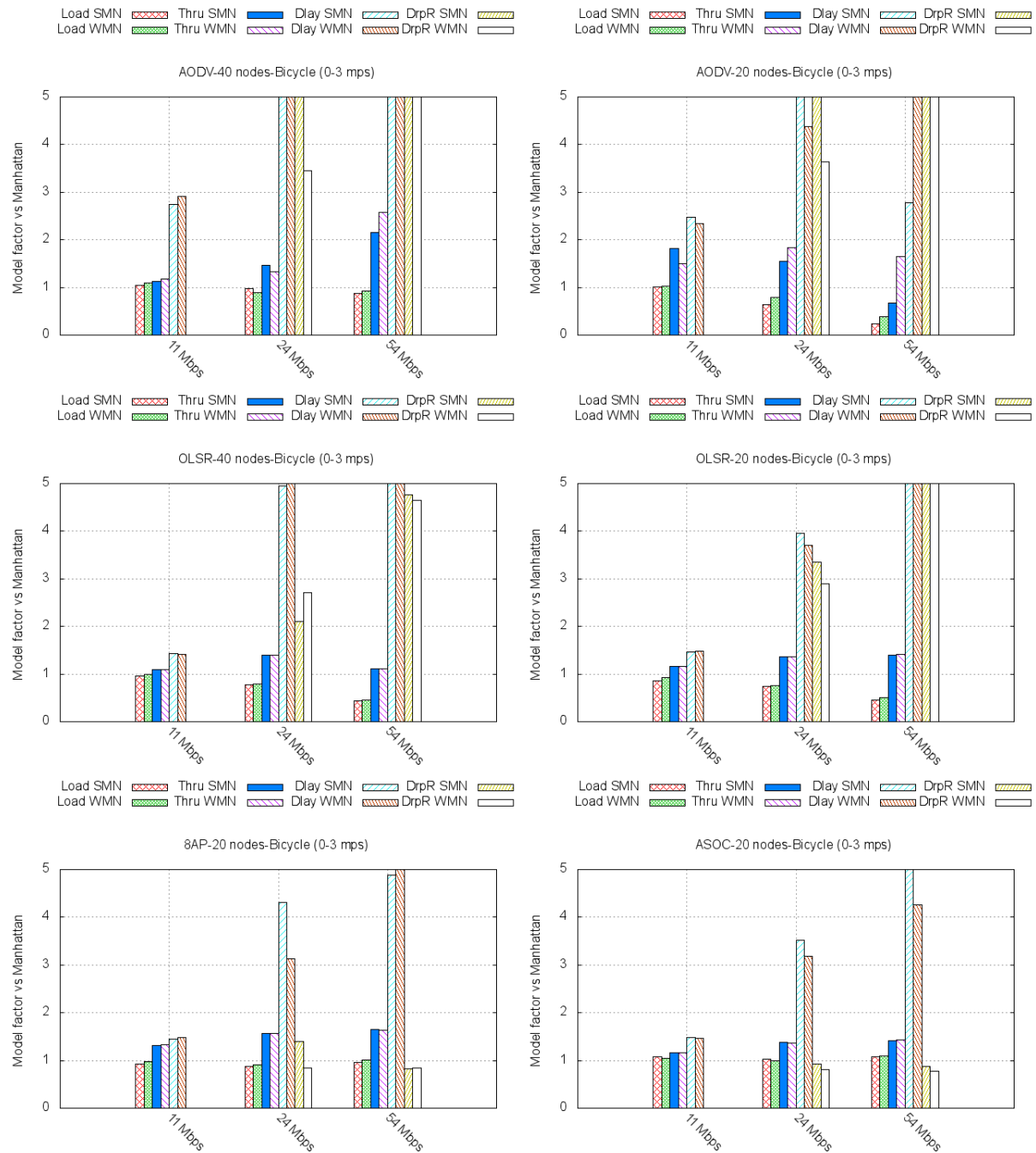


FIGURE 4.6 – WMN bicylette

Résultats Bicyclette (*Bycicle*) La Fig. 4.6 expose les résultats des simulations concernant la vitesse de déplacement des nœuds 0 à 3 *m/s* à savoir celle d'une bicyclette (*bicycle*), en indiquant le facteur de variation de *Weighted Manhattan (WMN)* par rapport à *Manhattan (MAN)*.

4.8. RÉSULTATS

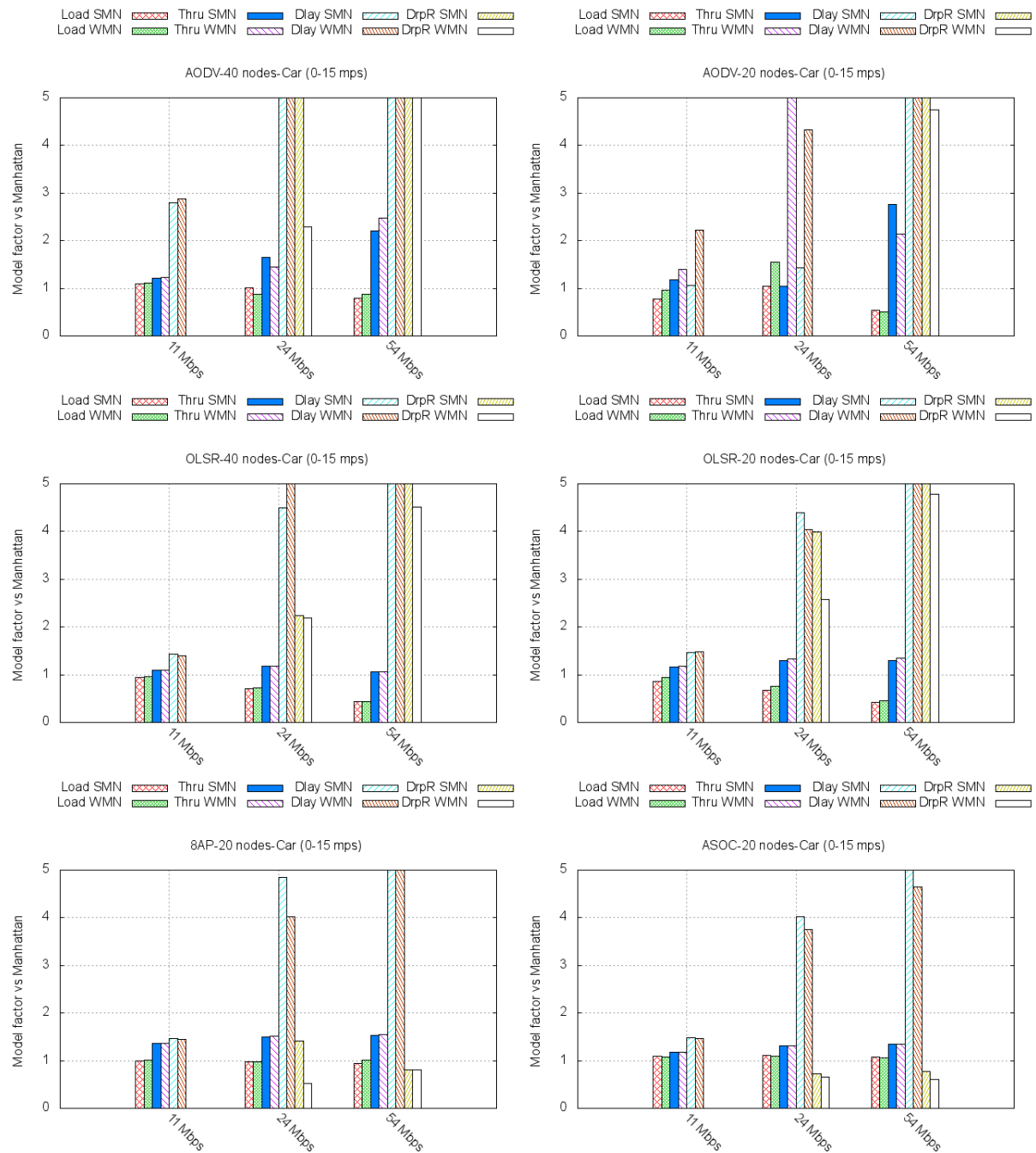


FIGURE 4.7 – WMN véhicule urbain

Résultats véhicule urbain (*Car*) La Fig. 4.7 expose les résultats des simulations concernant la vitesse de déplacement des nœuds 0 à 15 *m/s* à savoir celle d'une Automobile (*car*), en indiquant le facteur de variation de *Weighted Manhattan (WMN)* par rapport à *Manhattan (MAN)*.

4.8. RÉSULTATS

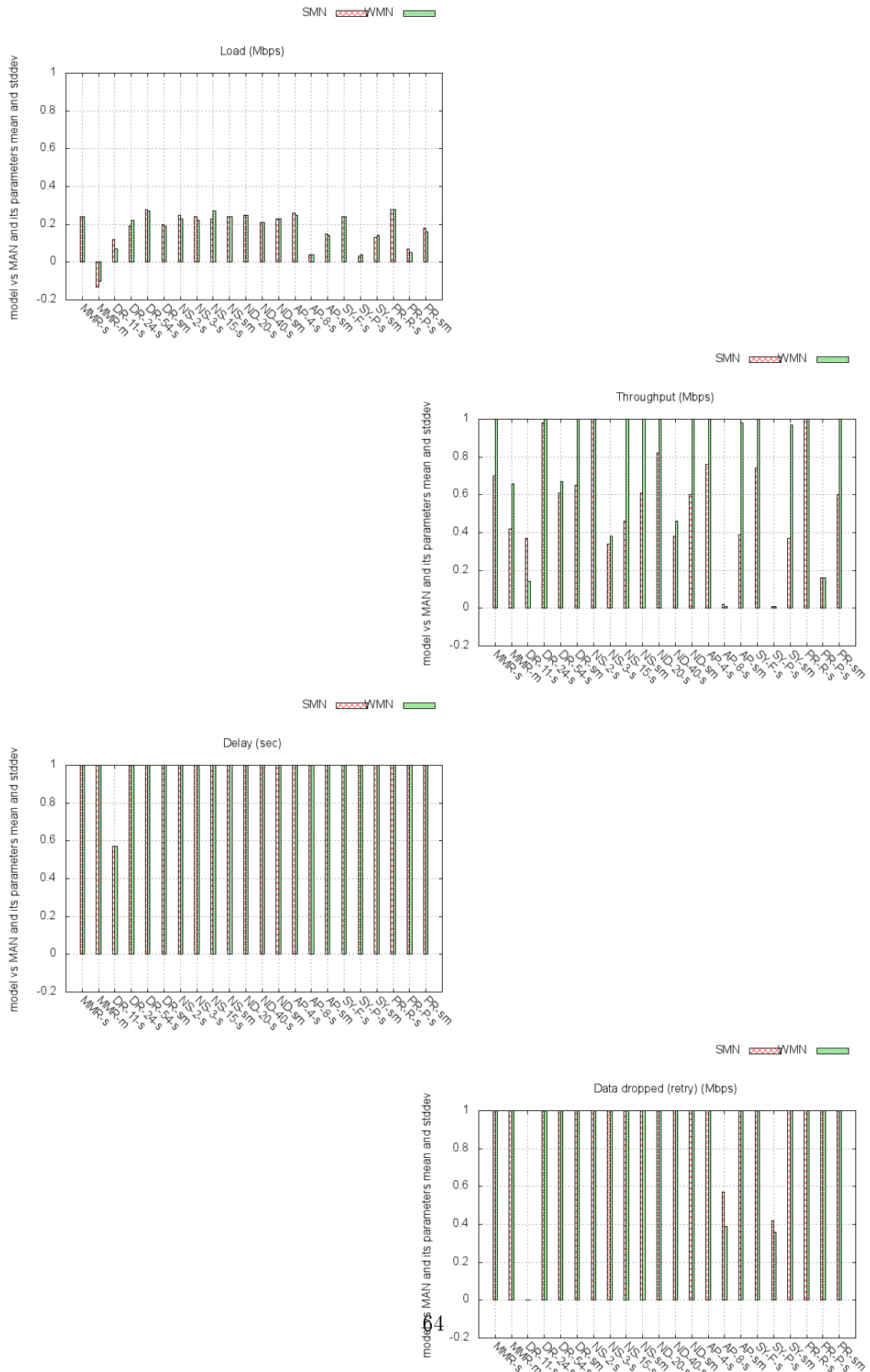


FIGURE 4.8 – WMN - Variables statistiques réseau

La perspective *réseau* présente une synthèse de l'influence de la prise en compte de l'aspect social sur les variables statistiques mesurées. La Fig. 4.8 rassemble les graphiques suivants : Load, Throughput (Thru), Delay (Dlay), Data dropped (Retry Threshold Exceeded, DrpR).

La Table suivante en donne la légende.

MMR	"Model me mean" to "MAN me mean" Ratio with Model in {SMN,WMN,AM1,AM2,AM3} and me in {Load, Thru, Dlay, DrpR}
MMR-s	stddev of MMR of ALL scenarios
MMR-m	mean of MMR of ALL scenarios
DR-n	Data Rate of n Mbps with n in {11,24,54}
NS-n	Node Speed in range 0-n mps with n in {2,3,15}
ND-n	Network Density of n nodes with n in {20,40}
AP-n	Number of Attraction Points with n in {4,8}
SY-t	Sociability Type with t in {(F)ull,(P)artial}
PR-t	Protocol Type with t in {(R)eactive,(P)roactive}
X	MMR of All scenarios covered by X with X in {DR,NS,ND,AP,SY,PR}
X-n	with n in {relative parameter value, see above}
X-n-s	stddev of X-n
X-sm	mean of all X-n-s

Un graphique comporte 22 histogrammes. Notons tout d'abord *MMR*, le rapport ou *ratio* du modèle considéré sur le modèle Manhattan de la variable statistique considérée. Un histogramme donne les moyenne et écart des MMR, d'une part, de façon globale à tous les scénarios de test, puis, d'autre part, par paramètre, pour chacune de ses valeurs, pour les scénarios concernés. C'est cette dernière « moyenne d'écarts » qui dénote l' « influence » du paramètre sur l'ensemble de la mesure. Une précision concernant la Sociabilité s'impose : le caractère entièrement social (*(F)ull*) signifie que tous les nœuds sont respectueux du modèle de mobilité en vigueur ; tandis que la présence d'éléments en « rébellion » ou asociaux (*(P)artial*) est modélisée par environ 15 % des nœuds qui manquent totalement de respect à cet ordre préétabli.

Notons en premier lieu que :

4.8. RÉSULTATS

La moyenne globale de Load lors du passage de Manhattan (MAN) à Weighted Manhattan (WMN) est de : -0,10. Cela signifie que Manhattan est surévalué en l'absence de prise en compte de l'aspect social.

La Table suivante résume l'« influence » des paramètres selon les prérogatives suivantes concernant l'écart global d'un paramètre (dénoté par X-s, conférer la Table légende ci-dessus et l'axe des abscisses des graphiques) :

- inférieur à 0,2 : influence nulle ;
- entre 0,2 et 0,6 : influence faible ;
- entre 0,6 et 1 : influence moyenne ;
- supérieur à 1 influence forte.

	Load	Throughput	Delay	Data dropped
Debit	faible	forte	forte	forte
Vitesse	faible	forte	forte	forte
Densité	faible	forte	forte	forte
Nombre de Points d'Attraction	nulle	moyenne	forte	forte
Sociabilité	nulle	moyenne	forte	forte
Réactif / Proactif	nulle	forte	forte	forte

Comme c'était déjà le cas dans *SMN*, quand le nombre de Points d'Attraction diminue, la concentration des nœuds augmente. Cela a pour effet de réduire le délai de transmission de bout en bout. En effet, il n'y a pratiquement plus besoin de nœuds intermédiaires. En revanche, nombre de données sont perdues pour cause de bourrage du médium air (Data dropped). L'explication est la suivante : la bande de fréquence étant fixe, l'augmentation

de la densité géographique des nœuds résulte en une augmentation du nombre de collisions.

Les variables statistiques réseau *Delay* et *Data dropped* qui représentent le délai entre couches MAC des émetteur et récepteur, d'une part, et, d'autre part, les données abandonnées au niveau de la couche MAC à l'émission pour cause de bourrage du médium-air, ont rendu, pour la totalité des paramètres (débit, vitesse, densité, nombre de Points d'Attraction, Sociabilité et Type de protocole réactif ou proactif), des écarts considérables, très proches de ceux de *SMN* mais sans les atteindre cependant. Pour les autres variables statistiques, Load et Throughput, la tendance de leurs influences a augmenté, par rapport à *SMN*, en devenant « moyenne » voire « forte », et presque jamais « nulle ». Cela nous permet de conclure que la prise en compte de l'aspect social évolutif des Points d'Attraction, c'est-à-dire, le fait de faire croître leurs attractivités en fonction des visites, doit faire l'objet d'attention dans la conception des protocoles de communications dans les réseaux sans fil, au même titre que *SMN*.

4.9 Conclusion

Nous avons élaboré dans ce chapitre, un deuxième modèle de mobilité à Points d'Attraction à attractivité croissant avec le nombre de visites.

Nous avons mis en évidence que le manque de prise en compte de l'aspect social résulte en une surestimation des performances, certes moindre par rapport à *Social Manhattan* mais malgré tout bien présente : -0,10 au lieu de -0,13 de moyenne globale de la variable statistique réseau Load.

Les variables statistiques réseau *Delay* et *Data dropped* ont rendu des écarts considérables, très proches de ceux de *SMN* mais sans les atteindre cependant. Pour les autres variables statistiques, Load et Throughput, la tendance de leurs influences est « moyenne » voire « forte », presque jamais « nulle ». Cela nous permet de conclure que la prise en compte de l'aspect social évolutif des Points d'Attraction, doit faire l'objet d'attention dans la conception des protocoles de communications dans les réseaux sans fil, au même titre que *SMN*.

4.9. CONCLUSION

Chapitre 5

Les systèmes de colonies de fourmis

5.1 Introduction

Dans ce chapitre, nous intégrons dans nos modèles de mobilité sociale, l'*intelligence en essaim* qui désigne les systèmes où l'intelligence de l'ensemble est supérieure à la somme de ses parties. Dans celle-ci, les systèmes de colonies de fourmis, *Ant Colony System (ACS)* forment une des branches la plus importante [Dorigo 1992], [Günes et al. 2002].

Malgré le fait qu'une fourmi soit a priori indépendante et non surveillée (hétérarchie), elle est aidée dans son développement par la communauté et à son tour contribue au bon fonctionnement de celle-ci. La colonie s'auto-gère grâce à un mécanisme de *stigmergie* qui recouvre l'idée d'une communication indirecte entre individus par modification de l'environnement, ici, la dépose de phéromone [Dorigo and Gambardella 1997], Une colonie de fourmis dans son processus de recherche de nourriture au voisinage d'un nid, Fig. 5.1, résout par ce mécanisme, un problème assimilable à celui de la convergence des nœuds d'un réseau mobile vers leurs centres d'intérêts, nos Points d'Attraction. Dans cette optique, la quantité de phéromone à un Point d'Attraction est proportionnelle à sa réputation.

Notons qu'un nœud mobile peut, en plus de ses similarités avec une fourmi, mémoriser l'historique de ses actions, de son environnement, de ses performances. En outre, un nœud mobile peut être équipé de capacités artificielles pour améliorer les performances : i) anticipation ; ii) retour à un état antérieur ; iii) alternatives : dépôt de différentes quantités de phéromone selon des caractéristiques sociales.

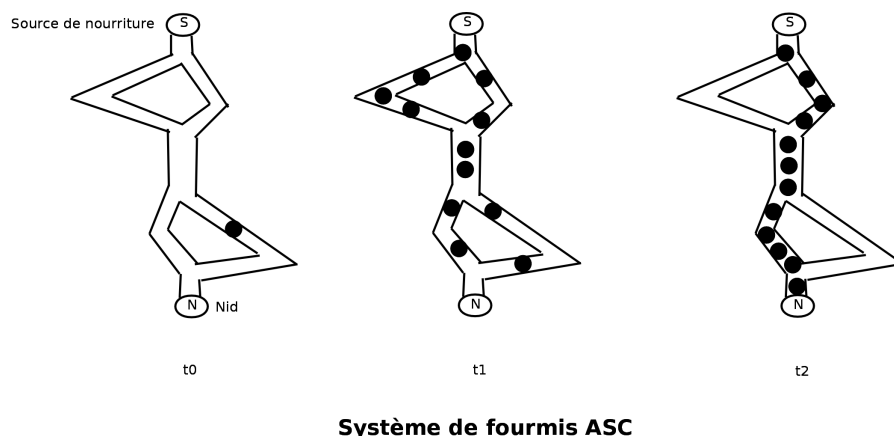


FIGURE 5.1 – Ant Colony Systems (ACS)

Les systèmes de colonies de fourmis (*ACS*) ont été introduits par Dorigo [1992] pour l'optimisation de processus combinatoires tel le problème du voyageur de commerce (*Travelling Salesman Problem, TSP*). Dans le domaine des réseaux mobiles, les ACS ont été utilisés pour la conception des protocoles de routage [Günes et al. 2002], [Open-Mesh.net] cité par Abolhasan et al. [2009]. Dans ce cas, les paquets représentent les fourmis, les liens entre les nœuds les chemins, le nœud destination, la source de nourriture et le nœud source, la fourmilière. Le problème consiste alors à trouver le plus court chemin entre les nœuds source et destination.

Il convient de mentionner aussi que Liao et al. [2004] inspiré par Heck and Ghosh [2000] définit un modèle qui génère des traces au lieu d'utiliser des modèles de mobilité mathématiques.

Dans ce chapitre nous utilisons les *ACS* pour affiner notre modélisation de la mobilité sociale et présenter une nouvelle classe de modèles de mobilité sociale basée sur les systèmes de colonies de fourmis *Ant Mobility Models*. La suite de ce chapitre est organisée comme suit, dans la section 5.2 nous présentons le problème du voyageur de commerce (*TSP*) qui propose l'algorithme *ACS* que nous allons utiliser comme base de notre classe de modèles, le modèle mathématique associé et un exemple d'application *Ant Manhattan Mobility Model (AMN)* sont respectivement présentés en sections 5.3 et 5.4. Une synthèse de tous nos modèles est donnée en section 5.5. Les configurations de tests sont ensuite décrites et les

résultats associés sont présentés et discutés en section 5.6, Enfin la section 5.7 conclue ce chapitre.

5.2 Le problème du voyageur de commerce

Nous allons utiliser le problème du voyageur de commerce (*Travelling Salesman Problem (TSP)*) comme base de notre modèle de mobilité sociale basée sur les systèmes de colonies de fourmis *Ant Mobility Models*

Dorigo and Gambardella [1997] définit l'algorithme de système de colonies de fourmis (*ACS*) et l'applique au problème du voyageur de commerce *Traveling Salesman Problem (TSP)*, qui étant donné un ensemble de villes séparées par des distances données, consiste à trouver le plus court chemin qui relie toutes les villes. Comme notre modèle se base sur ce système, nous décrivons ici le formalisme associé. Soit :

- $V = \{a, \dots, z\}$ un ensemble de villes ;
- $A = \{(r, s) : r, s \in V\}$ un ensemble d'arcs ;
- $\delta(r, s) = \delta(s, r)$ la longueur de l'arc $(r, s) \in A$;
- $\tau(r, s)$ la quantité de *phéromone* sur un arc $(r, s) \in A$;
- Règle de transition :

$$p_k(r, s) = \begin{cases} \frac{[\tau(r, s)] \cdot [\eta(r, s)]^\beta}{\sum_{u \in J_k(r)} [\tau(r, u)] \cdot [\eta(r, u)]^\beta} & \text{si } s \in J_k(r) \\ 0 & \text{sinon} \end{cases} \quad (5.1)$$

<> $\eta = \frac{1}{\delta}$ l'inverse de la distance $\delta(r, s)$;

<> $J_k(r)$ l'ensemble des villes restant à visiter par la fourmis k positionnée à la ville r ;

<> β l'importance relative de la phéromone par rapport à la distance ($\beta > 0$).

- Règle de mise à jour de la phéromone :

$$\tau(r, s) \leftarrow (1 - \alpha) \times \tau(r, s) + \sum_{k=1}^m \Delta\tau_k(r, s)$$

$$\Delta\tau_k(r, s) = \begin{cases} \frac{1}{L_k} & \text{si } (r, s) \in \text{tour fait par la fourmi } k \\ 0 & \text{sinon} \end{cases}$$

- $0 < \alpha < 1$ paramètre de décroissance de la phéromone ;
- L_k longueur du tour fait par la fourmi k ;
- m nombre total de fourmis.

Fig. 5.2 présente l'algorithme associé.

```

Initialize
Loop /* iteration loop */
  Each ant is positioned on a starting node
  Loop /* step loop */
    Each ant applies the state transition rule
      to incrementally build a solution
  Until all ants have built a complete solution
  A pheromone updating rule is applied
Until End_condition

```

FIGURE 5.2 – Algorithme des colonies de fourmis pour le TSP

5.3 Modélisation

Nous introduisons le mécanisme de phéromone de la section précédente, dans notre modèle mathématique. Eq. 4.1, p. 48 devient :

$$r_1(a, n, t) = \frac{\frac{r(a, t)}{d(a, n, t)} + \tau(a, t)}{\sum_{a' \in A} \frac{r(a', t)}{d(a', n, t)} + \tau(a', t)}$$

où $\tau(a, t)$ est la quantité de phéromone au Point d'Attraction a à l'instant t tel que :

$$\tau(a, t) = \begin{cases} 0 & t = 0 \\ (1 - \alpha)\tau(a) & t \neq 0 \\ \tau(a, t) + w(n) & n \text{ arrive à } a \end{cases} \quad (5.2)$$

où $0 \leq \alpha \leq 1$ est le taux de décroissance i.e. d'évaporation de la phéromone.

Enfin, pour modéliser l'importance relative de la distance par rapport au taux d'évaporation de la phéromone, l'équation Eq. 4.3, p. 49 devient : n attiré par $a \iff$

$$a = \arg \max_{a' | v(a', n, t) = T} \frac{r(a', t)}{d(a', n, t)^\beta} \quad (5.3)$$

où β permet de paramétrer l'importance relative de la distance par rapport au taux d'évaporation de la phéromone. ($\beta > 0$).

5.4 Application : Ant Manhattan (AMN)

Comme pour les modèles *Social Manhattan (SMN)* et *Weighted Manhattan (WMN)*, un exemple d'application est donné en utilisant comme base le modèle *Manhattan (MN/MAN)* auquel nous appliquons notre modèle mathématique.

Rappelons que *AMN* selon notre classification sociale donne ceci :

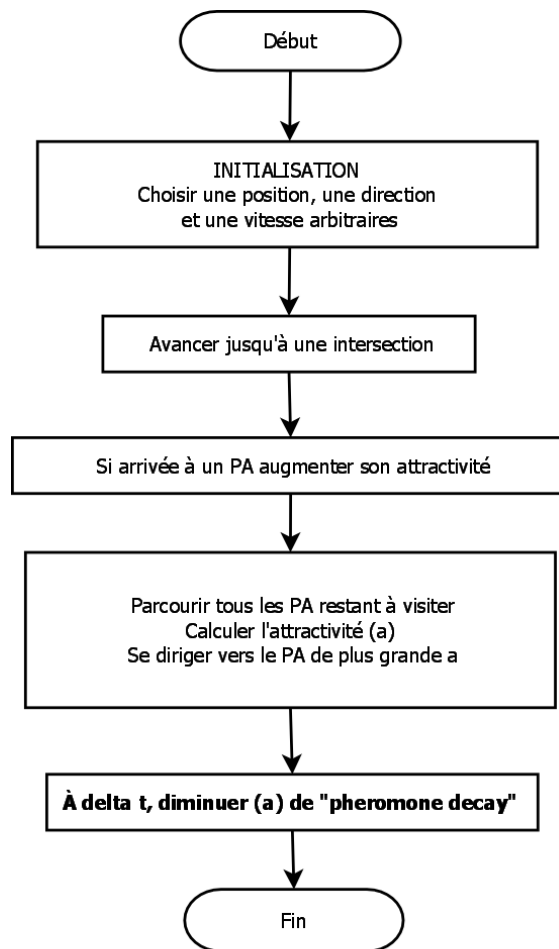
Région	Topologie en Grille
Nœuds	$N \geq 2$
Communauté	<i>social</i> : → avoir les même centres d'intérêts ;
Comportement	<i>social</i> : → attirés par les mêmes Points d'Attraction (PA) ; → désintéressement dans la durée ;
Déplacement	<i>social</i> : suivant ACS → vitesse aléatoire des nœuds ; → se déplacer suivant une ligne/colonne ; → à une intersection, choisir de se déplacer vers un PA, suivant le modèle ASC ; → l' <i>attractivité</i> des PA n'est pas constante : elle peut croître ou décroître selon ASC.

Son schéma fonctionnel est donné en Fig. 5.3.

Le modèle *Ant Manhattan* rajoute *Pheromone Decay Factor* et *Pheromone Distance Exponent* qui implémente les paramètres α et β , cf. les équations Eq. 5.2 et Eq. 4.3, p. 49.

5.5 Synthèse de nos modèles de mobilité

Le tableau suivant résume nos modèles de mobilité décrits selon les critères de notre classification sociale :



Ant Manhattan

FIGURE 5.3 – Mouvement d'un nœud selon Ant Manhattan : Schéma fonctionnel

Modèle	MN	SMN	WMN	AMN
Région	Grille	Grille	Grille	Grille
Communauté	<i>non social</i>	<i>social</i> : mêmes intérêts	<i>social</i> : mêmes intérêts	<i>social</i> : mêmes intérêts
Comportement	<i>non social</i>	<i>social</i> : attirés par mêmes PA	<i>social</i> : attirés par mêmes PA	<i>social</i> : attirés par mêmes PA
Mouvement	<i>non social</i>	<i>social</i> : Harfouche et al. [2009]	<i>social</i> : Harfouche et al. [2010]	<i>social</i> : ACS

Le tableau suivant résume les attributs ajoutés pour chacun de nos modèles de mobilité :

Parameter	MN	SMN	WMN	AMN
Safety Distance	x	x	x	x
Move Step	x	x	x	x
Block Length	x	x	x	x
Maximum Acceptable Factor			x	x
Pheromone Decay				x

5.6 Résultats

5.6.1 Introduction

Dans cette section, nous relatons les tests de simulation effectués selon les prérogatives de la section précédente. Les modèles *Manhattan (MN/MAN)* et *Ant Manhattan (AMN)* sont rapprochés afin de mettre en évidence leurs différences. Le(s) rapprochement(s) présenté(s) au(x) chapitre(s) précédent(s) est(sont) également conservé(s) afin de le(s) mettre en perspective avec le présent rapprochement.

5.6.2 Configurations de test

Pour récapituler, les nouveaux paramètres de notre modèle sont :

- *Pheromone Decay factor* : α ;
- *Pheromone Distance Exponent* : β .

Nous déclinons nos tests selon les critères suivants :

- modèle de mobilité ;
- paramètres du modèle ;
- densité du réseau mobile ;
- nombre de Points d'Attraction.

Nous rapprochons notre modèle fourmis (*AMN*) du modèle Manhattan de base (*MN* ou *MAN*) pour mettre en perspective les apports de notre modèle.

Il va de soi que ce rapprochement nécessite un *cadre de test* commun. C'est la raison pour laquelle nous décidons de fixer certains paramètres de simulation. Il s'agit des paramètres suivants, nous utilisons ci-dessous leurs équivalents dans l'implémentation :

- *Safety Distance* : distance de sécurité routière ; valeur : 30 m, valeur moyenne en zone urbaine ;

5.6. RÉSULTATS

- *Move Step* : unité de mouvement d'un nœud mobile ; valeur : 25 m, pour ne pas alourdir les temps de simulation ; diviseur du *Block Length* afin que les nœuds « passent » par les intersections ;
- *Block Length* : distance entre 2 lignes ou 2 colonnes de la grille ; valeur : 50 m, largeur moyenne d'un pâté de maisons ;
- x_{min} , y_{min} , x_{max} , y_{max} : aire de simulation ; valeur : 1 km^2 , superficie moyenne d'un quartier.

Pour les paramètres de simulation variables, les valeurs choisies nous apparaissent suffisantes dans un premier temps pour connaître l'influence des paramètres correspondants sur les modèles étudiés. Il s'agit de :

- protocole de routage ; valeurs : OLSR et AODV, les plus utilisés dans les réseaux sans fil et les plus représentatifs des grands ensembles que sont les protocoles réactifs et les protocoles proactifs, respectivement ;
- débit ; valeurs : 11, 24 et 54 Mbit/s, un débit faible, un autre moyen, un débit plus important ;
- vitesse *maximale* des nœuds ; valeurs : 2, 3 et 15 m/s soit 7 km/h pour les piétons, 11 km/h pour les bicyclettes et 52 km/h pour les véhicules urbains ;
- nombre de nœuds ; valeurs : 20 nœuds, un groupe moyen (classe d'école...) et 40 nœuds, un groupe plus important (groupe de touristes, conférence...) ;
- nombre de Points d'Attraction ; valeurs : 4, quartier périphérique et 8, centre ville ;

Les différentes valeurs de nos paramètres de simulation sont résumés en Table 5.1.

Les résultats sont exposés selon 2 angles de vue.

En premier lieu, ils suivent la perspective *social* : piéton, bicyclette et véhicule urbain.

En second lieu, ils suivent la perspective *réseau* : Load, Throughput (Thru), Delay (Dlay), Data dropped (Retry Threshold Exceeded, DrpR).

Pour ce qui est de la perspective *social*, nous proposons une figure par *Vitesse de nœud* : piéton, bicyclette et véhicule urbain. Chacune de ces figures comporte 2 colonnes et 3 lignes qui renferment de gauche à droite et de haut en bas, les résultats suivants :

- *AODV 40* ;
- *AODV 20* ;

TABLE 5.1 – Simulation

Simulateur	OPNET Modeler 16.0
Couche MAC	IEEE 802.11
Taux de transmission	250 m
Type de trafic	UDP
Destination	aléatoire
Caractéristiques physiques	Extended Rate PHY (802.11g)
Protocoles de routage	OLSR, AODV
Débit (Mbit/s)	11, 24, 54
Vitesse des nœuds (m/s)	2 (7 km/h, piéton, <i>pedestrian</i> , <i>ped</i>) 3 (11 km/h, bicyclette, <i>bicycle</i> , <i>bic</i>) 15 (52 km/h, véhicule, <i>car</i>)
Nombre de nœuds	20, 40
Aire de simulation	1000 m × 1000 m
Block length	50 m
Safety distance	30 m
Move step	25 m
Nombre de Points d'Attraction	4, 8
Attractivity	5
Modèle de mobilité	Manhattan (MN) Social Manhattan (SMN) Weighted Social Manhattan (WMN) Ant Manhattan (AMN)
(α, β)	AMN1 : 0.005 : norm. decay, 1.5 : norm. dist. AMN2 : 0.002 : weak decay, 2 : heavy dist. AMN3 : 0.01 : heavy decay , 0.5 : weak dist.

5.6. RÉSULTATS

- *OLSR 40*;
- *OLSR 20*;
- *OLSR 20 8AP*;
- *OLSR 20 ASOC*.

Où :

- 20 et 40 représentent la densité du réseau ;
- 8AP représente le doublement des Points d'Attraction (*Attraction Points, AP*) de 4 à 8 et
- ASOC représente la « rébellion » d'une proportion de nœuds qui sont totalement asociaux en ce sens où ils ne respectent aucunement les modèles de mobilité de l'ensemble.

Chacun de ces résultats renferme les paramètres suivants :

- les 3 débits les plus usités : 11, 24 et 54 Mbit/s et
- les 4 mesures réseau classiques : Load, Throughput (Thru), Delay (Dlay) et Data Dropped (Retry Threshold exceeded) (DrpR).

CHAPITRE 5. LES SYSTÈMES DE COLONIES DE FOURMIS

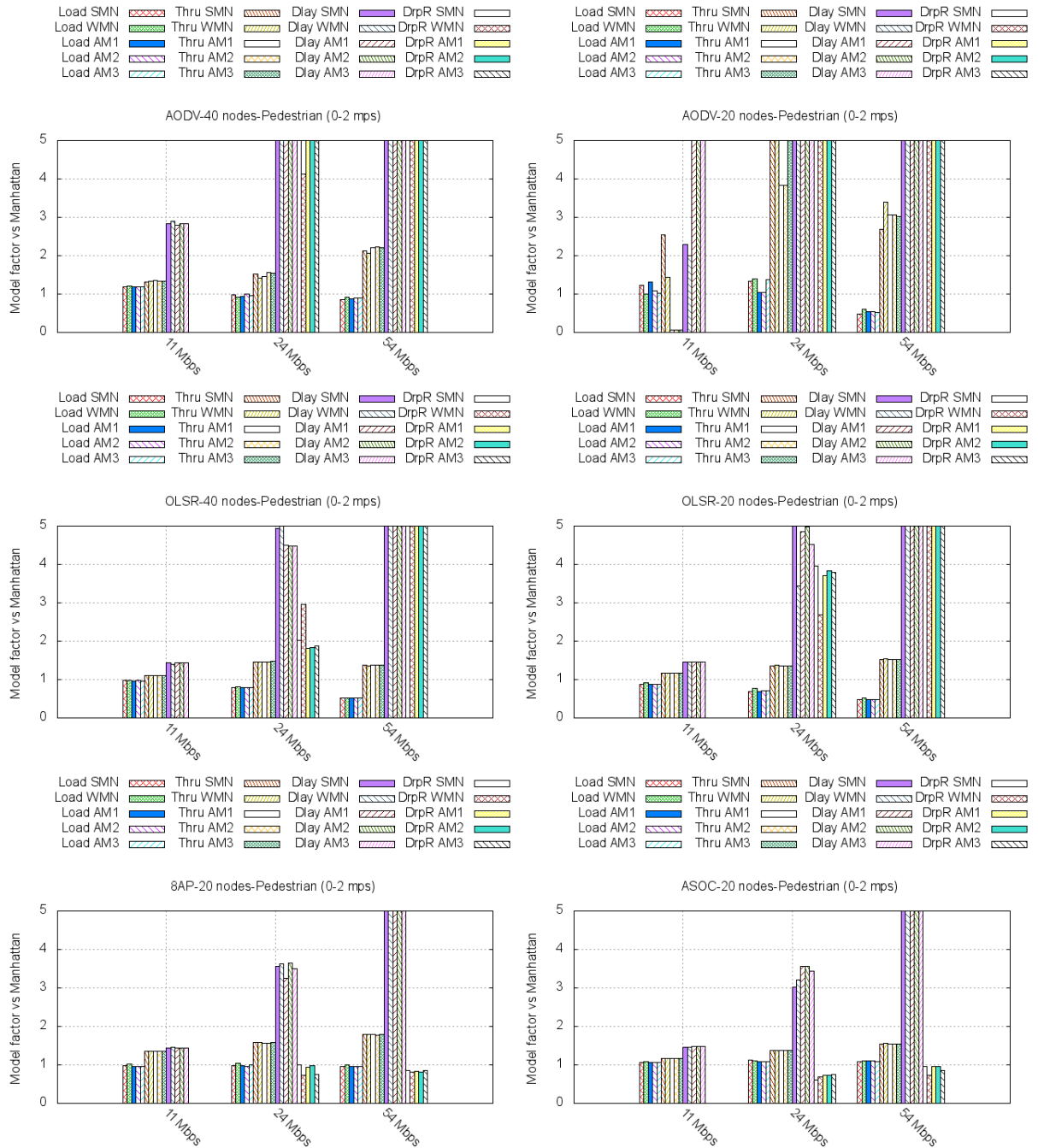


FIGURE 5.4 – AMN piéton

5.6. RÉSULTATS

La Fig. 5.4 expose les résultats des simulations concernant la vitesse de déplacement des nœuds 0 à 2 m/s à savoir celle d'un piéton (*pedestrian*), en indiquant le facteur de variation d'*Ant Manhattan (AMN)* par rapport à *Manhattan (MAN)*.

CHAPITRE 5. LES SYSTÈMES DE COLONIES DE FOURMIS

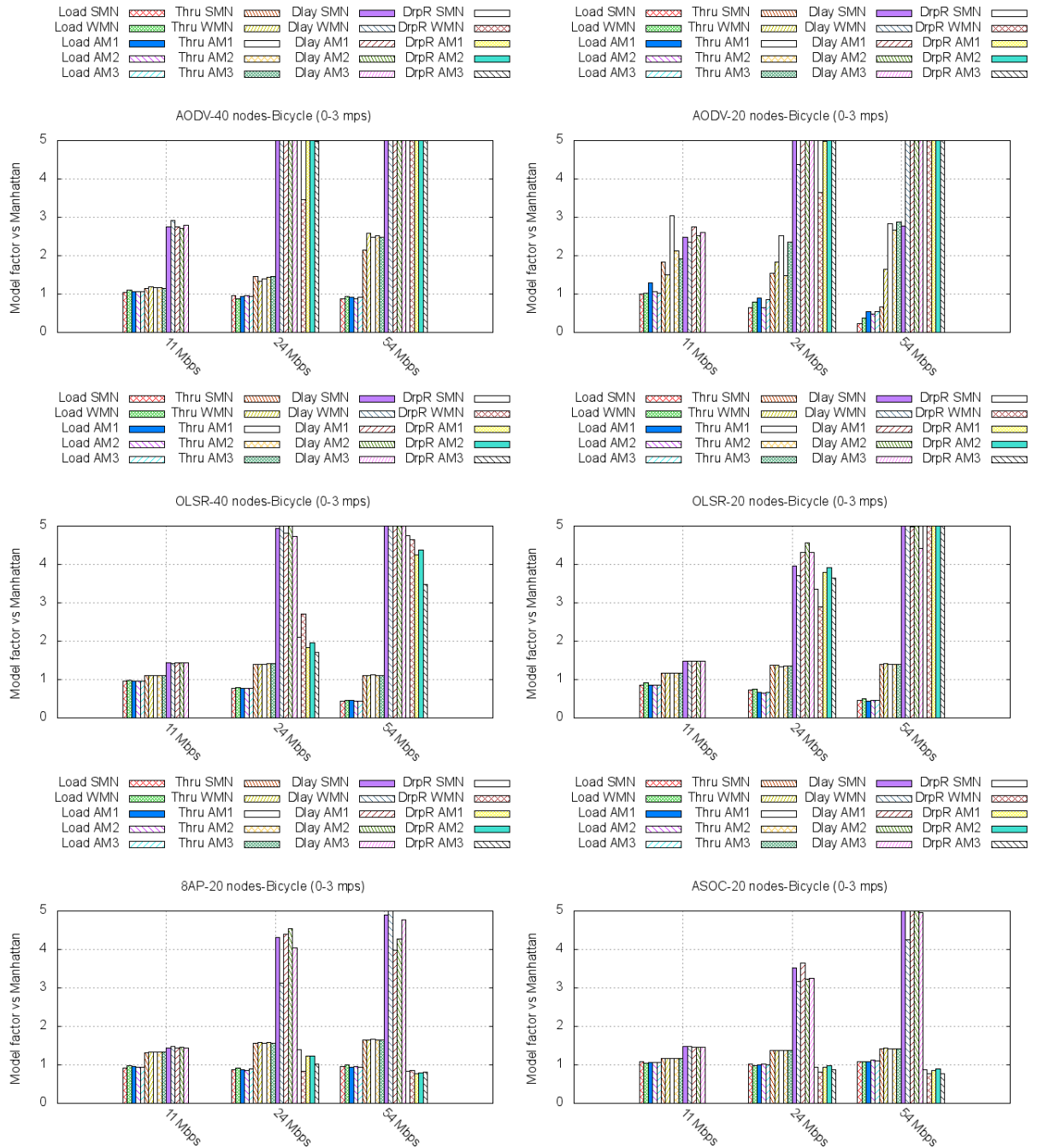


FIGURE 5.5 – AMN bicyclette

5.6. RÉSULTATS

La Fig. 5.5 expose les résultats des simulations concernant la vitesse de déplacement des nœuds 0 à 3 m/s à savoir celle d'une bicyclette (*bicycle*), en indiquant le facteur de variation d'*Ant Manhattan (AMN)* par rapport à *Manhattan (MAN)*.

CHAPITRE 5. LES SYSTÈMES DE COLONIES DE FOURMIS

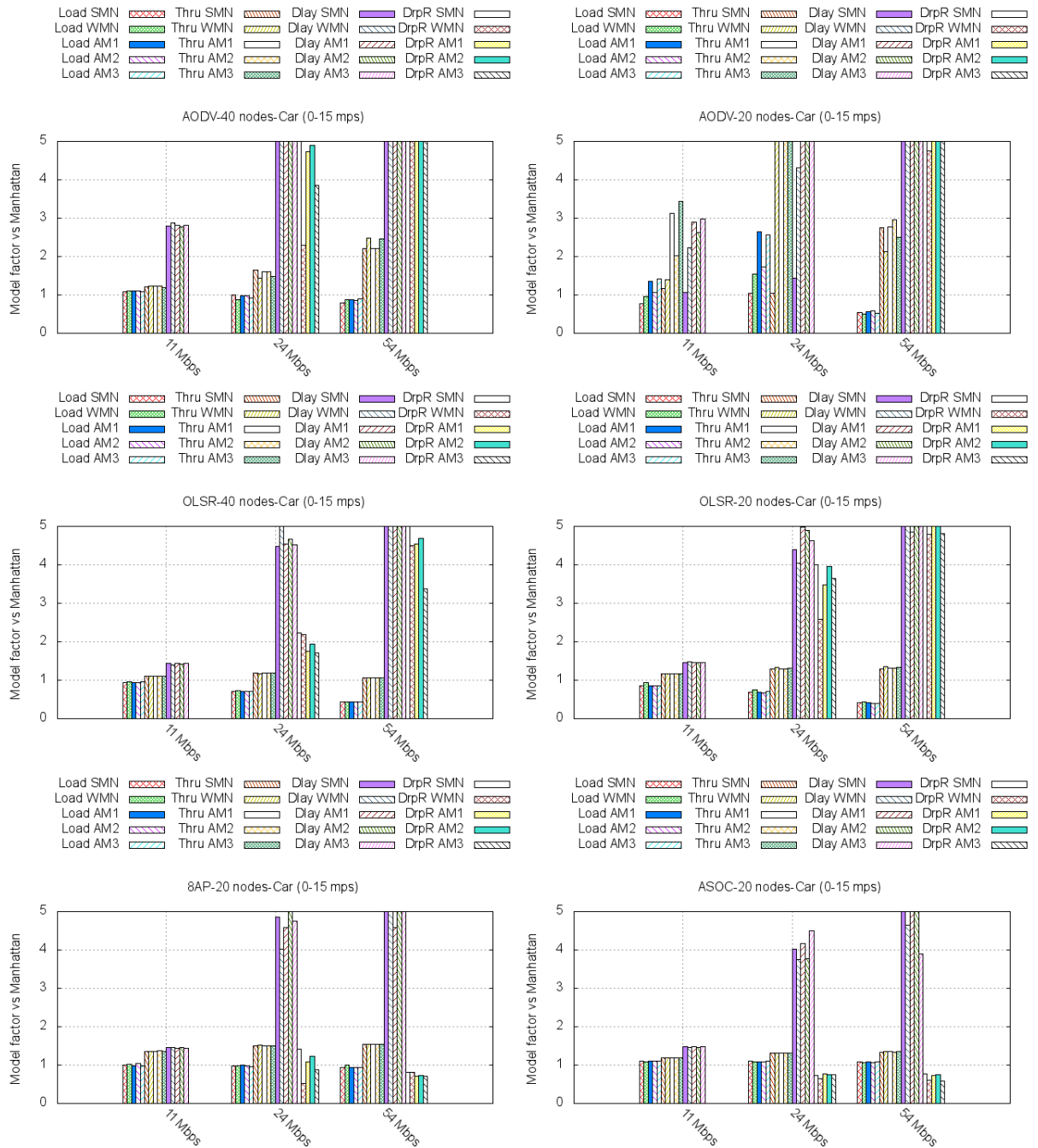


FIGURE 5.6 – AMN véhicule urbain

Résultats véhicule urbain (*Car*) La Fig. 5.6 expose les résultats des simulations concernant la vitesse de déplacement des nœuds 0 à 15 m/s à savoir celle d'une Automobile (*car*), en indiquant le facteur de variation d'*Ant Manhattan (AMN)* par rapport à *Manhattan (MAN)*.

CHAPITRE 5. LES SYSTÈMES DE COLONIES DE FOURMIS

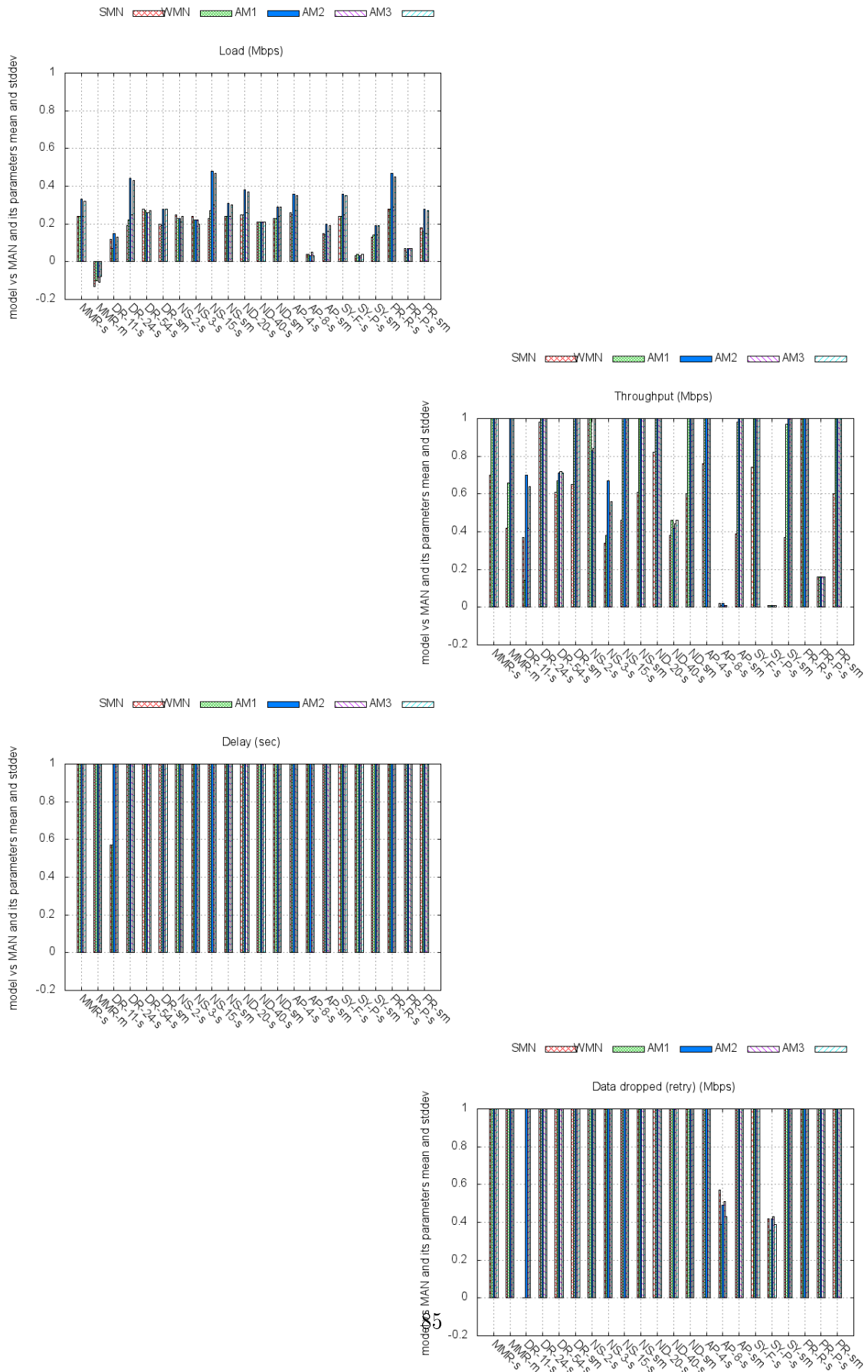


FIGURE 5.7 – AMN - Variables statistiques réseau

5.6. RÉSULTATS

La perspective *réseau* présente une synthèse de l'influence de la prise en compte de l'aspect social sur les variables statistiques mesurées. La Fig. 5.7 rassemble les graphiques suivants : Load, Throughput (Thru), Delay (Dlay), Data dropped (Retry Threshold Exceeded, DrpR).

La Table suivante en donne la légende.

MMR	"Model me mean" to "MAN me mean" Ratio with Model in {SMN,WMN,AM1,AM2,AM3} and me in {Load, Thru, Dlay, DrpR}
MMR-s	stddev of MMR of ALL scenarios
MMR-m	mean of MMR of ALL scenarios
DR-n	Data Rate of n Mbps with n in {11,24,54}
NS-n	Node Speed in range 0-n mps with n in {2,3,15}
ND-n	Network Density of n nodes with n in {20,40}
AP-n	Number of Attraction Points with n in {4,8}
SY-t	Sociability Type with t in {(F)ull,(P)artial}
PR-t	Protocol Type with t in {(R)eactive,(P)roactive}
X	MMR of All scenarios covered by X with X in {DR,NS,ND,AP,SY,PR}
X-n	with n in {relative parameter value, see above}
X-n-s	stddev of X-n
X-sm	mean of all X-n-s

Un graphique comporte 22 histogrammes. Notons tout d'abord *MMR*, le rapport ou *ratio* du modèle considéré sur le modèle Manhattan de la variable statistique considérée. Un histogramme donne les moyenne et écart des MMR, d'une part, de façon globale à tous les scénarios de test, puis, d'autre part, par paramètre, pour chacune de ses valeurs, pour les scénarios concernés. C'est cette dernière « moyenne d'écarts » qui dénote l' « influence » du paramètre sur l'ensemble de la mesure. Une précision concernant la Sociabilité s'impose : le caractère entièrement social (*(F)ull*) signifie que tous les nœuds sont respectueux du modèle de mobilité en vigueur ; tandis que la présence d'éléments en « rébellion » ou asociaux (*(P)artial*) est modélisée par environ 15 % des nœuds qui manquent totalement de respect à cet ordre préétabli.

Notons en premier lieu que :

La moyenne globale de Load lors du passage de Manhattan (MAN) à Ant Manhattan (AMN) est de : -0,10. Cela signifie que Manhattan est surévalué en l'absence de prise en compte de l'aspect social.

La Table suivante résume l'« influence » des paramètres selon les prérogatives suivantes concernant l'écart global d'un paramètre (dénoté par X-s, conférer la Table légende ci-dessus et l'axe des abscisses des graphiques) :

- inférieur à 0,2 : influence nulle ;
- entre 0,2 et 0,6 : influence faible ;
- entre 0,6 et 1 : influence moyenne ;
- supérieur à 1 influence forte.

	Load	Throughput	Delay	Data dropped
Debit	faible	forte	forte	forte
Vitesse	faible	forte	forte	forte
Densité	faible	forte	forte	forte
Nombre de Points d'Attraction	nulle	moyenne	forte	forte
Sociabilité	nulle	moyenne	forte	forte
Réactif / Proactif	faible	forte	forte	forte

À nouveau, quand le nombre de Points d'Attraction diminue, la concentration des nœuds augmente. Cela a pour effet de réduire le délai de transmission de bout en bout. En effet, il n'y a pratiquement plus besoin de nœuds intermédiaires. En revanche, nombre de données sont perdues pour cause de bourrage du médium air (Data dropped). L'explication

est la suivante : la bande de fréquence étant fixe, l'augmentation de la densité géographique des nœuds résulte en une augmentation du nombre de collisions.

Les variables statistiques réseau *Delay* et *Data dropped* qui représentent le délai entre couches MAC des émetteur et récepteur, d'une part, et, d'autre part, les données abandonnées au niveau de la couche MAC à l'émission pour cause de bourrage du médium-air, ont rendu, pour la totalité des paramètres (débit, vitesse, densité, nombre de Points d'Attraction, Sociabilité et Type de protocole réactif ou proactif), des écarts considérables, compris entre ceux de *SMN* et ceux de *WMN*. Pour les autres variables statistiques, Load et Throughput, la tendance de leurs influences a augmenté, par rapport à *SMN*, comme pour *WMN*, en devenant « moyenne » voire « forte », et presque jamais « nulle ». Notons toutefois, le passage d'une influence « forte » à une influence « plus forte » du choix du Type de protocole de routage (Réactif/Proactif) pour la variable statistique réseau *Throughput*. Cela nous permet de conclure que la prise en compte de l'aspect social évolutif des Points d'Attraction, c'est-à-dire, le fait de faire croître leurs attractivités en fonction des visites MAIS AUSSI, et c'est là la spécificité d'*AMN*, le fait de la faire décroître en fonction de la raréfaction des visites, doit faire l'objet d'attention dans la conception des protocoles de communications dans les réseaux sans fil, au même titre qu'*SMN* et *WMN*.

5.7 Conclusion

Nous avons élaboré dans ce chapitre, un deuxième modèle de mobilité à Points d'attraction à attractivité croissant avec le nombre de visites.

Nous avons mis en évidence que le manque de prise en compte de l'aspect social résulte en une surestimation des performances, certes moindre par rapport à *Social Manhattan* mais malgré tout bien présente, dans *Weighted Manhattan* : -0,10 au lieu de -0,13 de moyenne globale de la variable statistique réseau Load.

Les variables statistiques réseau *Delay* et *Data dropped* ont rendu des écarts considérables, compris entre ceux de *SMN* et ceux de *WMN*. Notons toutefois, le passage d'une influence « forte » à une influence « plus forte » du choix du Type de protocole de routage (Réactif/Proactif) pour la variable statistique réseau *Throughput*. Cela nous permet de

conclure que la prise en compte de l'aspect social évolutif des Points d'Attraction, c'est-à-dire, le fait de faire croître leurs attractivités MAIS AUSSI le fait de la faire décroître en fonction des visites, doit faire l'objet d'attention dans la conception des protocoles de communications dans les réseaux sans fil, au même titre qu'*SMN* et *WMN*.

5.7. CONCLUSION

Chapitre 6

Les protocoles de routage sociaux

6.1 Introduction

Dans les chapitres précédents, nous avons traité des aspects sociaux de la mobilité en vue d'en améliorer la modélisation ; celle-ci ayant comme objectif unique de rendre plus réalistes les résultats des mesures de performances en milieu simulé des protocoles de routage des réseaux mobiles. Dans ce chapitre, nous abordons les protocoles de routage eux-mêmes en tant que nouvel espace prospectif alternatif à l'utilisation des comportements communautaires dans les réseaux mobiles. Après un succinct état de l'art en section 6.2, nous nous proposons de revisiter les tenants et les aboutissants du protocole *Optimized Link State Routing (OLSR)* en section 6.3. Nous signalons des résultats intéressants concernant : la gestion de l'énergie, *e-OLSR*, en section 6.4, ces résultats nous ont inspirés dans notre démarche d'optimisation du protocole *OLSR*. Ensuite, en section 6.5, nous passons au cœur de ce chapitre qui est une approche sociale du protocole *OLSR*, *OLSR-S*. En section 6.6, nous décrivons notre algorithme *OLSR-S* et nous testons cette approche par la simulation d'un réseau mobile mettant en œuvre ce protocole en section 6.7. Nous relatons les résultats des mesures réalisées sur les différents organes de notre réseau mobile simulé en section 6.8 et concluons notre chapitre en section 6.9 sur les avantages d'une telle démarche.

6.2 État de l'art des protocoles sociaux

« L'acheminement d'un message dans un réseau mobile épars est rendu difficile par le fait que le graphe du réseau est rarement (voire jamais) connexe. Dans ces conditions où les nœuds sont libres dans leur déplacement, trouver un chemin qui offre de bonnes performances d'acheminement de bout en bout en un temps réduit est un véritable défi », écrit Daly and Haahr [2007b] dans un article qui présente une solution multidisciplinaire fondée sur la prise en compte de l'effet du petit monde (*small world dynamics*). Celui-ci a été proposé initialement comme levier facilitant les études économiques ou sociales. Récemment, il a été appliqué avec succès à la propagation des informations caractéristiques des réseaux sans fil. À cette fin, certains nœuds ponts sont identifiés en fonction de leur aptitude à être central (au sens central téléphonique), c'est-à-dire, selon leur faculté à jouer le rôle de courtier d'échange d'informations entre des nœuds qui seraient sans cela déconnectés. Compte tenu de la difficulté de mesurer la centralité dans les réseaux peuplés, la notion de *réseaux ego* est exploitée où les nœuds ne sont pas tenus d'échanger des informations globales sur la topologie du réseau, mais uniquement locales. Le routage *SimBet* [Daly and Haahr 2007a] est ensuite proposé, lequel exploite la centralité prévisionnelle (*betweenness*) ainsi que la similitude sociale au nœud destination (*similarity*), déterminée localement. L'article présente des simulations utilisant des données de traces réelles qui démontrent que le routage *SimBet* donne des résultats en termes d'acheminement des messages qui sont proches du routage *épidémique* [Yoneki et al. 2008], mais pour un coût considérablement réduit. De plus, l'article montre que le routage *SimBet* surpasse le protocole de routage *PRoPHET* de Lindgren et al. [2003], en particulier lorsque les nœuds émetteur et récepteur ont une connectivité faible.

Costa et al. [2008] énonce que les applications qui nécessitent la diffusion d'informations destinées aux humains (publicité, nouvelles, alertes environnement) s'appuient souvent sur le modèle publication-abonnement (*publish-and-subscribe*), dans lequel le réseau délivre (un message relatif à) une publication uniquement à ses (nœuds) abonnés. Par essence, le modèle publication-abonnement est particulièrement adapté aux environnements mobiles, en ce sens où il minimise le couplage entre les partis communicants. Toutefois, à notre connais-

sance, aucun des (rares) travaux qui abordent le modèle publication-abonnement dans les environnements mobiles, n'a encore traité des réseaux humains connectés de façon intermittente. Les personnes socialement liées ont souvent des collocations proches. Cette caractéristique peut être prise en compte dans les décisions d'acheminement au sein d'une couche transport centrée sur l'intérêt, laquelle sous-tend le réseau de publication-abonnement. Cela produit de meilleures performances et accroît la tolérance aux taux élevés de mobilité ainsi que la capacité à endiguer de longs épisodes de déconnexion. Costa et al. [2008] propose *SocialCast*, un cadre de travail pour le routage de type publication-abonnement qui exploite des prédictions basées sur la mesure de l'interaction sociale (comme par exemple les modes de déplacement à l'intérieur d'une communauté) pour identifier les meilleurs porteurs d'informations. Ce travail met en évidence les principes qui sous-tendent ce protocole, illustre son fonctionnement et évalue ses performances en utilisant un modèle de mobilité basé sur un réseau social validé à l'aide de traces réelles. L'analyse des résultats montre que la prédiction des collocation et mobilité des nœuds permet de publier à leur intention, des événements à un rythme régulier et soutenu, en maintenant au plus bas les trafic de contrôle et latence, indépendamment des fluctuations au niveau de la densité, du nombre de répliquas par message et de la vitesse.

Un protocole de routage pour réseaux mobiles, sensible aux relations sociales, conçu pour le partage de contenu des applications sociales mobiles, est proposé par An et al. [2009]. Depuis qu'un contenu peut être partagé par un groupe d'utilisateurs qui ont des intérêts similaires, cette similitude est un bon outil de prédiction de « *qui va consommer quel contenu* ». Ces intérêts communs peuvent en effet être exploités dans le routage et la réplification des demandes/réponses de contenu afin d'accroître l'efficacité de leur partage. Remarquons que le routage seul détermine quel contenu sera transmis par qui. De plus, un itinéraire est basé sur la similarité des intérêts. Il vient que la réplification du contenu peut avoir lieu dans chacun des nœuds intermédiaires, ces nœuds étant intéressés par ce contenu. L'auteur décrit la conception et la mise en œuvre d'un tel protocole de routage qui, sensible aux relations sociales, donne à celles-ci une place centrale dans le système d'évaluation (*metrics*) du routage. Les résultats de simulation montrent que le protocole de routage proposé a un taux d'acheminement plus élevé et un flux de contrôle réseau plus

réduit que ceux du protocole de routage *dynamic Destination-Sequenced Distance-Vector*, *DSDV*, proposé par Perkins and Bhagwat [1994].

6.3 Rappel sur le protocole *OLSR*

Le protocole *Optimized Link State Routing*, *OLSR* (Routage à État de lien Optimisé) peut être décrit comme une adaptation du *Open Shortest Path First (OSPF)*, *RFC 2328*, pour les réseaux mobiles. L'optimisation consiste en l'introduction de la notion de *MultiPoint Relay*, *MPR* (Relais Multipoint). L'idée est de réduire le nombre de paquets de contrôle en évitant le *flooding* (inondation), c'est-à-dire le fait que tous les nœuds diffusent systématiquement tous les paquets de contrôle. Dans *OLSR*, un nœud n désigne parmi ses voisins, à un saut, V_{n_1} , ses *MPR*, V_{n_m} , c'est-à-dire ceux qui se chargeront de diffuser ses paquets de contrôle. Ses autres voisins, $V_{n_1} - V_{n_m}$, reçoivent et traitent normalement ses paquets sans en rediffuser aucun. Par ailleurs, n maintient un autre sous-ensemble de ses voisins, V_{n_s} qui comprend ceux qui l'ont désigné leur *MPR*. Les éléments de V_{n_s} sont dits les *MPR Selectors* de n . Tout message en diffusion en provenance de $s \in V_{n_s}$, reçu par n est impérativement diffusé par celui-ci. V_{n_m} et V_{n_s} sont définis et évoluent au rythme des messages *Hello* émis toutes les deux secondes, *Hello Interval*.

Un message *Hello* contient, outre le paramètre *Willingness* qui indique la propension du nœud à assumer le rôle de *MPR*, a_n , l'adresse du nœud émetteur n , et $\forall v \in V_{n_1}$, a_v , l'adresse de v et l_v , le type du lien (n, v) qui prend ses valeurs dans $T_L = \{\text{symétrique, asymétrique, perdu, inconnu, voisin symétrique, MPR, ...}\}$. Tout nouveau lien de (n, v) est déclaré par n comme *asymétrique* a priori. Il devient *symétrique* dès lors que n reçoit à son tour ce même lien en provenance de v . Dès lors v est éligible en tant que *MPR* de n par n et réciproquement. V_{n_1} est ainsi constitué des v tels que (n, v) est *symétrique*. V_{n_2} , les voisins à 2 sauts de n , est constitué des nœuds v_2 tels que v_2 est voisin de $v \in V_{n_1}$ ET $v_2 \notin V_{n_1}$.

La Fig. 6.1 montre le gain obtenu par l'utilisation des *MPR* : 12 retransmissions sont suffisantes pour diffuser un message aux nœuds éloignés de 3 sauts du nœud émetteur i.e. le nœud central de la Fig. 6.1a. Sans cette notion, il faut 24 retransmission, Fig. 6.1b. Il s'agit

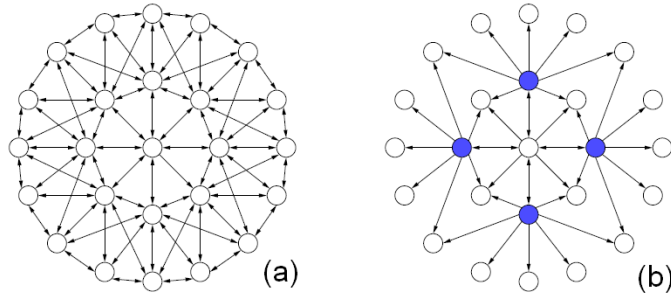


FIGURE 6.1 – Gain de retransmission 50 % : 12 au lieu de 24 : (a) sans MPR, (b) avec MPR

de la *première optimisation*. La *deuxième optimisation* vient du fait que dans un message *Topology Control, TC*, qui fait connaître la topologie du réseau périodiquement, seul le voisinage des nœuds est transmis et non l'intégralité du réseau comme dans le message *Link State Advertisement, LSA* du protocole *OSPF*. Enfin, la *troisième optimisation* vient du fait que ces messages *TC* (ainsi que les autres messages de contrôles *MID* et *HNA* qui sont en dehors de notre propos) ne sont jamais relayés par les nœuds non *MPR*.

6.4 Une approche énergétique du protocole *OLSR* : e-OLSR

Le gain relaté à la section précédente, Fig. 6.1, a bien entendu des conséquences très positives sur la consommation d'énergie du réseau mobile. Mais il est possible d'aller plus loin.

L'efficacité énergétique du routage dans un réseau mobile dépend de l'exactitude du niveau d'énergie instantané, fait remarquer Alhalimi and Kunz [2007]. Les contraintes de bande passante, de coûts de communication, de taux de perte élevés et de dynamique de la topologie des réseaux mobiles, complexifie considérablement ses collecte et mise à jour. L'erreur relative correspond à la différence moyenne entre les niveaux d'énergie perçu (par le nœud qui prend la décision du routage) et réel (connue suite au routage). Elle augmente avec le débit. Quoique le paramétrage de base du protocole *OLSR* ne permette pas de la corriger, différentes techniques sont proposées et la réduisent significativement. Notamment, la *prédiction intelligente* rend un niveau d'énergie précis quelque soit la densité du trafic [Alhalimi and Kunz 2007].

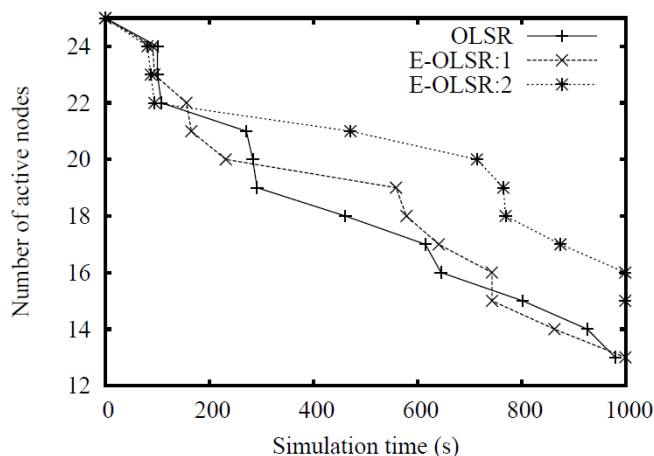


FIGURE 6.2 – 2 mécanismes d’optimisation énergétique utilisant OLSR

Parallèlement, Ghanem et al. [2005] propose 2 nouveaux mécanismes d’optimisation énergétique dans les réseaux mobiles utilisant *OLSR*. Lorsque par exemple 2 nœuds arrivent ex-æquo dans l’algorithme de choix des *MPR*, le fait de choisir celui qui a le plus d’énergie augmente l’autonomie globale du réseau mobile. Les résultats sont présentés Fig. 6.2.

Cette approche, simple et originale, nous a orientés vers une optimisation du protocole *OLSR* par une « optimisation » du choix des *MPR*.

6.5 Une approche sociale du protocole *OLSR* : OLSR-S

La pertinence de la démarche de Ghanem et al. [2005] vient de son originalité et sa simplicité de mise en œuvre pour des résultats fort intéressants. Du coup, nous nous en inspirons et considérons le cœur du protocole *OLSR*, c’est-à-dire son algorithme de choix *MPR*, comme étant la place idéale pour y introduire la prise en compte des regroupements sociaux des utilisateurs d’un réseau mobile, et créer notre *OLSR Social*.

La notion de critère de regroupement peut être mise en œuvre de différente façon. Un mécanisme de *publication-abonnement* (*publish-and-subscribe*) peut permettre à un nœud d’émettre une offre et aux autres de l’accepter pour une durée déterminée par une date, et une heure de début et une date et heure de fin. Pendant ce laps de temps, notre protocole *OLSR Social* saura s’adapter afin d’être local à ce groupe. Un autre mécanisme plus proche

de nos travaux est de considérer que le premier Point d'Attraction visité par un nœud, fait prendre à ce dernier la « couleur locale » du Point d'Attraction en ce sens où celui-ci lui attribue un identificateur de groupe qui lui est propre, cela, bien entendu pour une durée déterminée. On remarque en effet que souvent un groupe se forme à un événement et ne se défait pas de suite. Enfin, rien n'empêche d'attribuer aux nœuds un identificateur de groupe choisi arbitrairement avant de lancer les tests de simulation. L'aspect regroupement étant une spécialité en soi qui va bien au-delà de nos travaux, c'est cette dernière option que nous choisissons pour l'implémentation de notre algorithme et nos test.

La question qui se pose à nous à présent est la suivante : comment ajouter à moindre frais cette « perception » des groupes communautaires dans *OLSR*? La réponse que nous donnons est la suivante : donner aux nœuds d'un même groupe communautaire la même propension à être *MPR*. En effet, comme mentionné à la section 6.3, le paramètre *Willingness* du message *Hello* indique la propension d'un nœud à assumer le rôle d'*MPR*. Ce paramètre peut prendre une valeur comprise entre 0 (*Will_Never*) et 7 (*Will_Always*) avec comme valeur par défaut 3. Rien ne nous empêche a priori de définir une valeur particulière pour chaque groupe communautaire et l'affecter au paramètre *Willingness* de toutes ses stations.

6.6 Application

Nous ajoutons donc de nouvelles valeurs possibles pour le paramètre *Willingness* : *Will_Social_Group_a* où $a = A, B, C, D$ qui définit un identifiant de groupe communautaire. Nous n'avons besoin pour nos tests que de 4 groupes mais rien ne s'oppose à en créer d'avantage si besoin.

En outre, nous ajoutons à l'algorithme de choix *MPR* déroulé au nœud n , la condition suivante écrite en pseudo-code :

```
// Placer le code suivant dans la branche où l'on a le choix
// entre plusieurs candidats MPR et où le critère de choix
// consiste à prendre celui de plus grande Willingness

// Initialiser m, le MPR choisi, à rien
m = null

// n est-il un nœud social ?
SI n.Willingness > Will_Always ALORS

    // n est un nœud social
```


6.7. IMPLÉMENTATION

```
// Parcourir les candidats MPR à la recherche de
// celui qui aurait même Willingness que n

POUR chaque candidat MPR c FAIRE
  SI c.Willingness = n.Willingness ALORS
    m = c
    SORTIE DU POUR
  FINSI
FINPOUR
FINSI

// Dérouler l'algorithme OLSR de base
SI m = null ALORS
  m = c[0]
  POUR chaque candidat MPR c FAIRE
    SI c.Willingness > m.Willingness ALORS
      m = c
  FINSI
FINPOUR
FINSI
```

Notons que la ligne « SORTIE DU POUR » est optionnelle. Dans la version *avec* « SORTIE DU POUR », nous choisissons le premier candidat MPR qui convient. C'est le choix que nous faisons. Dans la version *sans* « SORTIE DU POUR », nous choisissons le dernier candidat MPR qui convient en abandonnant les éventuels premiers candidats qui aurait pu être retenus.

6.7 Implémentation

Nous comparons notre approche avec le protocole *OLSR* de base en utilisant dans les 2 cas notre modèle de mobilité *Ant Manhattan Mobility Model (AMN)*. Les détails d'implémentation sont donnés en annexe B.

Pour nos études de performances, nous utiliserons la statistiques suivante :

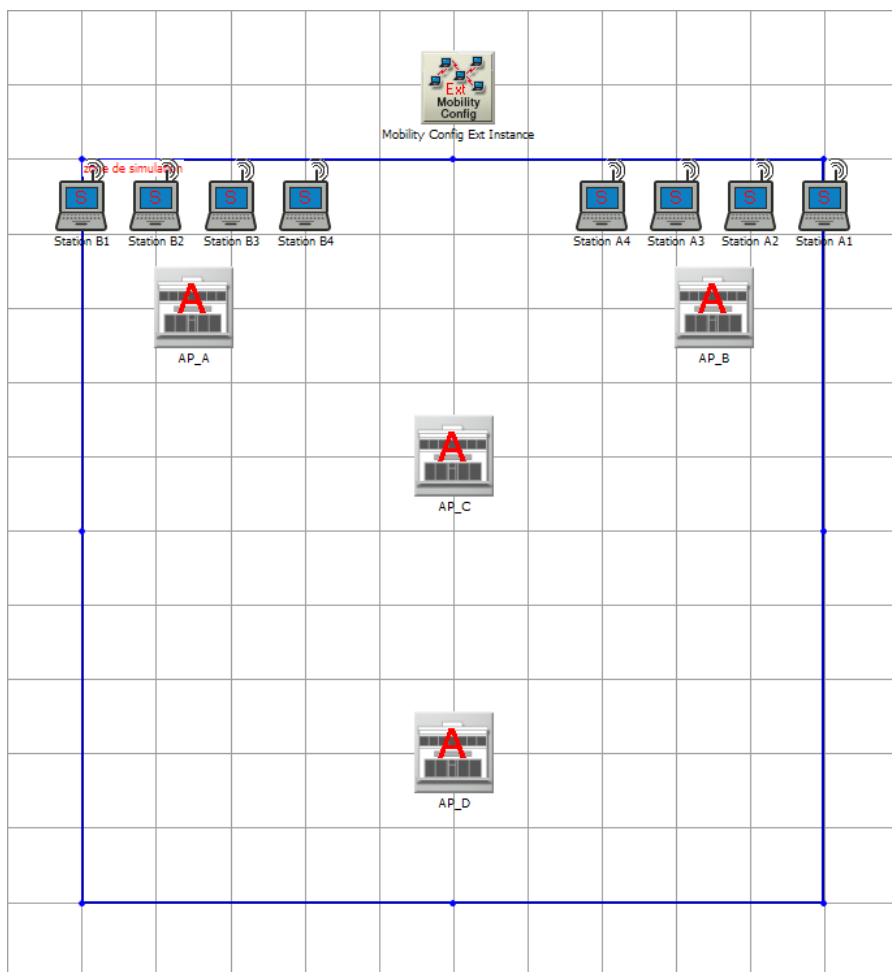
- *Node statistics, OLSR, MPR Status* qui relate l'attribut « *est MPR* » du nœud considéré.

Afin de créer notre réseau et configurer ses différents paramètres, nous y déposons une instance de notre composant *Mobility Config Ext*.

Ce réseau sera composé de :

- 4 points d'attractions : AP_a où $a = A, B, C, D$;
- 4 machines nommées "Station An" où $n = 1, 2, 3, 4$ pour le groupe A ;
- 4 machines nommées "Station Bn" où $n = 1, 2, 3, 4$ pour le groupe B.

Le réseau résultat sera :



Pour implémenter notre *OLSR-S*, nous ajoutons nos nouvelles valeurs de *Willingness* permettant à chaque nœud de reconnaître, en priorité, les nœuds de son groupes comme éventuels *MPR*.

Le code correspondant nous permet d'en établir le flux de contrôle :

- le processus s'initialise ;
- il passe en sommeil ;
- il est réveillé par l'un des événements suivants :
 - arrivée d'un paquet ;
 - échéance d'un temporisateur.

À chaque changement de voisinage¹ ou de topologie², le calcul³ de l'ensemble des *MPR* est lancé.

Nous étudions l'algorithme *OLSR* et après la détermination du lieu où *Willingness* est testé nous ajoutons notre code définissant nos nouvelles conditions. Notons la concision de l'intervention dans le code *Opnet* dont voici les lignes extraites sans aucune intervention supplémentaire :

```
/******  
/* OLSR SOCIAL --- DEBUT  
/*  
/***** Remplacer ceci :  
/*  
/* if (nbr_set_entry_ptr->willingness > willingness)  
/*  
/***** par cela :  
/*  
if (node_willingness > OLSRC_WILL_ALWAYS)  
  {  
    if (nbr_set_entry_ptr->willingness == node_willingness)  
      {  
        mpr_candidate_ptr = nbr_set_entry_ptr;  
        willingness = nbr_set_entry_ptr->willingness;  
        reachability = nbr_set_entry_ptr->reachability;  
        degree = nbr_set_entry_ptr->degree;  
      }  
  }  
}
```

1. apparition/disparition d'un nœud à 1 ou 2 sauts, ou bien nouveau lien asymétrique. Apparition \iff lien devient symétrique. Disparition \iff plus de lien du tout. Lien asymétrique \iff nœud potentiellement nœud à 1 saut.

2. création/suppression/expiration d'interfaces principales ou secondaires de nœud, consécutives à l'arrivée d'un *Topologie Control Message* ou message *TC* ou à l'échéance d'un temporisateur d'expiration de validité d'interface.

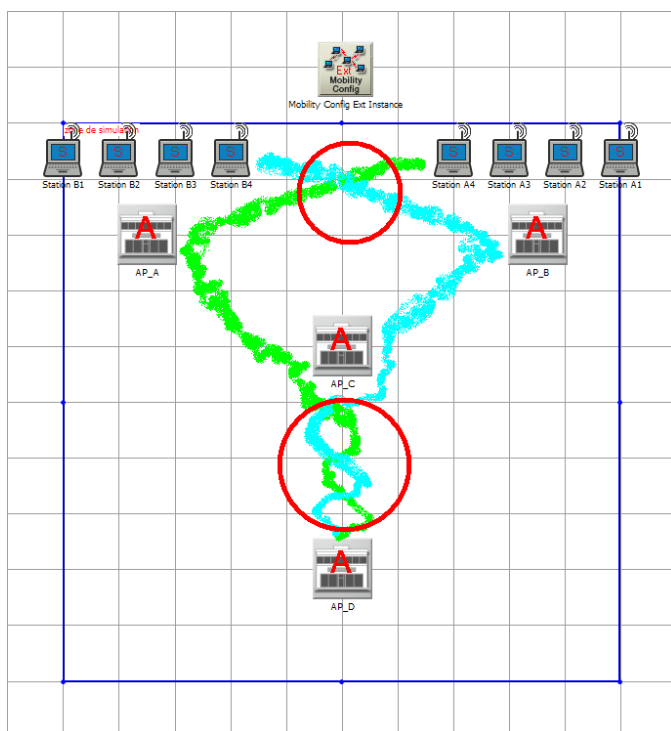
3. pour traduire au plus près du nom de fonction mais il s'agit plutôt d'une mise à jour

```

else if (nbr_set_entry_ptr->willingness > willingness)
    /*
    /* OLSR SOCIAL --- FIN
    /******
    
```

On posant que les nœuds *Station Ai* et les nœuds *Station Bi* appartiennent respectivement aux mêmes groupes, les nœuds d'un groupe vont, comme nous l'avons souligné dans les chapitres précédents, avoir les mêmes comportements et se diriger vers les mêmes endroits (i.e. les mêmes PA)

Les trajectoires des nœuds des groupes *A* et *B* sont décrites ci après :



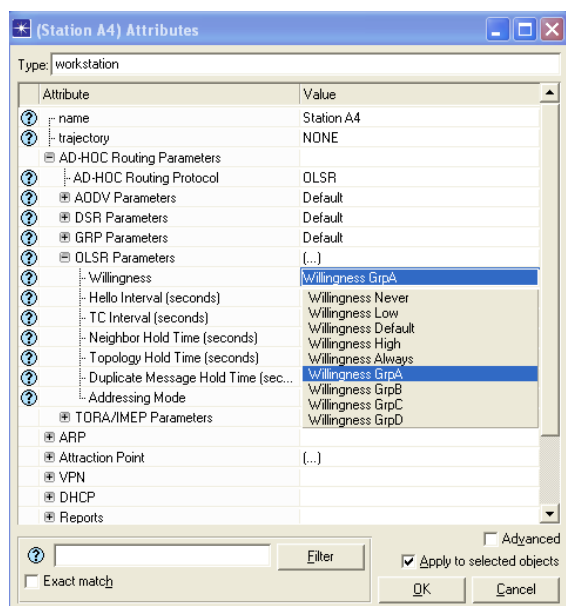
Cette configuration va nous permettre de voir l'impact de notre algorithme *OLSR-S* sur le choix des *MPR*.

Les lieux où la probabilité est grande pour un nœud de changer ses *MPR* sont représentés par 2 cercles.

Afin de forcer les groupes à suivre ces trajectoires, il nous suffit de sélectionner les Points d'Attraction à visiter, cf. annexe B.

Afin de comparer notre nouvelle approche avec l'algorithme *OLSR* de base, nous réalisons 2 scénarios, l'un avec *OLSR* de base et l'autre avec notre *OLSR-S* :

- « Without Social OLSR », laisse à *Will_default* le *Willingness* ;
- « With Social OLSR », initialise le *Willingness* à :
 - *Willingness_grpA* pour le groupe *A* :
 - *Willingness_grpB* pour le groupe *B*.



Notre choix est d'avoir des stations uniformes d'un point de vue fonctionnement (isofonctionnelles). Nous paramétrons donc le générateur de trafic uniformément sur toutes les stations.

Enfin, la statistique *OLSR Status* de *Opnet Modeler* est choisie.

OLSR Status est expliquée dans l'aide en ligne comme suit (traduction libre par nos soins) :

« Cette donnée statistique représente l'état "Est MPR" d'un nœud. Elle est à 1 lorsque le nœud est MPR et 0 dans le cas contraire. La fréquence de mise à jour de cette donnée suit celle des messages TC. C'est au moment où un nœud essaie d'émettre un message TC⁴ que cette donnée est mise à jour. D'une certaine manière, cette donnée indique également les points dans le temps où ce nœud a émis un message TC. En effet, seul les nœuds MPR

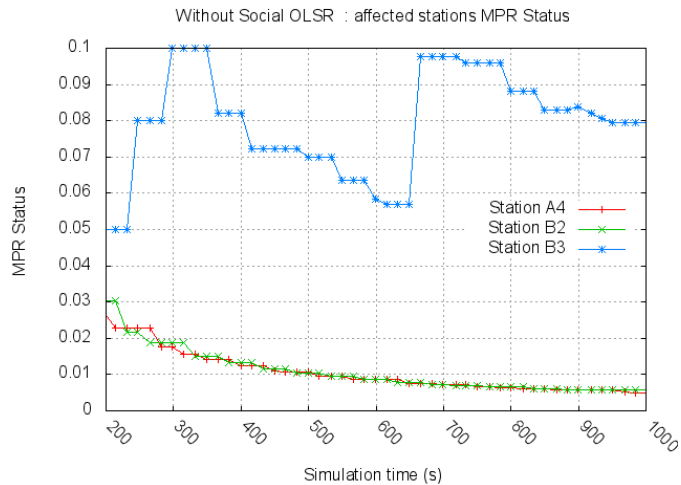
4. cela dépend donc de la taille de la liste des MPR Selectors

peuvent émettre des messages TC ».

Dans nos résultats, nous affichons la valeur moyenne de cette données statistique. Le lecteur ne s'étonnera donc pas de voir des valeurs différentes de 0 ou 1.

6.8 Résultats

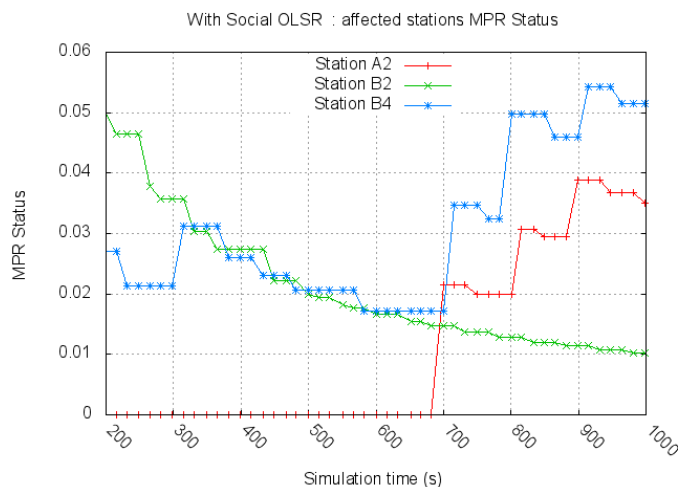
Le premier scénario n'utilise pas *OLSR Social* et donne les courbes suivantes :



ceci représente les 3 stations qui ont joué un rôle de *MPR*. Il s'agit des stations *A4*, *B2* et *B3*. *B3* est depuis le départ bien placée pour devenir l'unique *MPR*. Au bout d'environ 4 minutes, nous constatons effectivement que *A4* et *B2* s'inclinent et cèdent leurs positions de *MPR* au profit de *B3*. Le groupe *A* a donc au finish adopté comme *MPR* une station du groupe *B*.

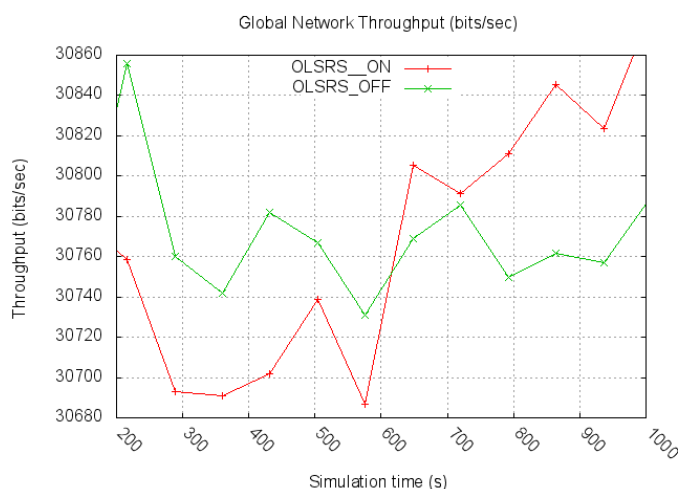
Le second scénario utilise *OLSR Social* et donne les courbes suivantes :

6.8. RÉSULTATS



ceci représente les 3 stations qui ont joué un rôle de *MPR*. Il s'agit des stations *A2*, *B2* et *B4*. *B2* est depuis le départ bien placée pour devenir l'unique *MPR*. Mais, au bout d'une dizaine de minutes, nous constatons au contraire que *B2* s'incline et cède sa position de *MPR* au profit de *A4* et non *B2*. Les stations du groupe *A* ont donc au finish opté pour un *MPR* de leur groupe. Il en va de même au sein du groupe *B*.

Nous pouvons nous poser la question de savoir qu'est-ce que cela change pour le réseau dans sa globalité d'avoir moins de changements d'*MPR*. Nous comparons alors les quantité de données reçues *Throughput* des deux scénarios, et la réponse nous est donnée par les courbes ci-dessous :



Celles-ci montrent que la quantité de données reçue globalement, en moyenne, sur l'ensemble des stations du réseau mobile est de plus en plus élevée lorsqu'on utilise *OLSR*

Social que lorsqu'on ne le met pas en œuvre où elle stagne.

6.9 Conclusion

Dans ce chapitre, nous avons présenté, une autre facette de l'utilisation des comportements communautaires dans les réseaux mobile : l'intégration d'une faculté de « perception » des aspects sociaux aux cœurs des protocoles de routage pour les réseaux mobiles.

Nous avons d'abord donné un bref aperçu de ce qui se fait en la matière à l'heure actuelle.

Ensuite, nous avons proposé un protocole de routage original basé sur le protocole *Optimized Link State Routing* auquel nous adjoignons la composante sociale de perception des regroupements des individus.

Nous avons assorti notre proposition de tests en milieu simulé qui montrent qu'effectivement la stabilité du réseau mobile est plus pérenne lorsque cette perception existe qu'en son absence.

Toutefois, un inconvénient doit être mentionné et il s'agit de la gestion de l'énergie des *MPR* sociaux qui, étant plus stables, risquent fort de consommer plus d'énergie et de devenir sur ce plan, le maillon faible de l'ensemble. D'où l'intérêt d'une solution hybride *e-OSLR + OSLR-S*. Mais, cela est en dehors du cadre de cette thèse.

6.9. CONCLUSION

Chapitre 7

Publications

Nos publications durant ce travail de thèse sont les suivantes :

- L. Harfouche, H. Costantini and S. Boumerdassi. *Social Mobility Models using Ant Colony Systems*. Proceedings of the 22nd IEEE Personal Indoor Mobile Radio Communications (PIMRC'11 - WNHC). SEP. 2011.
- L. Harfouche, S. Boumerdassi and E. Renault. *Weighted Social Manhattan : Modeling and performance analysis of a mobility model*. Proceedings of the PIMRC'10. pp. 2019-2023. SEP. 2010.
- L. Harfouche, S. Boumerdassi and E. Renault. *Towards a Social Mobility Model*. Proceedings of the 20th International Symposium on Personal, Indoor and Mobile Radio Communications. pp. 2876–2880. IEEE Publisher. SEP. 2009.
- L. Harfouche and Selma Boumerdassi. *Towards an Environment Aware Formalization for Mobility Modelling in MANET*. The 2009 International Conference on Wireless Networks (ICWN'09). Las Vegas.
- L. Harfouche and S. Boumerdassi. *Les modèles de mobilité*. Rapport de recherche interne. CEDRIC. 2007. (ref. CEDRIC 1289)

Chapitre 8

Conclusions et Perspectives

8.1 Conclusion

Notre thèse présente 3 volets innovants. Premier volet, elle propose 3 nouveaux modèles de mobilité sociaux. Second volet, elle montre leurs apports par rapport aux modèles existants, en soulignant le fait qu'en se rapprochant des comportements sociaux au travers de l'utilisation de la notion de Points d'attraction, ces modèles changent considérablement les résultats des tests de performances des protocoles de routage des réseaux mobiles. Troisième volet, elle propose une orientation sociale au protocole *OLSR*.

Dans le chapitre 2, nous avons introduit les modèles de mobilité et la nécessité de leur utilisation dans le domaine des réseaux mobiles. Malgré l'existence des nombreuses catégories de modèles de mobilité décrites, deux seules ont perduré : les modèles synthétiques et les traces réelles. Toutefois, la possession de traces réelles est un atout majeur pour celui qui en dispose. Celles-ci ont une réelle valeur marchande. On connaît les récentes tribulations d'Apple Corporation, concernant la capitalisation des traces de tous ses utilisateurs après que ceux-là ont accepté de leur propre chef d'activer la géolocalisation. C'est la raison pour laquelle la plupart des recherches et la nôtre également portent sur les modèles de mobilité synthétiques qui sont bien éloignés des enjeux financiers des traces. Enfin, nous avons proposé une nouvelle classification que nous avons qualifiée de sociale dans la mesure où elle tient compte des aspects sociaux introduits dans les modèles. Cette nouvelle classification nous a permis de mettre en perspective les modèles actuels sous un éclairage social et constater leur grande pauvreté en la matière. Cette classification facilite l'élaboration de

nouveaux modèles sociaux car elle précise quelle partie du modèle est concernée par quelle caractéristique sociale. Bien entendu, nous nous basons sur cette nouvelle classification pour définir nos propres modèles.

Dans le chapitre 3, nous avons présenté l'objet de notre recherche, à savoir, l'importance de la prise en compte de la composante sociale dans l'étude et la modélisation des réseaux sans fil parce qu'en tout premier lieu, les appareils mobiles sont portés par des êtres humains. Afin d'asseoir cette thèse qui est la nôtre, nous avons élaboré notre premier modèle de mobilité social, *Social Manhattan (SMN)*¹ en basant sur la notion de points d'attraction. Le résultat de nos tests ont de suite corroboré notre thèse : la prise en compte des aspects sociaux dans les réseaux mobiles a une réelle influence sur les résultats des études de performances de ceux-ci. En effet, nous avons vu que les résultats obtenus par le commun des testeurs en matière de protocole réseau par l'utilisation du modèle de mobilité le plus courant, à savoir *Manhattan (MAN)*, étaient éloignés de la réalité. Plus précisément, nos tests ont mis en évidence de façon globale, d'une part, le fait que *MAN* avait clairement été surestimé du point de vue d'une variable statistique réseau et que, d'autre part, des paramètres aussi divers que le débit des données (indice primordial pour le multimédia en plein essor), la vitesse de déplacement des nœuds (paramètre crucial dans la mobilité qui est LE problème du moment), la densité du réseau (qui préoccupe tous ceux qui étudient de près les traces en tout genre afin d'éviter par exemple les goulots d'étranglement lors des événements et manifestations, ou lors des mises à jour logiciel...) mais aussi le choix entre protocole réactif ou proactif, la présence d'éléments asociaux dans le réseau ou encore le nombre de points d'attraction influaient le plus souvent fortement sur les performances relevées dans notre laboratoire, à l'aide de *SMN*.

Dans le chapitre 4, nous avons pensé qu'en faisant évoluer l'attractivité de nos points d'attraction qui était figée dans *SMN*, nous nous approcherions sans doute un peu plus des comportements réels des utilisateurs. C'est de cette idée qu'est né *Weighted Manhattan (WMN)*, notre deuxième modèle de mobilité social. Le résultat de nos tests ont montré qu'effectivement une variable statistique réseau (*Throughput*) était dans ce cas plus

1. nommé ainsi parce que nous l'avons implémenté en tant que prolongement de *Manhattan* mais nous aurions pu choisir un tout autre point de départ. Seulement, Manhattan étant le plus utilisé, nous n'avons pas souhaité dénoter.

fortement influencée par la quasi-totalité des paramètres que dans *SMN* et ce de façon absolument imprévisible.

Dans le chapitre 5, nous avons souhaité ajouter un mécanisme qui puisse rendre compte du fait qu'un lieu perd de sa notoriété avec la raréfaction de ses visiteurs. La modélisation d'un tel comportement nous a été possible en utilisant les Systèmes de Colonies de Fourmis (*Ant Colony System, ACS*) et leur mécanisme de phéromone. Le résultat de nos tests nous ont permis de montrer que, par exemple, le choix entre protocole réactif ou proactif avait une influence plus importante que dans *SMN* et *WMN*, à nouveau, de façon insoupçonnée.

Nous avons présenté dans le chapitre 6, une autre facette de l'utilisation des comportements communautaires dans les réseaux mobile : l'intégration d'une faculté de « perception » des aspects sociaux au cœur des protocoles de routage pour les réseaux mobiles. Nous avons donné un bref aperçu de ce qui se fait en la matière à l'heure actuelle. Nous avons proposé un protocole de routage original basé sur le protocole *Optimized Link State Routing* auquel nous avons adjoint la composante sociale de perception des regroupements des individus. Nous avons assorti notre proposition de test en milieu simulé qui montre qu'effectivement la stabilité du réseau mobile est plus pérenne lorsque cette perception existe qu'en son absence.

L'implémentation des modèles de mobilité a donné lieu à une charge de développement logiciel en Opnet Proto-C très importante. Une partie de ce travail a été décrit en annexe.

8.2 Perspectives

Du point de vue recherche, ce travail a ouvert des voies qui sont au nombre des indices tangibles que nos tests ont mis en exergue.

Tout d'abord du point de vue des modèles de mobilité sociaux : la preuve est faite qu'il y a là de quoi améliorer l'évaluation des protocoles des réseaux sans fil. Nous avons certes axé nos travaux sur les points d'attraction. Cependant, d'autres démarches sont en cours : les traces bien sûr que nous avons déjà citées et qui sont très prometteuses pour réduire les goulots d'étranglement lors des meetings ou mises à jour logiciel ; les réseaux communautaires et tous ce qu'ils renferment d'éléments qui émeuvent chaque jour

le néophyte comme le scientifique des sciences humaines et sociales ; pour ne citer que 2 exemples.

Enfin, l'idée d'introduire une perception du groupement social au cœur d'un protocole réseau (*OLSR* dans notre cas) a été une piste féconde. Mais qu'en est-il de la consommation d'énergie ? En effet, les *MPR* sociaux étant plus stables, risquent fort de consommer plus d'énergie et de devenir sur ce plan, le maillon faible de l'ensemble. D'où l'intérêt d'une solution hybride e-OSLR + OSLR-S ! Mais, cela est en dehors du cadre de cette thèse.

Bibliographie

Bibliographie

- Paramics, 2011. URL <http://www.paramics-online.com/>. 14
- Udel, 2011. URL <http://udelmodels.eecis.udel.edu/>. 12
- Corsim, 2011. URL <http://mctrans.ce.ufl.edu/featured/tsis/Version5/corsim.htm>. 14
- transims, 2011. URL http://tmip.fhwa.dot.gov/community/user_groups/transims. 14
- vissim, 2011. URL <http://www.vissim.com/>. 14
- M. Abolhasan, B. Hagelstein, and J.C.-P. Wang. Real-world performance of current proactive multi-hop mesh protocols. In *Communications, 2009. APCC 2009. 15th Asia-Pacific Conference on*, pages 44–47, 2009. doi : 10.1109/APCC.2009.5375690. 70
- Rana Alhalimi and Thomas Kunz. Energy level accuracy in mobile ad-hoc networks using olsr. In *Proceedings of the 3rd international conference on Wireless internet, WICON '07*, pages 14 :1–14 :9, ICST, Brussels, Belgium, Belgium, 2007. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering). ISBN 978-963-9799-12-7. URL <http://portal.acm.org/citation.cfm?id=1460047.1460065>. 95
- K. Alzoubi and M. Ayyash. Maintaining a quality of service routing tree for mobile ad hoc networks. In *Proceedings of the 2006 international conference on Wireless communications and mobile computing, IWCMC '06*, pages 127–134, New York, NY, USA, 2006. ACM. ISBN 1-59593-306-9. doi : <http://doi.acm.org/10.1145/1143549.1143577>. URL <http://doi.acm.org/10.1145/1143549.1143577>. 45
- J. An, Y. Ko, and D. Lee. A social relation aware routing protocol for mobile ad hoc networks. In *Pervasive Computing and Communications, 2009. PerCom 2009. IEEE*

- International Conference on*, pages 1–6, march 2009. doi : 10.1109/PERCOM.2009.4912879. 93
- Anonymous. Foo plot, oct 2007. URL <http://fooplot.com/>. 50
- F. Bai and A. Helmy. Impact of mobility on last encounter routing protocols. In *in Proceedings of the Fourth Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, 2007. 6
- T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. *Wireless Communication and Mobile Computing*, 2(5) :483–502, 2002. 6, 8, 12
- Kuo-Hsing Chiang and N. Shenoy. A random walk mobility model for location management in wireless networks. In *Personal, Indoor and Mobile Radio Communications, 2001 12th IEEE International Symposium on*, volume 2, pages E-43–E-48 vol.2, sep/oct 2001. doi : 10.1109/PIMRC.2001.965260. 9
- Kuo Hsing Chiang and Nirmala Shenoy. A 2-d random-walk mobility model for location-management studies in wireless networks. *IEEE Transactions on Vehicular Technology*, 53 :413–424, 2004. doi : 10.1109/TVT.2004.823544. 9
- P. Costa, C. Mascolo, M. Musolesi, and G. P. Picco. Socially-aware routing for publish-subscribe in delay-tolerant mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 26(5) :748–760, June 2008. ISSN 0733-8716. doi : 10.1109/JSAC.2008.080602. URL <http://dx.doi.org/10.1109/JSAC.2008.080602>. 92, 93
- Elizabeth M. Daly and Mads Haahr. Social network analysis for routing in disconnected delay-tolerant manets. In *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing, MobiHoc '07*, pages 32–40, New York, NY, USA, 2007a. ACM. ISBN 978-1-59593-684-4. doi : <http://doi.acm.org/10.1145/1288107.1288113>. URL <http://doi.acm.org/10.1145/1288107.1288113>. 92
- E.M. Daly and M. Haahr. Social network analysis for routing in disconnected delay-tolerant manets. In *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing, MobiHoc '07*, pages 32–40, New York, NY, USA, 2007b.

-
- ACM. ISBN 978-1-59593-684-4. doi : <http://doi.acm.org/10.1145/1288107.1288113>.
URL <http://doi.acm.org/10.1145/1288107.1288113>. 92
- B. Divecha, A. Abraham, C. Grosan, and S. Sanyal. Impact of node mobility on manet routing protocols models. 26
- M. Dorigo. *Optimization, Learning and Natural Algorithms*. PhD thesis, Politecnico di Milano, Italy, 1992. 69, 70
- M. Dorigo and L.M. Gambardella. Ant colony system : A cooperative learning approach to the traveling salesman problem. *IEEE Transactions On Evolutionary Computation*, 1(1) :53–66, apr 1997. 69, 71
- G. Eason, B. Noble, and I. N. Sneddon. On certain integrals of lipschitz-hankel type involving products of bessel functions. *Philosophical Transactions of the Royal Society of London*, 247(935) :529–551, april 1955. doi : 10.1098/rsta.1955.0005. 18
- M. Fiore, J. Haerri, F. Filali, and C. Bonnet. Understanding vehicular mobility in network simulation. In *IEEE Internatonal Conference on Mobile Adhoc and Sensor Systems*, MASS, oct 2007. 6
- Nabil Ghanem, Selma Boumerdassi, and Éric Renault. New energy saving mechanisms for mobile ad-hoc networks using olsr. In *Proceedings of the 2nd ACM international workshop on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks*, PE-WASUN '05, pages 273–274, New York, NY, USA, 2005. ACM. ISBN 1-59593-182-1. doi : <http://doi.acm.org/10.1145/1089803.1090006>. URL <http://doi.acm.org/10.1145/1089803.1090006>. 96
- R. Ghosh, A. Das, P. Venkateswaran, S.K. Sanyal, and R. Nandi. An efficient social network-mobility model for manets. In Ajit Pal, Ajay Kshemkalyani, Rajeev Kumar, and Arobinda Gupta, editors, *7th International Workshop on Distributed Computing*, volume 3741 of *IWDC*, pages 349–354. Springer Berlin / Heidelberg, 2005. URL http://dx.doi.org/10.1007/11603771_38. 17
- M.C. Gonzalez, C.A. Hidalgo, and A.-L. Barabasi. Understanding individual human

- mobility patterns. *Nature*, 458(7235) :238–238, Mar 2009. ISSN 0028-0836. doi : 10.1038/nature07850. URL <http://dx.doi.org/10.1038/nature07850>. 12
- Network Working Group. Optimized link state routing protocol (olsr), oct 2003. 28
- M. Günes, U. Sorges, and I. Bouazizi. Ara - the ant-colony based routing algorithm for manets. In *International Conference on Parallel Processing Workshops, ICPPW*, page 79, Washington, DC, USA, 2002. IEEE Computer Society. ISBN 0-7695-1680-7. doi : <http://dx.doi.org/10.1109/ICPPW.2002.1039715>. 69, 70
- J. Haerri, F. Filali, and C. Bonnet. A framework for mobility models generation and its application to inter-vehicular networks. In *3rd IEEE International Workshop on Mobility Management and Wireless Access, MobiWac*, jun 2005. doi : 10.1109/WIRLES.2005.1549382. 6, 17
- L. Harfouche, S. Boumerdassi, and É. Renault. Towards a social mobility model. In *20th International Symposium on Personal, Indoor and Mobile Radio Communications*, pages 2876–2880. IEEE Publisher, sep 2009. doi : 10.1109/PIMRC.2009.5450050. 23, 24, 74
- L. Harfouche, S. Boumerdassi, and É. Renault. Weighted social manhattan : Modeling and performance analysis of a mobility model. In *PIMRC'10*, pages 2019–2023, sep 2010. 48, 51, 52, 74
- P.S. Heck and S. Ghosh. A study of synthetic creativity : Behavior modeling and simulation of an ant colony. *IEEE Intelligent Systems*, 15(6) :58–66, 2000. ISSN 1541-1672. doi : <http://dx.doi.org/10.1109/5254.895863>. 13, 17, 70
- I. S. Jacobs and C. P. Bean. Fine particles, thin films and exchange anisotropy, 1963. 18
- A.P. Jardosh, E.M. Belding-Royer, K.C. Almeroth, and S. Suri. Real-world environment models for mobile network evaluation. *Selected Areas in Communications, IEEE Journal on*, 23(3) :622 – 632, 2005. ISSN 0733-8716. doi : 10.1109/JSAC.2004.842561. 17, 20
- G. Jayakumar and G. Gopinath. Performance comparison of manet protocol based on manhattan grid model. *Journal of Mobile Communications*, 2.1 :18–26, 2008. doi :

- <http://dx.doi.org/10.3923/jmcomm.2008.18.26>. URL <http://www.medwelljournals.com/abstract/?doi=jmcomm.2008.18.26>. 22, 23, 26
- H.-C. Liao, Y.-W. Ting, S.-H. Yen, and C.-C. Yang. Ant mobility model platform for network simulator. In *International Conference on Information Technology*, volume 2 of *ITCC*, page 380, Washington, DC, USA, 2004. IEEE Computer Society. ISBN 0-7695-2108-8. 13, 70
- Anders Lindgren, Avri Doria, and Olov Schelén. Probabilistic routing in intermittently connected networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 7 :19–20, July 2003. ISSN 1559-1662. doi : <http://doi.acm.org/10.1145/961268.961272>. URL <http://doi.acm.org/10.1145/961268.961272>. 92
- J. C. Maxwell. *A treatise on electricity and magnetism*. Clarendon Press, Oxford, 1873. 18
- S. M. Mousavi, H. R. Rabiee, M. Moshref, and A. Dabirmoghaddam. Mobisim : A framework for simulation of mobility models in mobile ad-hoc networks. *Wireless and Mobile Computing, Networking and Communication, IEEE International Conference on*, 0 :82, 2007. doi : <http://doi.ieeecomputersociety.org/10.1109/WIMOB.2007.52>. 5
- M. Musolesi and C. Mascolo. Designing mobility models based on social network theory. *SIGMOBILE Mob. Comput. Commun. Rev.*, 11 :59–70, July 2007. ISSN 1559-1662. doi : <http://doi.acm.org/10.1145/1317425.1317433>. URL <http://doi.acm.org/10.1145/1317425.1317433>. 45
- M. Musolesi and C. Mascolo. Mobility models for systems evaluation. *Middleware for Network Eccentric and Mobile Applications*, 3 :43–62, 2009. doi : 10.1007/978-3-540-89707-1_3. URL http://dx.doi.org/10.1007/978-3-540-89707-1_3. 17
- A. Myer. New stanford computing lab imagines the mobile-social future, 2011. URL <http://news.stanford.edu/news/2011/may/mobisocial-050911.html>. 17
- M. E. J. Newman. Analysis of weighted networks. *Phys. Rev. E*, 70(5) :056131, Nov 2004. doi : 10.1103/PhysRevE.70.056131. 44

Open-Mesh.net. B.a.t.m.a.n. (better approach to mobile ad-hoc networking). Online. Available : <http://www.open-mesh.net/>. 70

C. Perkins, E. Royer, and S. Das. Rfc 3561 ad hoc on-demand distance vector (aodv) routing. Technical report, IETF, 2003. URL <http://tools.ietf.org/html/rfc3561>. 28

Charles E. Perkins and Pravin Bhagwat. Highly dynamic destination-sequenced distance-vector routing (dsv) for mobile computers. In *Proceedings of the conference on Communications architectures, protocols and applications*, SIGCOMM '94, pages 234–244, New York, NY, USA, 1994. ACM. ISBN 0-89791-682-4. doi : <http://doi.acm.org/10.1145/190314.190336>. URL <http://doi.acm.org/10.1145/190314.190336>. 94

Eiko Yoneki, Pan Hui, and Jon Crowcroft. Bio-inspired computing and communication. chapter Wireless Epidemic Spread in Dynamic Human Networks, pages 116–132. Springer-Verlag, Berlin, Heidelberg, 2008. ISBN 978-3-540-92190-5. doi : http://dx.doi.org/10.1007/978-3-540-92191-2_11. URL http://dx.doi.org/10.1007/978-3-540-92191-2_11. 92

Qunwei Zheng, Xiaoyan Hong, and Jun Liu. An agenda based mobility model²¹. In *Proceedings of the 39th annual Symposium on Simulation*, ANSS '06, pages 188–195, Washington, DC, USA, 2006. IEEE Computer Society. ISBN 0-7695-2559-8. doi : <http://dx.doi.org/10.1109/ANSS.2006.11>. URL <http://dx.doi.org/10.1109/ANSS.2006.11>. 12

Annexes

Annexe A

Choix relatifs à la simulation et implémentation de nos modèles

A.1 Introduction

Afin d'évaluer nos modèles, nous devons choisir :

1. un cadre de travail parmi ceux existant en matière de mobilité ;
2. un outil de simulation.

Le cadre de travail que nous choisissons est celui du modèle de mobilité *Manhattan* car de très loin le plus répandu dans la littérature.

L'outil de simulation que nous choisissons est *Opnet Modeler* pour les raisons exposées à la section suivante qui présente cet outil.

Ensuite, nous relatons notre étude de l'existant en matière de modèle de mobilité, i.e. ce qui existe dans la version d'*Opnet Modeler* destinée aux académies.

Puis, nos premiers développements du modèle de mobilité *Manhattan* sont exposés.

Vient ensuite le problème des intersections manquées qui représente le verrou technique majeur de cette thèse. La cause de ce problème puis la solution choisie pour lever ce verrou qui littéralement débloque l'ensemble de notre travail, font l'objet de sections séparées.

Les sections suivantes sont ensuite consacrées au mode de développement d'un objet nouveau dans la palette d'objets de l'outil. La découverte de ce mode de développement représente en effet un investissement en temps considérable. Celui-ci trouve toute sa jus-

tification dans la simplification drastique que ces nouveaux objets visuels entraînent dans l'implémentation de nos modèles qui comprennent parfois plus d'une soixantaine d'instance de ses nouveaux objets !

L'implémentation proprement dite de nos modèles fera l'objet d'une section dédiée dans les chapitres correspondants.

Enfin et surtout nos résultats de simulation sont donnés en dernière section desdits chapitres. Ceux-ci sont le but de l'ensemble du processus de développement : démontrer l'intérêt de nos modèles !

L'implémentation des modèles des chapitres suivants s'appuie, répétons-le, sur les sections qui vont suivre immédiatement. Le lecteur ne s'étonnera donc pas de la longueur de l'exposé compte tenu du fait qu'il sert de base à l'ensemble de l'implémentation des outils de simulations utilisés tout au long de ce travail de thèse.

A.2 Présentation d'*Opnet Modeler*

Au Centre de recherche en informatique et communication, au laboratoire *CEDRIC* du Conservatoire National des Arts et Métiers de Paris, *CNAM de Paris*, nous disposons de 14 licences d'utilisation du produit commercial *Opnet Modeler* (Fig. A.1). La version dont nous disposons pour ce travail de thèse est donnée en Fig. A.2. Les mentions légales concernant les académies s'affichent au lancement de l'outil et sont reproduites en Fig. A.3.

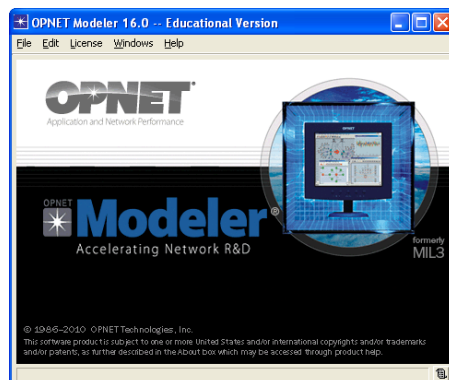


FIGURE A.1 – OPNET Modeler

Il s'agit d'un outil de simulation qui, par rapport à *Network Simulator 2* qui est le

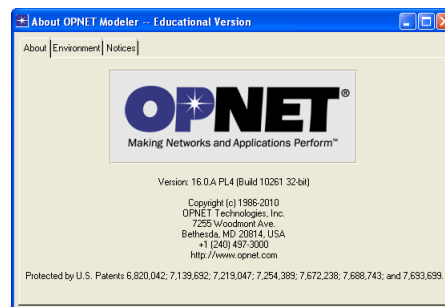


FIGURE A.2 – Notre version d’OPNET Modeler

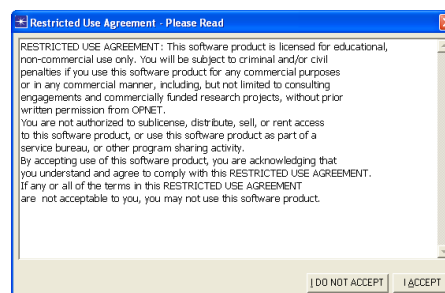


FIGURE A.3 – Mentions légales OPNET Modeler à destination des Académies

simulateur le plus utilisé dans les laboratoires de recherche, dispose d’au minimum 2 points forts :

- une interface homme-machine graphique évoluée orientée objet ;
- des objets ou nœuds de simulation qui suivent de près les besoins du marché.

L’interface homme-machine graphique évoluée orientée objet permet à l’utilisateur de décrire un scénario de simulation en choisissant les éléments du réseau simulé, dans une palette d’objets, Fig. A.4A. Ceux-ci sont déposés sur un espace représentant le lieu sur lequel s’étend le réseau simulé. Ensuite, chacun de ses objets instanciés peut faire l’objet d’un paramétrage spécifique via un éditeur d’attributs, Fig. A.4B.

Ces objets de simulation suivent de près les besoins du marché, comme en témoigne la présence de :

- La norme 802.11g implémentée au milieu de la première décennie 2000, en même temps que son adoption dans l’entreprise, Fig. A.4C ;
- La norme 802.11n MIMO implémentée fin de cette même décennie via la possibilité

A.3. ÉTUDE DE L'EXISTANT

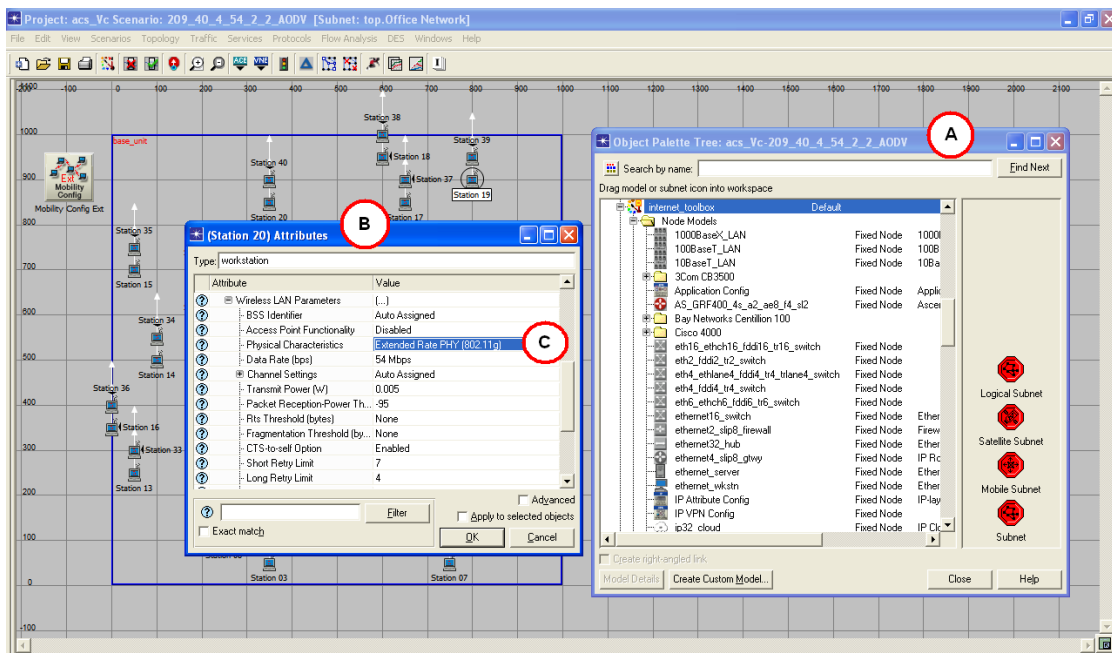


FIGURE A.4 – L'Interface Homme-Machine (IHM) d'OPNET Modeler

de définir le nombre d'antennes en émission comme en réception dont la simulation consiste en un accroissement sensible du rapport signal sur bruit (SNR) ;

- L'évolution à long terme ou LTE qui concerne tant la Super 3G que l'actuelle 4G en cours de normalisation.

En conséquence, *Opnet Modeler* est à notre connaissance le plus à jour à ce niveau.

Suite aux différentes présentations de la société *Opnet France* dans les locaux du *CNAM*, nous avons investi de nombreux mois de notre recherche à acquérir une solide expérience du *Modeler* que nous maîtrisons pour les aspects *802.11 WiFi* dont nous dispensons depuis plusieurs années des cours de formation dans le cadre des cursus Ingénieur et Mastère de la branche Réseaux du *CNAM Paris*.

A.3 Étude de l'existant

Sauf erreur ou omission de notre part, depuis le début de nos travaux de recherche dans le cadre de la présente thèse et jusqu'à la rédaction de celle-ci, l'unique modèle de mobilité présent dans la version mise à disposition par la société *Opnet* dans son *Modeler*

à destination des académies est : *Random Waypoint*.


Cette section se propose d'étudier cet unique « existant ».

Un projet d'exemple va nous y aider :

1. lancer « Opnet Modeler » ;
2. cliquer « File | Open... » ;
3. constater l'affichage de la boîte de dialogue standard « Open » ;
4. sélectionner le type de fichier « Project File (*.prj) » ;
5. se déplacer jusqu'au dossier suivant :

D:\Program Files\OPNET\16.0.A\models\std\example_networks\

où D représente la lettre du lecteur sur lequel est installé *Opnet Modeler* ;

6. ouvrir le dossier « Random_Mobility.project » ;
7. double-cliquer le fichier « Random_Mobility » qui a pour extension « .prj » ;
8. constater l'ouverture du projet d'exemple, Fig. A.5 ;
9. constater la présence de l'objet Mobility Config  : c'est lui qui renferme le paramétrage du (ou des) modèle(s) de mobilité utilisé(s) dans le scénario courant nommé « Baseline » ;
10. cliquer droit sur cet objet et dans le menu contextuel qui s'affiche sélectionner « Edit Attributes » ;
11. constater l'affichage de l'éditeur d'attributs de l'objet, voir Fig. A.6 ;

Décrivons les attributs de l'objet « Mobility Config » qui apparaissent dans la Fig. A.6, du premier au dernier :

- name = Mobility Profile Definition : nom donné à l'instance ;
- Mobility Modeling Status = Enabled : les modèles de mobilité de l'instance sont actifs ;
- Random Mobility Profiles : ensemble de modèles de mobilité :
 - Number of Rows = 1 : nombre de modèles de mobilité ;

A.3. ÉTUDE DE L'EXISTANT

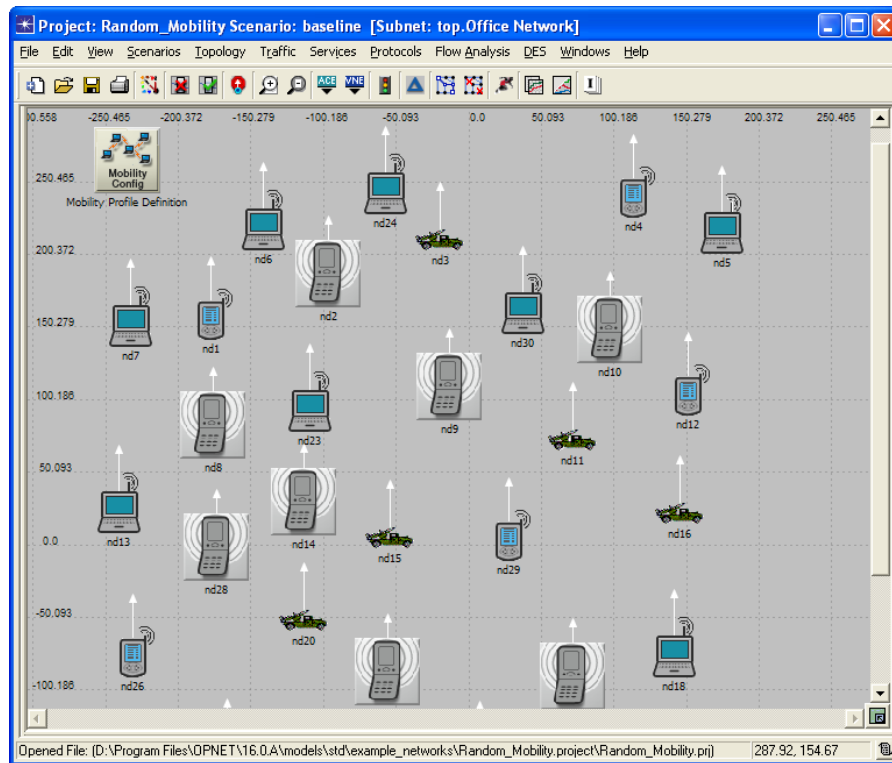


FIGURE A.5 – Projet Exemple Random_mobility d’Opnet Modeler

- Wanderer : nom du modèle de mobilité n° 1
- Profile Name = Wanderer : idem
- Mobility Model = Random Waypoint : type du modèle de mobilité
- Random Waypoint Parameters :

Attribut	Fonction
Mobility Domain Model	aire de déplacement : déf. visuelle
x_min, y_min, x_max, y_max	aire de déplacement : déf. non visuelle
Speed	vitesse des nœuds en m/s
Pause Time	temps de pause entre 2 déplacements
Start Time	délai en s entre début simu et déplac.
Stop Time	durée de l'ensemble des déplacements
Animation Update Frequency	période animation
Record Trajectory	enregistrement trajectoires des nœuds

Le seul modèle de mobilité existant est exclusivement « Random Waypoint » comme en témoigne la liste de choix ouverte qui ne contient que cet élément.

L'Éditeur d'attributs comporte une case à cocher intitulée « Advanced ». Celle-ci en la

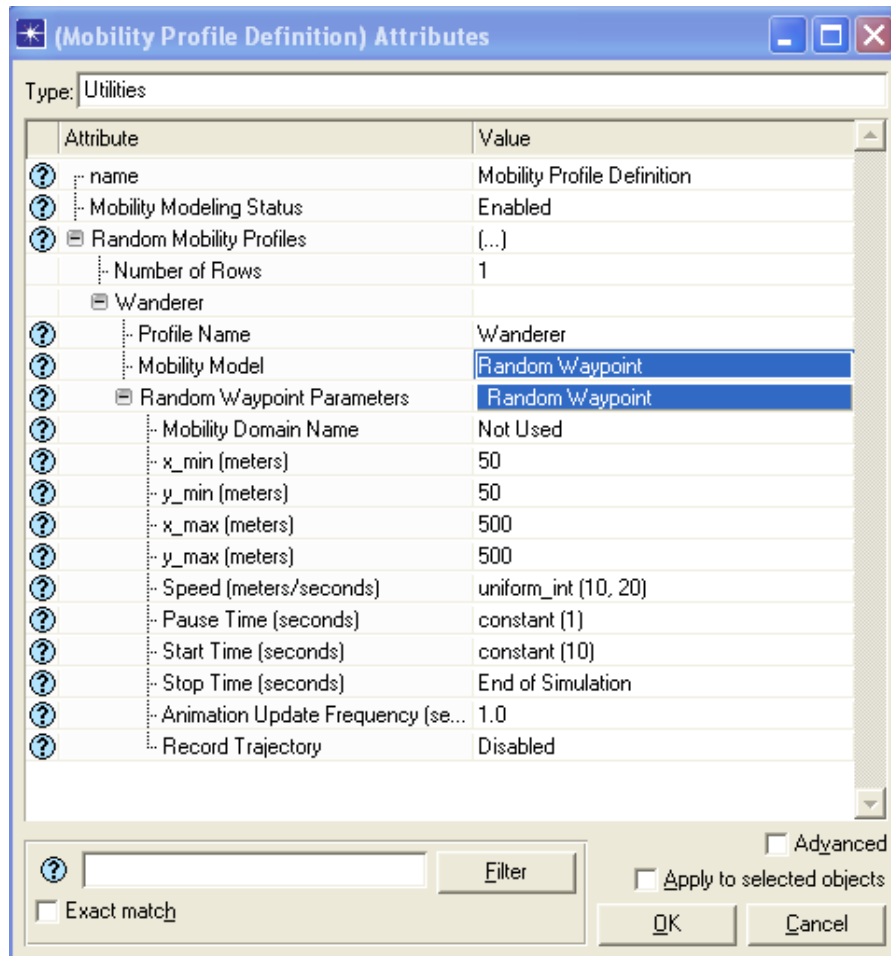


FIGURE A.6 – Editeur d'attributs de l'objet Mobility Config

cochant permet d'afficher des attributs supplémentaires, voir Fig. A.7.

Notamment, l'attribut « model » nous permet d'accéder aux organes internes de l'objet. En cliquant sur ce dernier, nous recevons la mise en garde de la Fig. A.7, qui stipule qu'il est possible de créer un modèle de sauvegarde du nœud considéré via *Device Creator* afin de pouvoir importer ses données en cas de changement du modèle d'objet. Recherches faites dans l'aide en ligne, les seuls objets concernés par cette fonctionnalité sont donnés ci-dessous :

Groupe	Objet
Suite page suivante	

Groupe Suite ...	Objet
Bridge	Ethernet (10BaseT, 100BaseT, 1000BaseX) FDDI Token Ring
Cloud	ATM Frame Relay SLIP Multiprotocol
Fibre	Channel Devices Hub Switch Endnode Multihomed Endnode
Firewall	ATM Ethernet (10BaseT, 100BaseT, 1000BaseX), Ethernet LANE, and EtherChannel FDDI Frame Relay SLIP Token Ring and TR LANE
Hub	Ethernet (10BaseT, 100BaseT, 1000BaseX) FDDI Fibre Channel Token Ring
LAN Model	Shared Switched
Mainframes	ATM Ethernet (10BaseT, 100BaseT, 1000BaseX), Ethernet LANE, and EtherChannel
Suite page suivante	

ANNEXE A. CHOIX RELATIFS À LA SIMULATION ET IMPLÉMENTATION DE
NOS MODÈLES

Groupe Suite ...	Objet
	FDDI Frame Relay Fibre Channel SLIP Token Ring and TR LANE Wireless LAN
Multihomed	Host Client Server
Multi-Service Switch	Circuit Switched Ethernet (10BaseT, 100BaseT, 1000BaseX) and EtherChannel FDDI Frame Relay SLIP Token Ring Voice over ATM
Router	ATM Ethernet (10BaseT, 100BaseT, 1000BaseX), Ethernet LANE, and EtherChannel FDDI Frame Relay SLIP Token Ring and TR LANE Wireless LAN
Switch	ATM Ethernet (10BaseT, 100BaseT, 1000BaseX) and EtherChannel FDDI Fibre Channel Frame Relay LANE
Suite page suivante	

Groupe Suite ...	Objet
	Multiprotocol Switch SSP Token Ring UMTS RNC
Vendor Device	3Com Avici Systems Bay Networks Cabletron Cisco eXtreme Networks Fore Systems Foundry Networks Hewlett-Packard Juniper Networks Lucent (Ascend) Newbridge NEC Nortel Networks
C'est fini	

Ces objets concernés par *Device Creator* ne comprennent ni « Mobility Config » ni « manet_station » qui sont les 2 objets sur lesquels nous basons nos développements.

Du coup, nous répondons *Yes* la mise en garde de la Fig. A.8 pour signifier au système que nous avons bien compris le message !

La liste de choix pour l'attribut « model est affichée comme suit :

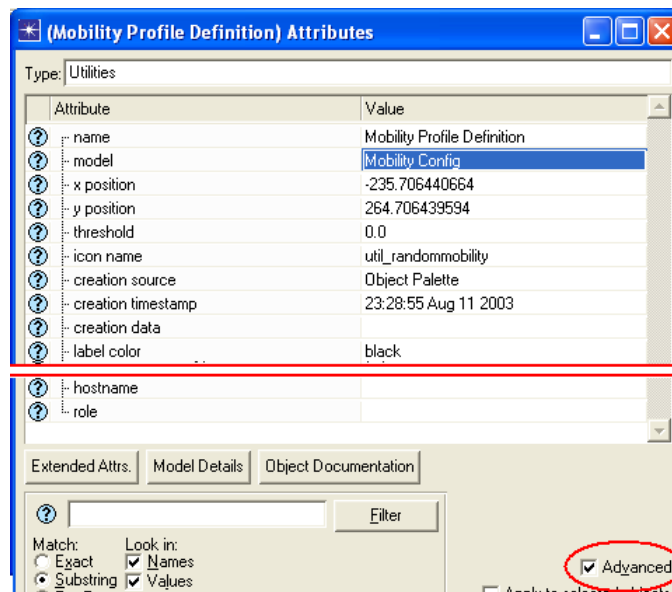
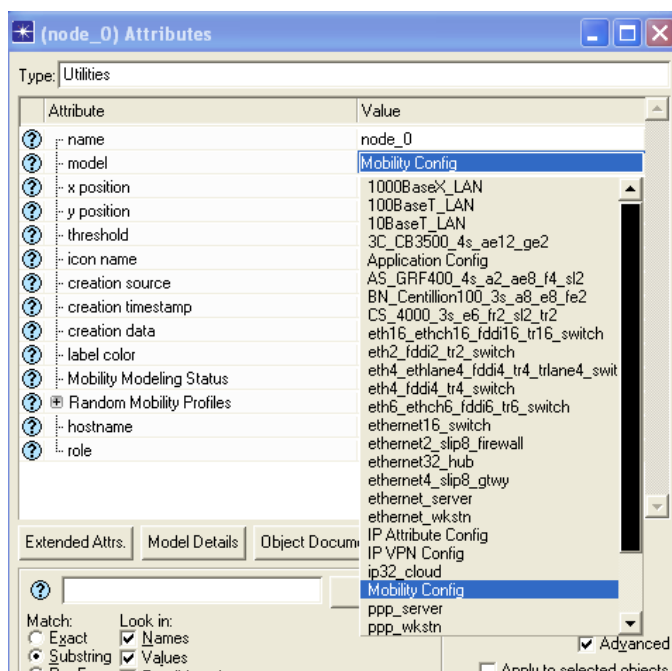


FIGURE A.7 – Éditeur d'attributs avancé de l'objet Mobility Config



Mais, nous en tenons pas compte car notre but est d'étudier l'existant et non pas de le modifier. Nous cliquons donc à nouveau la valeur actuelle de l'attribut « model » à savoir Mobility Config et ouvrons par là même, la boîte de dialogue ci-dessous :

A.3. ÉTUDE DE L'EXISTANT

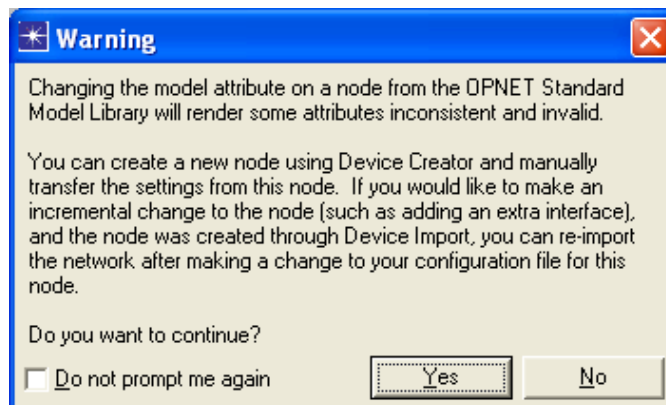
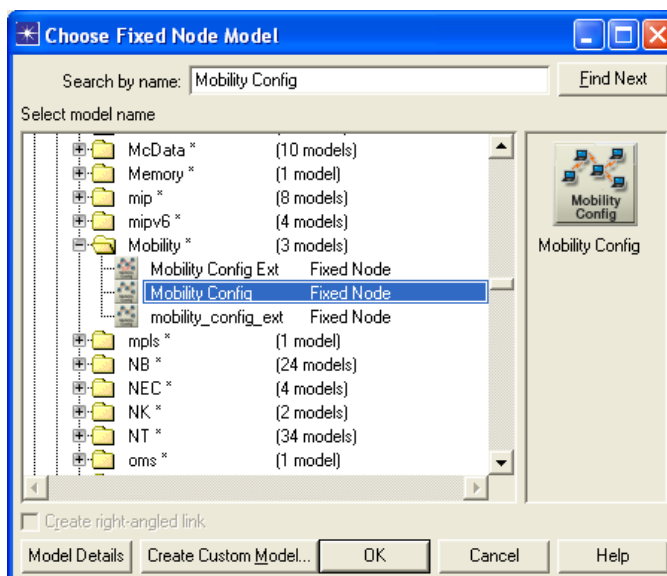
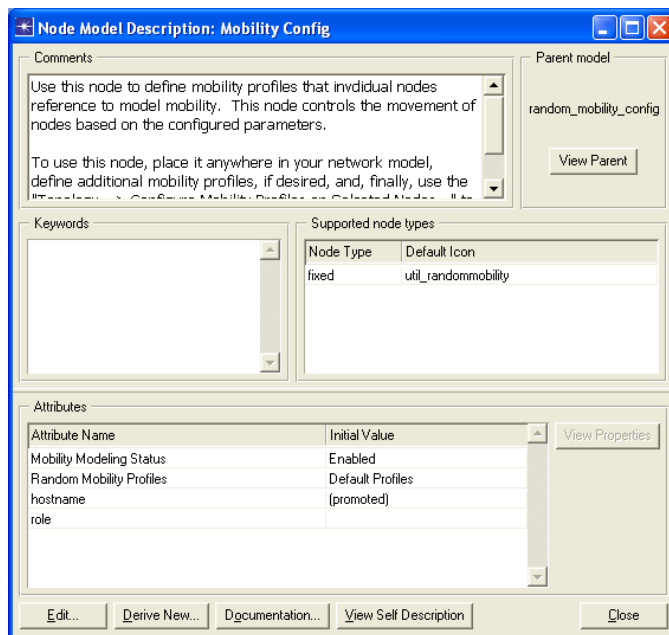


FIGURE A.8 – Mise en garde concernant les modifications de modèle d'objet *Opnet*



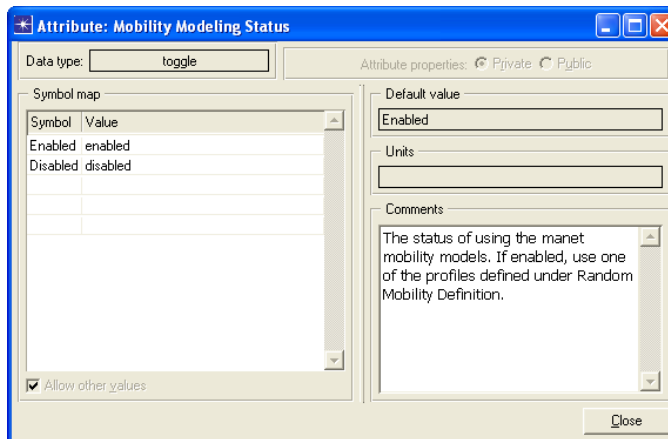
Nous cliquons droit sur l'icône en marge droite et faisons apparaître ceci :



Il s'agit de la description du modèle de nœud de l'objet « Mobility Config ». Un volet en haut à gauche donne un court commentaire sur la fonction du nœud. En haut à droite, nous trouvons le modèle parent qui est « random_mobility_config ». Au centre sur l'axe horizontal, nous trouvons un volet « Keywords » vide et un autre « Supported node types » en forme de tableau qui pour chaque type de nœud (fixed, mobile, satellite) donne l'icône associée. En l'occurrence, seule la ligne « fixed » est renseignée avec la valeur « util_randommobility ». Enfin en bas, le volet « Attributes » est un tableau qui liste les attributs et leurs valeurs initiales. Nous avons déjà rencontré plus haut, « Mobility Modeling Status » et « Random Mobility Profiles », qui était dans l'éditeur d'attributs simple. Nous y trouvons également « hostname » qui était présent dans l'éditeur d'attributs avancé. Nous constatons que la valeur initiale de cet attribut est « (promoted) », entre parenthèse donc : c'est sans doute ce qui fait qu'il ne s'affiche que dans l'éditeur d'attributs avancé. À droite de ce tableau, nous trouvons un bouton inactif (grisé) nommé « View Attribute ».

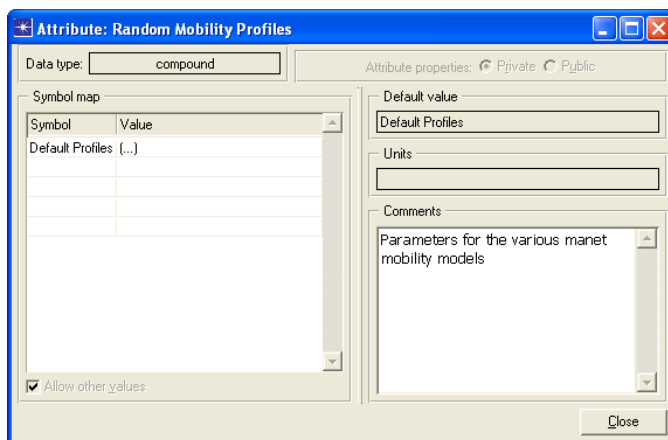
Nous sélectionnons l'attribut « Mobility Modeling Status », puis cliquons sur ce bouton « View Attribute » qui ouvre la fenêtre suivante :

A.3. ÉTUDE DE L'EXISTANT



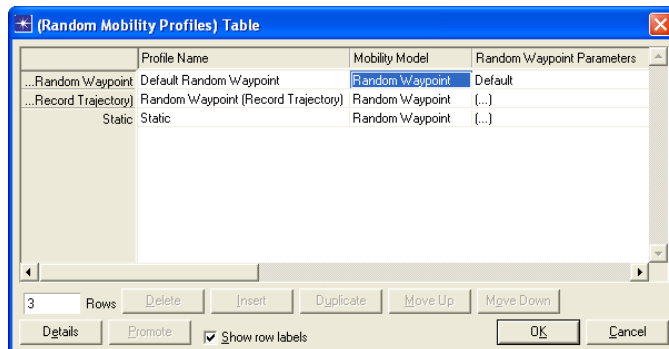
Elle nous donne : le type de l'attribut qui est « Toggle » (interrupteur à bascule) ; les valeurs possibles de cet attribut qui sont données dans un tableau à 2 colonnes « Symbol » (étiquette) et « Value » et qui sont l'étiquette « Enabled » de valeur « enabled » qui signifie que les modèles de mobilité définis dans l'objet « Mobility Config » sont actifs, et l'étiquette « Disabled » de valeur « disabled » qui signifie que ces modèles sont inactifs ; sa valeur par défaut qui est « Enabled » ainsi qu'un commentaire succinct qui nous dit que cette bascule contrôle l'utilisation des modèles (ou profils) de mobilité par les stations du réseau. Nous cliquons « OK » afin de revenir à la fenêtre précédente.

Nous sélectionnons maintenant l'attribut « Random Mobility Profiles » et cliquons le bouton « View Properties » qui ouvre la fenêtre suivante :



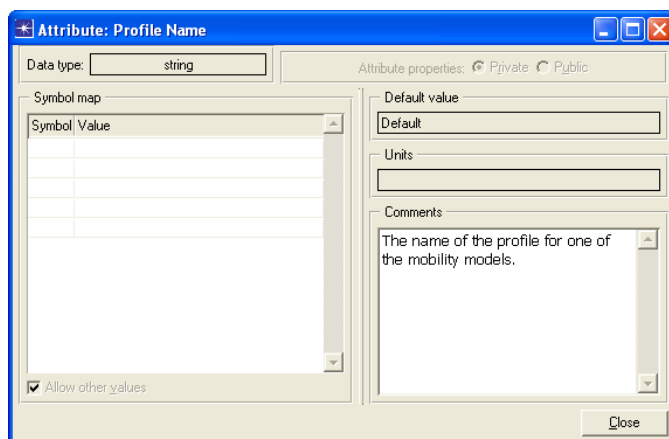
Elle nous donne : le type de l'attribut qui est « Compound » (composé) ; les valeurs possibles de cet attribut qui sont une à savoir l'étiquette « Default Profiles » et sa valeur « (...) » ; un commentaire qui nous informe du fait que cet attribut renferme l'ensemble

des modèles (profils) de mobilité. Nous cliquons la valeur « (...) » qui ouvre la fenêtre suivante :



Elle est un tableau de 3 colonnes qui sont « Profile Name », « Mobility Model » et « Random Waypoint Parameters » que nous avons déjà rencontrés Fig. A.6, p. 129, et 3 lignes qui sont les 3 modèles (profils) de mobilité définis par défaut : « Default Random Waypoint », « Random Waypoint (Record Trajectory) » et « Static ».

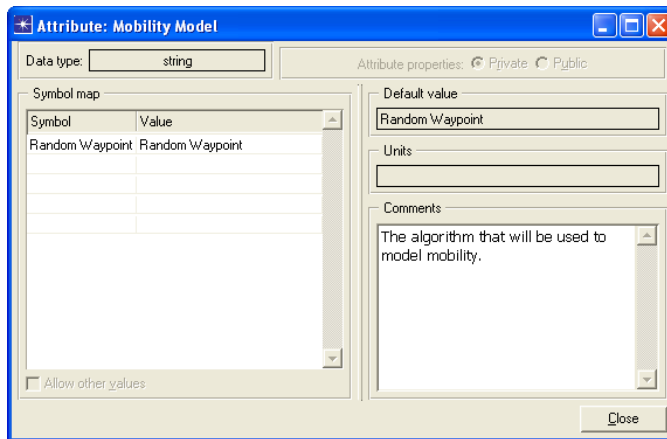
Nous sélectionnons « Default Random Waypoint » et cliquons le bouton « Details » qui ouvre la fenêtre suivante :



Elle nous donne le type « string » (chaîne de caractères) ; la valeur par défaut « Default » ainsi qu'un commentaire qui nous apprend que l'attribut « Profile Name » n'est autre que le nom du modèle (profil) de mobilité considéré. Nous cliquons « OK » pour revenir à la fenêtre précédente.

Nous sélectionnons « Random Waypoint » et cliquons le bouton « Details » qui ouvre la fenêtre suivante :

A.3. ÉTUDE DE L'EXISTANT



Elle nous donne : le type « string » ; l'étiquette « Random Waypoint » de valeur « Random Waypoint » ; la valeur par défaut « Random Waypoint » ainsi qu'un commentaire nous indiquant qu'il s'agit de l'algorithme qui implémente le modèle de mobilité. Nous cliquons « OK » pour revenir à la fenêtre précédente.

Nous sélectionnons « Random Waypoint Parameters » et cliquons le bouton « Details » qui ouvre la fenêtre suivante :

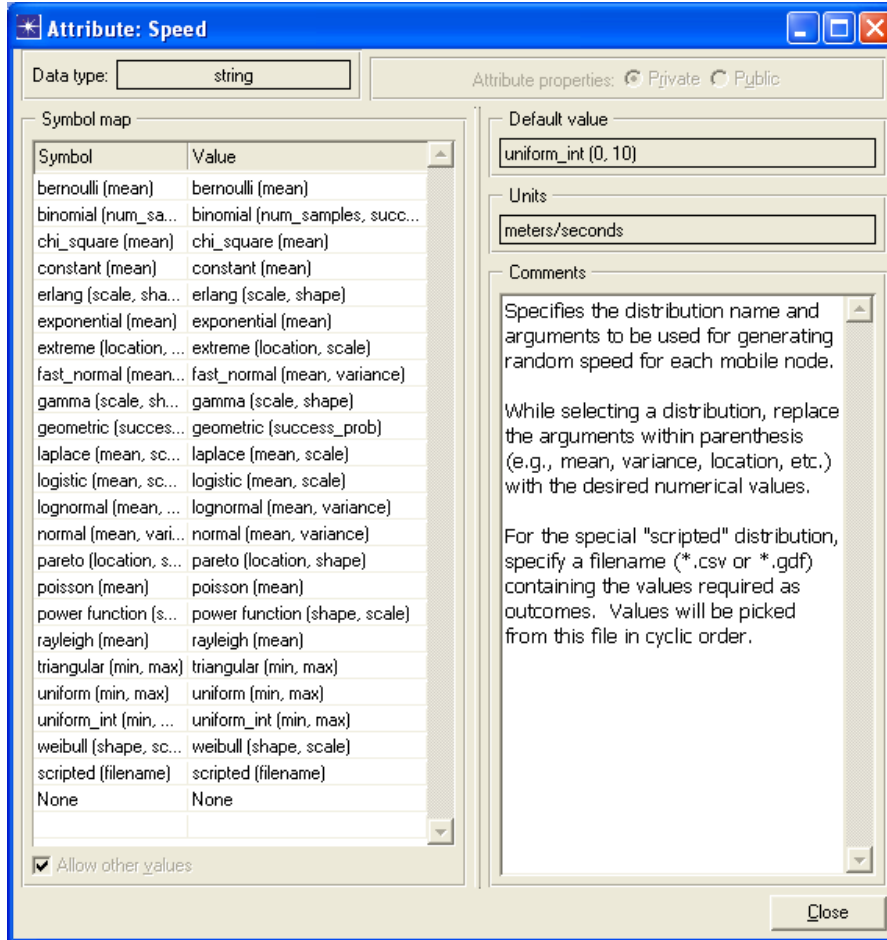
Attribute	Value
Mobility Domain Name	Not Used
x_min (meters)	0.0
y_min (meters)	0.0
x_max (meters)	500
y_max (meters)	500
Speed (meters/seconds)	uniform_int (0, 10)
Pause Time (seconds)	constant (100)
Start Time (seconds)	constant (10)
Stop Time (seconds)	End of Simulation
Animation Update Frequency (seconds)	1.0
Record Trajectory	Disabled

Elle nous donne un tableau à 2 colonnes « Attribute » et « Value » qui nous donne pour chaque paramètre du modèle de mobilité déjà rencontré Fig. A.6, p. 129, sa valeur.

Nous passons rapidement sur les premiers attributs de la liste qui sont soit des « string » (chaîne de caractères), soit des « double » (valeur numérique en virgule flottante).

Nous sélectionnons maintenant « Speed (meters/seconds) » et cliquons le bouton « De-

tails » qui ouvre la fenêtre suivante :

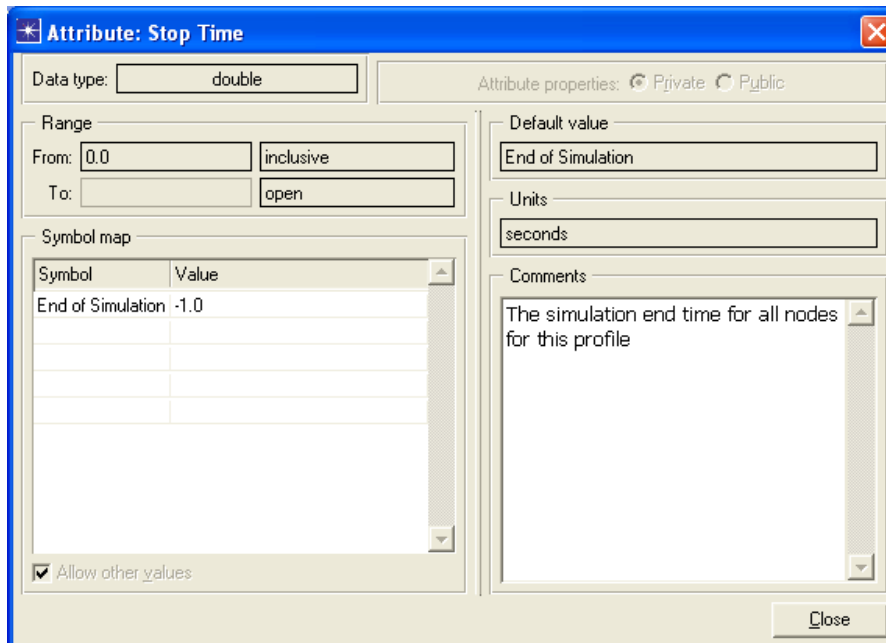


Elle illustre combien *Opnet Modeler* est complet d'un point de vue probabiliste : pas moins de 23 choix différents ! La valeur par défaut est « uniform_int(0, 10) » qui signifie « distribution uniforme en 0 et 10 m/s ». On remarque l'unité donnée dans la zone « Units ». Et bien sûr, le commentaire qui explique que l'on peut choisir dans la liste la distribution qui convient ainsi que ses arguments qu'il faut prendre soin de remplacer par les valeurs réelles. À noter la présence du type « scripted » où les valeurs de la distribution peuvent être lues dans un fichier externe du type *.csv, par exemple.

Les attributs « Pause (seconds) » et « Time (seconds) » dispose d'un choix identique et leurs valeurs par défaut sont respectivement « constant(100) » (pour la valeur constante 100) et « constant(10) ».

Nous sélectionnons maintenant « Stop Time (seconds) » et cliquons le bouton « Details »

qui ouvre la fenêtre suivante :



Elle nous donne : le type « double » ; l'étiquette « End of Simulation » de valeur « -1 » ; la valeur par défaut « End of Simulation » ainsi qu'un commentaire nous indiquant qu'il s'agit du temps de terminaison pour tous les nœuds qui implémente le modèle de mobilité. Nous cliquons « OK » pour revenir à la fenêtre précédente.

Nous cliquons « OK » pour fermer la fenêtre « (Random Waypoint Parameters) Table » et revenir à la fenêtre « (Random Mobility Profiles) Table ».

Nous cliquons sur la deuxième ligne sur « (...) » afin de constater que la seule différence entre « Default Random Waypoint » et « Random Waypoint (Record Trajectory) » est justement le fait que l'attribut « Record Trajectory » est à « Enabled dans ce dernier.

Quant au troisième modèle de mobilité par défaut, « Static », il a son attribut « Start Time (secondes) » à la valeur « None » qui signifie que la mobilité ne démarre jamais, résultant en un état statique des nœuds l'implémentant.

Nous cliquons « OK » pour fermer la fenêtre « (Random Mobility Profiles) Table » et revenir à la fenêtre « Attribute : Random Mobility Profiles ».

Nous cliquons « Close » pour fermer la fenêtre « Attribute : Random Mobility Profiles » et revenir à la fenêtre « Node Model Description : Mobility Config » que nous fermons par

un clic sur « Close » pour revenir à la fenêtre « Choose Fixed Node Model » que l'on ferme par « Cancel » qui nous fait revenir à l'éditeur d'attribut de « Mobility Config ».

Avec cela se termine notre étude de l'existant.

A.4 Nos premiers développements

Nous développons nos modèles mathématiques décrits dans les chapitres précédents, en proto-C, le langage de programmation d'*Opnet Modeler* et en amont, nous développons l'interface utilisateur dédiée qui facilite grandement la mise en œuvre des scénarios de simulation de nos modèles, par l'utilisation d'objets visuels. Mais avant de décrire ces développements, nous présentons dans les section suivantes, le verrou technique majeur que nous avons rencontré, le problème des intersections manquées, sa cause et sa solution.

A.5 Le problème des intersections manquées

Nous essayons donc naturellement de décliner ce modèle afin d'implémenter directement le modèle *Manhattan*, cf. la section 3.5 et la Fig. A.9.

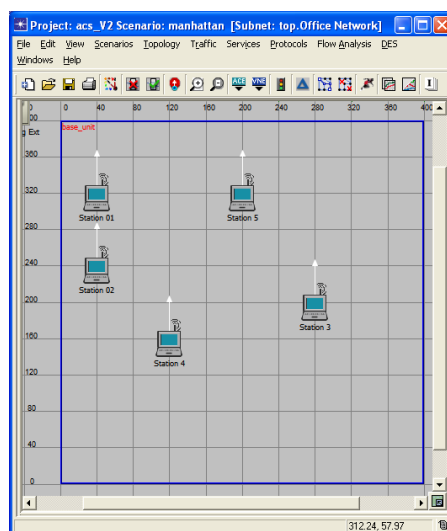


FIGURE A.9 – Implementation de Manhattan dans Opnet Modeler

Nous prenons bien soin de :

- placer chaque nœud sur une intersection, à l'initialisation du scénario ;
- définir l'incrément c'est-à-dire le pas de progression d'un nœud de sorte à ce qu'il soit bien un diviseur de la longueur standard d'un bloc d'immeuble.

Toutefois, nous constatons que de façon apparemment aléatoire, ce qui en soit est toujours très troublant, les nœuds de notre réseau ne parviennent pas exactement à certains points d'intersection de notre grille Manhattan ! En d'autres termes :

Certains nœuds manquent certaines intersections.

Après de nombreux essais très coûteux en temps, nous finissons par entrevoir la cause et trouver la solution. Afin de les exposer, il nous faut d'abord comprendre les différents systèmes de coordonnées *Opnet Modeler*. C'est l'objet de la section suivante.

A.6 Les 3 systèmes de coordonnées d'*Opnet Modeler*

En premier lieu, nous présentons dans cette section les 3 systèmes de coordonnées définis dans *Opnet Modeler* :

- le *système de coordonnées global* ;
- le *système de coordonnées relatif au sous-réseau* ;
- le *système de coordonnées géocentrique*.

A.6.1 Système de coordonnées global

Dans le *système de coordonnées global*, la position d'un objet du réseau est décrite par ses latitude, longitude et altitude. La latitude précise l'éloignement de l'objet vers le nord ou vers le sud, par rapport à l'équateur. Les latitudes sont positives dans l'hémisphère nord et négatives dans l'hémisphère sud. La longitude précise l'éloignement de l'objet vers l'est ou l'ouest, par rapport au méridien d'origine. Les longitudes positives, de 0° à $+180^\circ$, sont dans l'hémisphère oriental et les longitudes négatives, de 0° à -180° , sont dans l'hémisphère occidental. L'altitude est la hauteur de l'objet à la verticale de son lieu. L'altitude est mesurée en mètres au-dessus du niveau des mers. Voir la Fig. A.10.

Pour les sous-réseaux et les nœuds contenus dans un sous-réseau dont les coordonnées sont données en degrés, les attributs liés à la position de ces objets sont relatifs au *système*

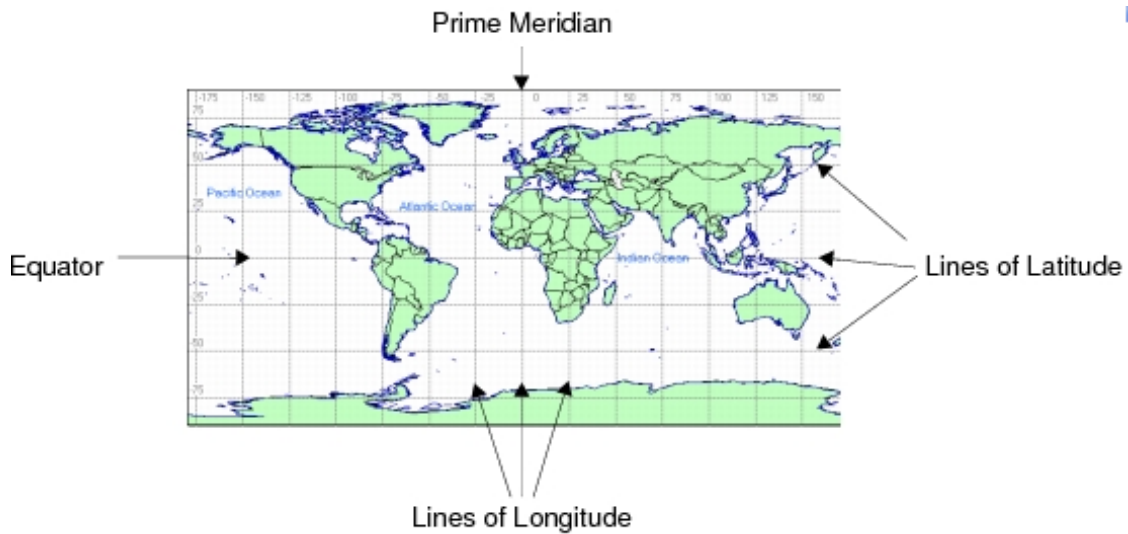


FIGURE A.10 – Le système de coordonnées global d’Opnet Modeler

de coordonnées global. Par exemple, dans le sous-réseau racine, l’attribut « position x » d’un nœud est sa longitude, tandis que son attribut « position y » est sa latitude ; enfin, son attribut « altitude » est son altitude en mètres.

Quel que soit le sous-réseau dans lequel un nœud se trouve, sa position globale peut être obtenue lors de la simulation par un appel à `op_ima_obj_pos_get()` qui est une fonction de base (*Kernel Procedure*). Si tous les parents d’un nœud sont des sous-réseaux fixes, les valeurs retournées pour un nœud fixe ne changent pas au fil du temps. Pour les nœuds mobiles ou satellitaires, les valeurs retournées reflètent la position du nœud à l’instant précis de l’appel de la fonction de base, dans la simulation.

A.6.2 Système de coordonnées relatif au sous-réseau

Dans le *système de coordonnées relatif au sous-réseau*, la position d’un objet du réseau est déterminée par son emplacement dans son sous-réseau. Si l’unité de ce dernier n’est pas le degré, les coordonnées de l’objet représentent son décalage par rapport à l’origine de son sous-réseau, qui est son coin supérieur gauche.

Un site possède les attributs « position x », « position y » et (à l’exception des sous-réseaux fixes) « altitude » qui définissent sa localisation. Dans le *système de coordonnées*

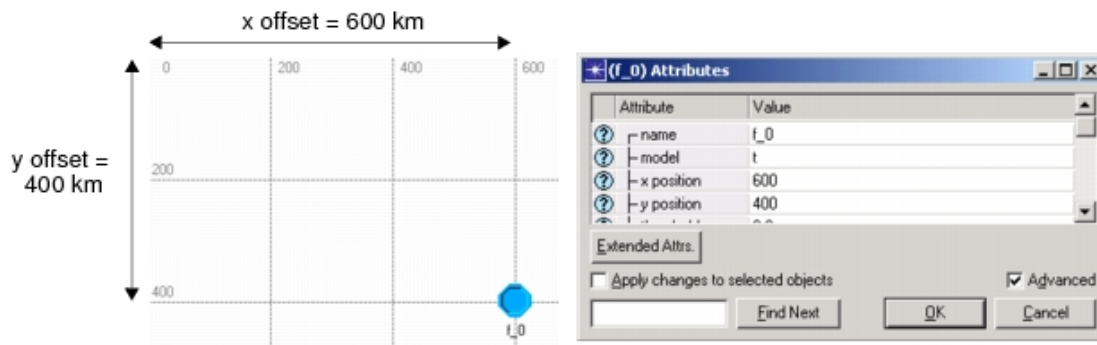


FIGURE A.11 – Le système de coordonnées relatif au sous-réseau d'Opnet Modeler

relatif au sous-réseau, les valeurs d'attribut sont dans l'unité du sous-réseau. L'attribut « position x » spécifie le décalage en abscisse et l'attribut « position y » spécifie le décalage en ordonnée par rapport à l'origine du sous-réseau, i.e. son coin supérieur gauche. L'attribut « altitude » indique l'altitude relative à son sous-réseau. L'élévation du sous-réseau lui-même est sa distance absolue au-dessus du niveau des mers. Les sous-réseaux fixes ont une altitude implicite de 0. À mesure qu'un site mobile ou satellitaire se déplace lors de la simulation, la valeur de ses attributs « position x », « position y » et « altitude » changent en conséquence. Voir la Fig. A.11.

Dans le *système de coordonnées relatif au sous-réseau*, déterminer la position globale d'un objet du réseau peut impliquer un ensemble de calculs récursifs. Si l'étendu de son sous-réseau est spécifié dans le *système de coordonnées global*, l'obtention de la position globale d'un objet ne nécessite que le calcul de son décalage en degrés depuis le coin supérieur gauche du sous-réseau. En revanche, si le sous-réseau est également dans un *système de coordonnées relatif au sous-réseau*, alors la position globale de son étendue, aussi, doit être calculée. Par conséquent, une série de calculs peut être nécessaire, afin que la position globale du sous-réseau soit connue. Tous ces calculs sont gérés par le noyau de simulation.

A.6.3 Système de coordonnées géocentrique

Le *système de coordonnées géocentrique* est une grille cartésienne tri-dimensionnelle dont l'origine est le centre de la terre. Il comporte trois axes (x, y et z) et a pour unité le

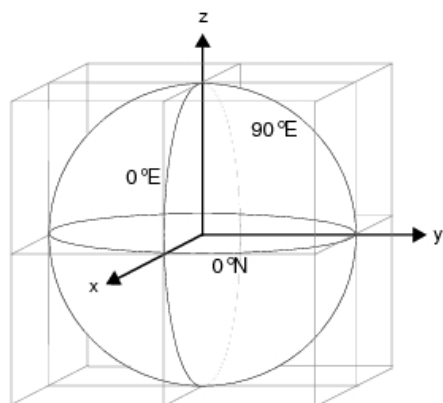


FIGURE A.12 – Le système de coordonnées géocentrique d’Opnet Modeler

mètre. Chacun des axes est défini par un vecteur qui part du centre de la terre et croise un point de la surface terrestre. Voir la Fig. A.12.

A.7 La cause du problème

Quoiqu’*Opnet Modeler* aient effectivement 3 systèmes de coordonnées différents, objet de la section précédente, de façon interne à *Opnet Modeler*, le seul système de coordonnées utilisé est celui des coordonnées géographiques.

Or, cette façon interne de procéder ne répond pas aux besoins de précision de nos modèles de mobilité. En effet, la conversion interne des coordonnées cartésiennes en coordonnées géographiques, n’est pas suffisamment exacte. Nous le constatons en traçant au pas à pas le code source du premier modèle que nous implémentons : *Manhattan*.

C’est ce qui explique que parfois un nœud du réseau « manque » une intersection.

Nous n’en croyions pas nos yeux malgré les tests innombrables que nous avons joués et rejoués ! La cause s’est effectivement avérée être cette imprécision relative de la conversion des coordonnées cartésiennes vers celles géographiques, voir Fig. A.13 où le décalage (et donc le saut de l’intersection) est multiplié par 1000 environ afin de rendre visible l’erreur de conversion.

Nous renonçons donc à prendre comme point de départ *Random Waypoint* et sa représentation interne des coordonnées.

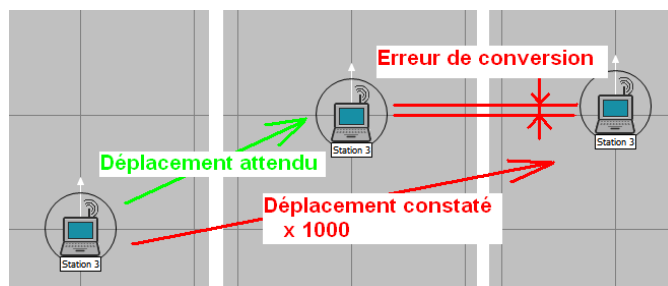


FIGURE A.13 – Erreur de conversion dans Opnet Modeler X 1000

A.8 La solution du problème

La solution du problème des intersections manquées est donc, malgré le fait que cela allait nous causer un effort de développement logiciel de grosse envergure, comme nous allons le voir, de développer notre modèle *Manhattan* à partir de zéro, afin d'être en coordonnées cartésiennes au niveau de l'Interface-Homme-Machine et au niveau interne. Pour commencer, nous devons créer nos propres objets logiciels d'interface visuelle *Opnet Modeler*, c'est ce dont traite la section suivante.

A.9 Création d'objet

Dans *Opnet Modeler*, nous disposons de 2 méthodes de création d'un objet à partir d'un objet existant :

- **la méthode par dérivation d'objet** depuis la palette des objets. Cette méthode ne permet pas de changer les traitements de l'objet père dans l'objet dérivé ou objet fils. En d'autres termes, le père et le fils partagent le même code. Une *dérivation Opnet Modeler* ne sert donc qu'à réaliser des changements dans la partie visuelle du nouvel objet ;
- **la méthode par duplication d'objet** depuis le gestionnaire de fichiers du système d'exploitation, par copier/coller du fichier de traitement de l'objet, à savoir son modèle de processus dont l'extension est pr.m pour *process model*, puis par son intégration dans la palette des objets. Dans ce cas, non seulement l'interface visuelle du nouvel objet est modifiable mais également ses traitements.

A.9.1 Par duplication

Le développement d'un nouvel objet visuel n'étant pas particulièrement bien documenté, nous en recherchons le mode de développement par nous même. Nous l'exposons ci-après en commençant par le premier objet que nous développons pour implémenter *Manhattan* à savoir une extension de l'objet *Mobility Config*.

Il convient ici d'ouvrir une parenthèse. Le nom de fichier du modèle de processus de l'objet visuel *Mobility Config* est : « random_mobility_mgr.pr.m ». On aurait pu s'attendre à ce que le nom de fichier du modèle de processus de l'objet visuel *Mobility Config* soit justement « mobility_config.pr.m » mais c'est malgré tout « random_mobility_mgr.pr.m » qui a été choisi. Cela semble indiquer que les développeurs d'*Opnet Modeler* ne pensaient a priori pas implémenter de modèle de mobilité qui soit non aléatoire du point de vue du choix de l'azimut. Nous fermons la parenthèse.

Dans notre cas, nous changeons le comportement de l'objet père. Nous devons donc mettre en œuvre la *méthode par duplication d'objet*, voir la section A.9.


En conséquence, nous commençons par copier/coller le fichier de modèle de processus de l'objet visuel *Mobility Config*, à savoir « random_mobility_mgr.pr.m » en « mobility_config_ext_BIS.pr.m ». L'ajout de « ext » signifie « extension » et le « BIS » exige le *nota* qui suit.

nota

Pour rédiger les procédures de mise en œuvre *Opnet Modeler* de cette thèse, nous les avons entièrement rejouées. Et, afin de disposer de nos nouveaux objets tout en les recréant, nous avons choisi de les recréer en suffixant leurs noms de BIS.

Nous choisissons d'abandonner l'adjectif *random* car nos modèles ne sont pas aléatoires du point de vue de l'azimut comme c'est le cas du *Random Walk*. Par ailleurs, *mgr* qui signifie *manager* c'est-à-dire *gestionnaire* n'apporte pas suffisamment d'information pour le préférer à *conf* qui a l'avantage d'être proche du nom de son objet visuel.

Nous l'intégrons à la palette d'objets comme suit :

1. lancer *Opnet Modeler* ;
2. ouvrir un projet existant ou en créer un nouveau sans changer les valeurs par défaut, l'objectif étant d'accéder à la palette d'objets ;
3. cliquer « Topology | Open Object Palette » afin d'ouvrir la palette d'objets si elle ne s'est pas ouverte automatiquement ;
4. cliquer le bouton  en haut à gauche, afin de passer en « Icon View », voir Fig. A.14 ;

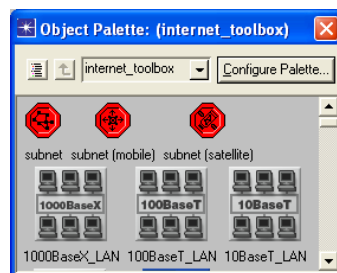


FIGURE A.14 – Boîte de dialogue Icon View d'Opnet Modeler

5. cliquer le bouton « Configure Palette » en haut à droite ;
6. constater l'affichage de la boîte de dialogue « Configure Palette », voir Fig. A.15 ;

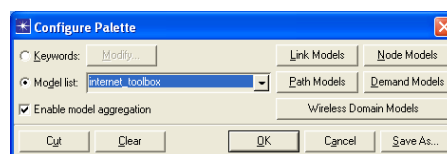


FIGURE A.15 – Boîte de dialogue Configure Palette d'Opnet Modeler

7. cliquer le bouton « Clear » en bas à gauche ;
8. constater que seuls 3 objets demeurent dans la palette, à savoir *subnet*, *subnet (mobile)*, *subnet (satellite)* ;
9. cliquer le bouton « Node Models » ;
10. constater l'affichage de la boîte de dialogue « Select Included Entries » ;
11. se déplacer jusqu'à la ligne « mobility_config_ext » ;

12. cliquer dans la colonne « Status » sur cette ligne afin de basculer de « not included » à « included », voir Fig. A.16 ;

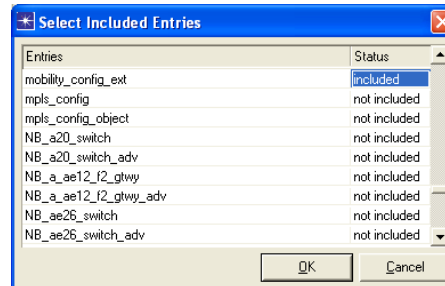


FIGURE A.16 – Boîte de dialogue Select Included Entries d’Opnet Modeler

13. cliquer le bouton « OK » ;
14. constater le retour sur la boîte de dialogue « Configure Palette » ;
15. cliquer le bouton « Save as... » ;
16. constater l’affichage de la boîte de dialogue standard « Save As » ;
17. se déplacer jusqu’au dossier suivant :

```
D:\Program Files\OPNET\16.0.A\models\std\manet
```

où D représente la lettre du lecteur sur lequel est installé *Opnet Modeler* ;
18. nommer le fichier « MANET_EXT » qui aura par défaut l’extension « .cml » pour « Custom Model List » ;
19. constater l’affichage de « MANET_EXT » dans la liste de choix « Model List » ;
20. constater l’affichage de l’icône de l’objet « Mobility Config » à la suite des 3 objets sus-cités ;
21. constater l’affichage de « mobility_config_ext » sous cette icône, voir Fig. A.17 ;
22. cliquer le bouton « OK » ;
23. constater le retour sur la palette d’objets ;
24. cliquer le bouton « OK » pour la fermer.

À ce stade, notre nouvel objet est ajouté dans la palette d’objets dans la nouvelle branche « MANET_EXT ».

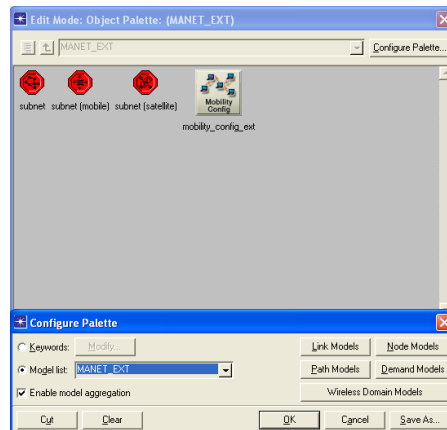


FIGURE A.17 – Configure Palette après ajout de l'objet `mobility_config_ext`

Il nous faut cependant procéder à 2 mises au point sur notre nouvel objet, l'une facultative mais de bon aloi afin de lui donner un nom qui suive le standard des noms des objets dans la palette, l'autre impérative pour distinguer de façon visuelle notre objet de son parent, à savoir lui donner une icône distinctive.

A.9.2 Par dérivation

Afin qu'un nom d'objet suive le standard de désignation des objets dans la palette, son nom doit être formé de mots écrits en lettres minuscules sauf la lettre initiale qui est en majuscule. Ces mots doivent être espacés par des espaces et non des sous-tirets, comme c'est le cas actuellement. En d'autres termes, « `mobility_config_ext` » doit devenir « Mobility Config Ext ». Pour ce faire, nous déroulons les étapes complémentaires suivantes qui correspondent à la première méthode de création d'objet, à savoir, la *méthode par dérivation d'objet*, voir la section A.9 :

1. cliquer « Topology | Open Objet Palette » ;
2. constater l'affichage de la palette d'objets ;
3. se déplacer jusqu'à la branche « MANET_EXT » et l'ouvrir puis ouvrir « Node Models » ;
4. cliquer droit sur notre nouvel objet « `mobility_config_ext` » et choisir « View Model Details » ;

- constater l'affichage de la boîte de dialogue « Node Model Description : mobility_config_ext », voir Fig. A.18 ;

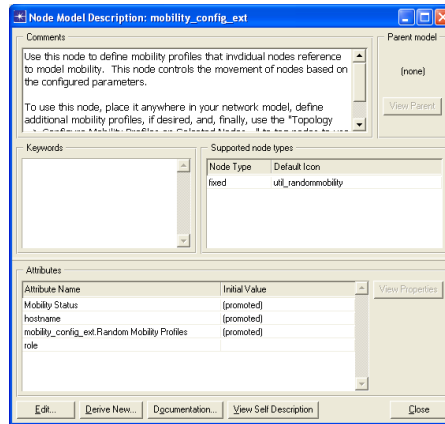


FIGURE A.18 – Configurer Palette après ajout de l'objet mobility_config_ext

- cliquer le bouton « Derive New... » ;
- constater l'affichage de la boîte de dialogue « Derive New Node Model », voir Fig. A.19 ;

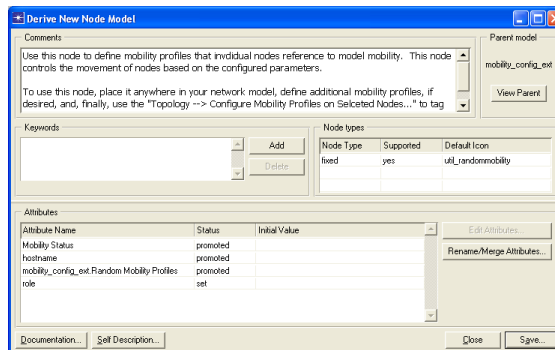


FIGURE A.19 – Boîte de dialogue Derive New Node Model d'Opnet Modeler

- cliquer le bouton « Save... » ;
- constater l'affichage de la boîte de dialogue standard « Save As » ;
- se déplacer jusqu'au dossier suivant :
D:\Program Files\OPNET\16.0.A\models\std\wireless
où D représente la lettre du lecteur sur lequel est installé *Opnet Modeler* ;
- nommer le fichier « Mobility Config Ext » qui aura par défaut l'extension « .nd.m » pour « Node Model » et cliquer « OK » ;

A.9. CRÉATION D'OBJET

- constater que l'affichage de la boîte de dialogue « Derive New Node Model » a été supplanté par celui de la boîte de dialogue « Edit Derived Model : Mobility Config Ext », voir Fig. A.20 ;

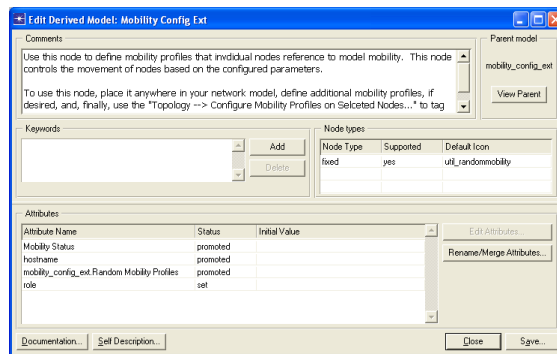


FIGURE A.20 – Boîte de dialogue Edit Derived Model d'Opnet Modeler

- relever le nom de l'icône (qui est par défaut celle de l'objet père) qui est situé dans le tableau « Node Types » dans la colonne « Default Icon », à savoir : *util_randommobility* ;
- cliquer sur ce nom ;
- constater l'affichage de la boîte de dialogue « Icon Palette », voir Fig. A.21 ;

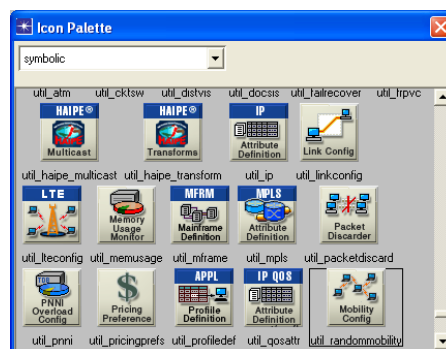


FIGURE A.21 – Boîte de dialogue Icon Palette d'Opnet Modeler

- relever le nom situé dans la boîte de liste sous le titre de la boîte de dialogue : il s'agit de la base de données où l'icône est situé qui est donc : *symbolic* ;
- cliquer sur la case de fermeture (la croix blanche sur un carré rouge) en haut à droite de la boîte de dialogue afin de la fermer ;

18. cliquer le bouton « Close » afin de fermer la boîte de dialogue « Edit Derived Model : Mobility Config Ext » ;
19. cliquer le bouton « Close » afin de fermer la palette d'objets ;

Le nom de l'icône de l'objet père est donc :

util_randommobility.

Tandis que sa base de données d'icônes est :

symbolic.

Nous nous servons de ceux-ci à la section suivante.

A.9.3 Création d'icône

Afin de dupliquer l'icône de l'objet père et la modifier pour l'attribuer à notre nouvel objet, nous avons besoin de connaître le nom de l'icône de l'objet père. Par ailleurs, il nous faut connaître également le lieu où cette icône se trouve dans l'arborescence des fichiers d'*Opnet Modeler*. Fort heureusement, nous les avons relevés à la section précédente, il s'agit respectivement de : *util_randommobility* et *symbolic*.

Procédons maintenant à la duplication et à la modification de l'icône parente :

1. cliquer « File | Open... » ;
2. constater l'affichage de la boîte de dialogue standard « Open » ;
3. sélectionner le type de fichier « Icon Database File (*.icons) » ;
4. se déplacer jusqu'au dossier suivant :
`D:\Program Files\OPNET\16.0.A\models\std\icons`
où D représente la lettre du lecteur sur lequel est installé *Opnet Modeler* ;
5. sélectionner le fichier « symbolic » qui a pour extension « .icons » ;
6. rechercher visuellement (il n'y a pas d'autre moyen... il s'agit en gros de la 800^{ième} icône...), l'icône *util_randommobility* ;
7. cliquer droit sur l'icône et choisir « Copy » afin de copier l'icône dans le presse-papier ;
8. cliquer dans le menu principal « Edit | Paste » afin de coller la nouvelle icône en cliquant dans l'espace libre de la base d'icône ;

A.9. CRÉATION D'OBJET

9. cliquer droit sur la nouvelle icône et choisir « Set Name » et la nommer en : *util_randommobility_ext* ;
10. cliquer droit sur la nouvelle icône et choisir « Edit » ;
11. constater l'affichage de la boîte de dialogue « Image Editor », voir Fig. A.22 ;

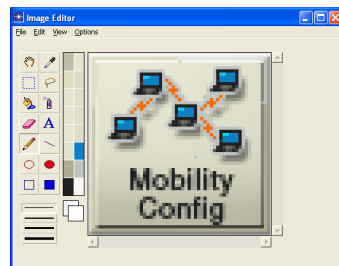


FIGURE A.22 – Boîte de dialogue Image Editor d'Opnet Modeler

12. cliquer « Options | Edit Colors... » afin de changer de palette de couleurs ;
13. constater l'affichage de la boîte de dialogue « Color Palette Editor », voir Fig. A.23 ;

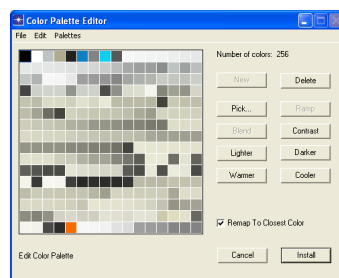


FIGURE A.23 – Boîte de dialogue Color Palette Editor d'Opnet Modeler

14. cliquer « Palettes | Palette couleurs... » afin de passer à la palette couleurs ;
15. cliquer le bouton « Install » ;
16. constater que la couleur rouge est maintenant disponible par les couleur de l'« Image Editor » ;
17. cliquer l'outil texte (lettre A), cliquer la couleur rouge et écrire « Ext » au milieu de la nouvelle icône, voir Fig. A.24 ;
18. cliquer sur « File | Save » afin d'enregistrer la modification de l'icône ;
19. cliquer sur « File | Close » pour fermer l'éditeur d'icône et revenir à la base d'icône ;
20. cliquer sur « File | Save » afin d'enregistrer la modification de la base d'icônes ;

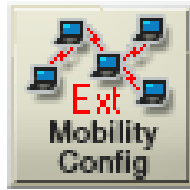


FIGURE A.24 – Icône distinctive de notre nouvel objet *Mobility Config Ext*

21. cliquer sur « File | Close » pour fermer la base d’icône ;

A.9.4 Intégration d’icône et désignation standard

De retour sur la fenêtre projet, nous insérons l’objet possédant la bonne désignation dans la palette d’objet et retirons par là même l’objet intermédiaire « mal nommé ». Enfin, nous lui attribuons sa nouvelle icône. Ci-après la procédure de mise en œuvre :

1. cliquer « Topology | Open Objet Palette » ;
2. constater l’affichage de la palette d’objets ;
3. cliquer le bouton en haut à gauche afin de passer en « Icon View », voir Fig. A.14, p. 148 ;
4. cliquer le bouton « Configure Palette » en haut à droite ;
5. constater l’affichage de la boîte de dialogue « Configure Palette » ;
6. cliquer le bouton « Clear » en bas à gauche ;
7. constater que seuls 3 objets demeurent dans la palette, à savoir *subnet*, *subnet (mobile)*, *subnet (satellite)*, voir Fig. A.15, p. 148 ;
8. cliquer le bouton « Node Models » ;
9. constater l’affichage de la boîte de dialogue « Select Included Entries » ;
10. se déplacer jusqu’à la ligne « Mobility Config Ext » ;
11. cliquer dans la colonne « Status » sur cette ligne afin de basculer de « not included » à « included » ;
12. cliquer le bouton « OK » ;
13. constater le retour sur la boîte de dialogue « Configure Palette » ;

14. cliquer le bouton « Save as... » ;
15. constater l'affichage de la boîte de dialogue standard « Save As » ;
16. se déplacer jusqu'au dossier suivant :
`D:\Program Files\OPNET\16.0.A\models\std\manet`
où D représente la lettre du lecteur sur lequel est installé *Opnet Modeler* ;
17. nommer le fichier « MANET_EXT » et répondre « OK » à la confirmation de remplacement dudit fichier ;
18. constater l'affichage de « MANET_EXT » dans la liste de choix « Model List » ;
19. constater l'affichage de l'icône de l'objet « Mobility Config » à la suite des 3 objets sus-cités ;
20. constater l'affichage de « Mobility Config Ext » sous cette icône ;
21. cliquer le bouton « OK » ;
22. constater le retour sur la palette d'objets ;
23. cliquer droit sur notre nouvel objet et choisir « View Model Details » ;
24. constater l'affichage de la boîte de dialogue « Node Model Description : Mobility Config Ext » ;
25. cliquer le bouton « Edit... » ;
26. constater que l'affichage de la boîte de dialogue « Edit Derived Model : Mobility Config Ext », voir Fig. A.20, p. 152 ;
27. cliquer sur le nom de l'icône situé dans le tableau « Node Types » dans la colonne « Default Icon », à savoir : *util_randommobility* ;
28. constater l'affichage de la boîte de dialogue « Icon Palette » ;
29. choisir la nouvelle icône, à savoir : *util_randommobility_ext* ;
30. cliquer le bouton « Save... » ;
31. constater l'affichage de la boîte de dialogue standard « Save As » ;
32. se déplacer jusqu'au dossier suivant :
`D:\Program Files\OPNET\16.0.A\models\std\wireless`

- où D représente la lettre du lecteur sur lequel est installé *Opnet Modeler* ;
33. nommer le fichier « Mobility Config Ext » et cliquer « OK » : pas de demande de confirmation de remplacement dudit fichier (...);
 34. cliquer le bouton « Close » afin de fermer la boîte de dialogue « Edit Derived Model : Mobility Config Ext » ;
 35. cliquer le bouton « Close » afin de fermer la palette d'objets ;

A.10 Attribut d'objet

À ce stade, nous disposons d'un nouvel objet avec un nom qui suit la désignation standard, une icône dérivée de l'icône mère mais spécifique au nouvel objet. Toutefois, les attributs de notre nouvel objet visuel sont en tous points identiques à ceux de l'objet père.

A.10.1 Nom d'attribut

L'étape suivante consiste donc à enrichir l'interface visuelle de notre nouvel objet afin qu'il propose parmi ses attributs, le modèle de mobilité *Manhattan*.

Cela demande non seulement d'ajouter des attributs à notre objet mais aussi et d'abord de modifier certains noms d'attribut afin de les mettre en conformité avec ce qui va être ajouté.

Par exemple, nous avons déjà mentionné au début de la section A.9.1, dans une parenthèse explicite, que le nom de fichier du modèle de processus de l'objet visuel *Mobility Config* était : « random_mobility_mgr.pr.m » et qu'on aurait pu s'attendre à ce que le nom de fichier du modèle de processus de l'objet visuel *Mobility Config* soit justement « mobility_config.pr.m » mais c'est malgré tout « random_mobility_mgr.pr.m » qui a été choisi. Nous avons proposé d'interpréter cela comme semblant indiquer que les développeurs d'*Opnet Modeler* ne pensaient a priori pas implémenter de modèle de mobilité qui soit non aléatoire du point de vue du choix de l'azimut. En conséquence, lors du copier/coller du fichier de modèle de processus de l'objet visuel *Mobility Config*, à savoir « random_mobility_mgr.pr.m » nous le renommâmes « mobility_config_ext.pr.m ». Nous choisissions d'abandonner l'adjectif *random* car nos modèles ne sont pas aléatoires

du point de vue de l'azimut. En sus, *mgr* qui signifie *manager* c'est-à-dire *gestionnaire* n'apportait pas suffisamment d'information pour le préférer à *conf* qui a l'avantage d'être proche du nom de son objet visuel.

Il en va de même pour les noms d'attribut ! À cela près que les noms d'attribut sont définissables et re-définissables. C'est-à-dire qu'un même attribut peut avoir un *nom original* lequel est utilisé dans le code des traitements de cet objet, et un *nom visuel* lequel est vu par l'utilisateur. Si de plus, cet objet a un fils, celui-ci pourra, s'il a été créé par la *méthode de dérivation d'objet* (cf. section A.9.2, p. 150), en changer le *nom visuel*, et s'il a été créé par la *méthode de duplication d'objet* ((cf. section A.9.1, p. 147), en changer le *nom original* ! En d'autres termes, le nombre maximal de noms différents que peut posséder un même attribut est :

$$n = 2^d$$

où d est le nombre de générations : $d = 1$ père seul, $d = 2$ père+fil, $d = 3$ père+fil+petit-fil etc..

NOTE IMPORTANTE

Bien qu'il soit toujours théoriquement possible de changer le *nom original* ou le *nom visuel* d'un attribut d'objet, si cet objet a déjà été instancié dans un scénario de projet *Opnet Modeler*, le changement aura pour effet de bord, le retour aux valeurs par défaut de tous les attributs de l'instance ! Il est clair que les conséquences sont catastrophiques, s'il s'agit d'un projet de plusieurs centaines de scénario, comme c'est le cas dans notre thèse : tous les attributs de toutes les instances de l'objet en question seraient à redéfinir dans tous les scénarios !

A.10.2 Modification

À titre d'exemple, affichons les noms originaux des attributs de premier niveau de notre nouvel objet visuel :

1. cliquer « Topology | Open Objet Palette » ;
2. constater l’affichage de la palette d’objets ;
3. se déplacer jusqu’à la branche « MANET_EXT » et l’ouvrir puis ouvrir « Node Models » ;
4. cliquer droit sur notre nouvel objet « Mobility Config Ext » et choisir « View Model Details » ;
5. constater l’affichage de la boîte de dialogue « Node Model Description : Mobility Config Ext » ;
6. cliquer le bouton « View Parent » en haut à droite ; en effet les attributs sont rattachés à l’objet « mobility_config_ext » et non à l’objet « Mobility Config Ext » qui n’est qu’une enveloppe visuelle comme nous l’avons vu à la section précédente ;
7. cliquer le bouton « Edit... » ;
8. constater l’affichage de la fenêtre « Node Model : mobility_config_ext », voir Fig. A.25 ;

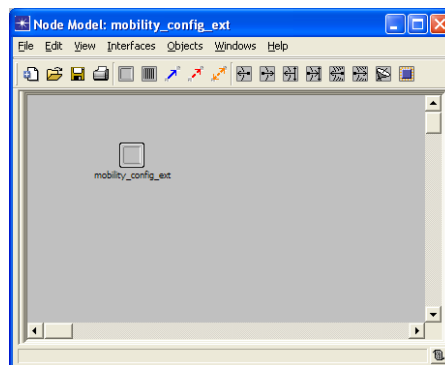


FIGURE A.25 – Fenêtre de l’éditeur de modèle de nœud d’Opnet

9. double-cliquer sur l’unique nœud de la zone cliente de la fenêtre ;
10. constater l’affichage de la fenêtre « Process Model : mobility_config_ext », voir Fig. A.26 ;
11. cliquer dans le menu « Interfaces | Model Attributes »
12. constater l’affichage de la boîte de dialogue « Model Attributes : mobility_config_ext », voir Fig. A.27.

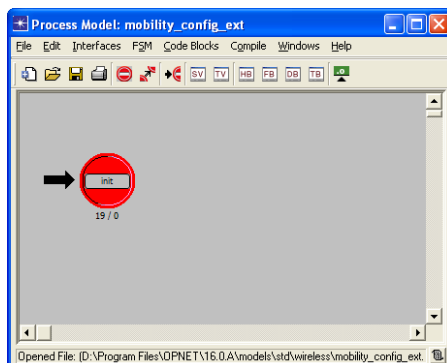


FIGURE A.26 – Fenêtre de l'éditeur de processus d'Opnet

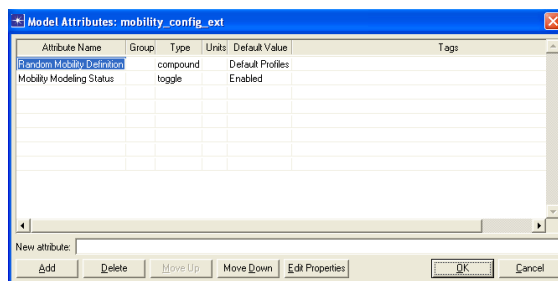


FIGURE A.27 – Boîte de dialogue Model Attributes Initiale de mobility_config_ext

Les sections qui suivent montre les modes opératoires des :

- Modification du nom original d'un attribut d'objet ;
- Modification du nom visuel d'un attribut d'objet.

Modification du *nom original* d'un attribut d'objet

Elle se fait dans la boîte de dialogue « Model Attributes : mobility_config_ext », voir Fig. A.27.

Nous choisissons de changer le nom « Mobility Modeling Status » en « Mobility Status » car c'est lors de la simulation que cette donnée est testée par les traitements, et à ce moment ça n'est pas de modélisation dont il s'agit mais bien de simulation. En outre, « Random Mobility Definitions » devient « Random Mobility Profiles », afin de rapprocher le *nom original* du *nom visuel* et nous simplifier la lecture des traitements.

La boîte de dialogue ci-dessus devient semblable à la Fig. A.28

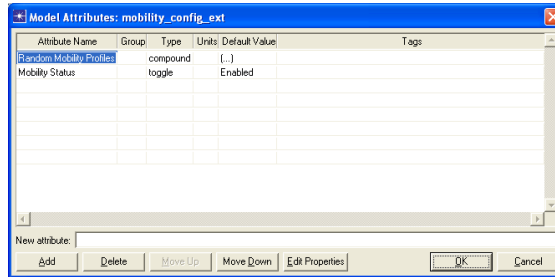


FIGURE A.28 – Boîte de dialogue Model Attributes de Mobility Config Ext

où *Mobility Modeling Status* est devenu *Mobility Status*.

Ensuite, comme nous l'avons déjà mentionné, il nous faut modifier les traitements qui adressent le *nom original*, et qui sont dans le fichier du modèle de processus associé : *mobility_config_ext.pr.m*. Pour ce faire, nous déroulons les actions suivantes (ici pour un seul des 2 attributs ci-dessus) :

1. cliquer le bouton « OK » pour fermer la boîte de dialogue « Model Attributes : mobility_config_ext » et revenir à la fenêtre « Process Model : mobility_config_ext », voir Fig. A.26, p. 160 ;
2. double-cliquer sur la partie *supérieure* du disque rouge de l'état « init » et dans la fenêtre qui s'ouvre remplacer *Mobility Modeling Status* par *Mobility Status*, voir Fig. A.29 ;

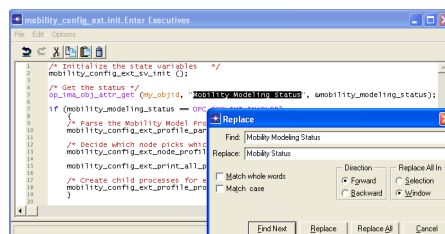


FIGURE A.29 – Modification du code de l'état Init du processus Mobility Config Ext

3. passer en revue les unes après les autres les commandes du menu « Code Blocks » et constater par une recherche de *Mobility Modeling Status* que ce texte n'apparaît dans aucun de ces blocks.
4. cliquer dans le menu « Compile | Compile code » ;

5. constater l'affichage de la boîte de dialogue « compilation réussie », voir Fig. A.30 ;

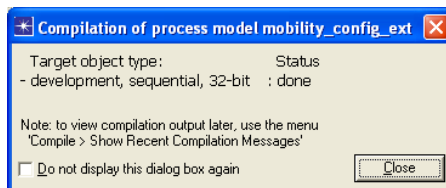


FIGURE A.30 – Boîte de dialogue de compilation réussie

6. cliquer le bouton « OK » pour fermer cette boîte de dialogue ;
7. cliquer le menu « File | Save » puis « File | Close » pour d'abord sauvegarder le modèle de processus puis fermer cette fenêtre ;
8. constater l'affichage de la fenêtre « Node Model : mobility_config_ext », voir Fig. A.25 ;
9. cliquer le menu « File | Save » puis « File | Close » pour d'abord sauvegarder le modèle de processus puis fermer cette fenêtre ;
10. constater que l'affichage de la boîte de dialogue « Edit Derived Model : Mobility Config Ext », voir Fig. A.20, p. 152 ;
11. cliquer le bouton « Close » pour fermer cette fenêtre ;
12. constater l'affichage de la palette d'objets ;
13. cliquer le bouton « Close » pour la fermer ;

Modification du *nom visuel* d'un attribut d'objet

Elle se fait comme suit :

1. cliquer « Topology | Open Objet Palette » ;
2. constater l'affichage de la palette d'objets ;
3. se déplacer jusqu'à la branche « MANET_EXT » et l'ouvrir puis ouvrir « Node Models » ;
4. cliquer droit sur notre nouvel objet « Mobility Config Ext » et choisir « View Model Details » ;

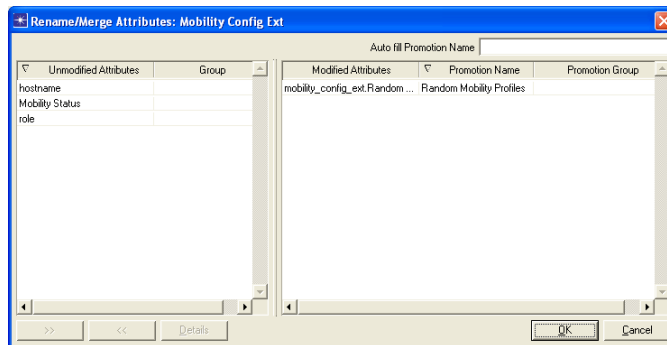


FIGURE A.31 – Boîte de dialogue Rename/Merge Attributes : Mobility Config Ext

5. constater l’affichage de la boîte de dialogue « Node Model Description : Mobility Config Ext » ;
6. cliquer le bouton « Edit... » ;
7. constater que l’affichage de la boîte de dialogue « Edit Derived Model : Mobility Config Ext », voir Fig. A.20 ;
8. cliquer sur l’attribut « Random Mobility Profiles » situé dans le tableau « Attributes » au bas de la boîte de dialogue ;
9. cliquer le bouton « Rename/Merge Attributes... » ;
10. constater l’affichage de la boîte de dialogue « Rename/Merge Attributes : Mobility Config Ext », voir Fig. A.31 ;
11. cliquer dans le tableau de gauche, dans la colonne intitulée « Promotion Name », le nom « Random Mobility Profiles » et le renommer en « Mobility Profiles » ;
12. cliquer le bouton « OK » pour fermer cette boîte de dialogue ;
13. cliquer dans la colonne « Status » de la ligne de l’attribut « Mobility Profiles » afin de le remettre sur « set » plutôt que « promoted » qui signifie que la valeur peut être entrée par l’utilisateur dans les paramètres de la simulation ;
14. cliquer le bouton « Save... » ;
15. constater l’affichage de la boîte de dialogue standard « Save As » ;
16. se déplacer jusqu’au dossier suivant :

D:\Program Files\OPNET\16.0.A\models\std\wireless

- où D représente la lettre du lecteur sur lequel est installé *Opnet Modeler* ;
17. nommer le fichier « Mobility Config Ext » et cliquer « OK » : pas de demande de confirmation de remplacement dudit fichier (...);
 18. cliquer le bouton « Close » afin de fermer la boîte de dialogue « Edit Derived Model : Mobility Config Ext » ;
 19. cliquer le bouton « Close » afin de fermer la palette d'objets ;

NOTA BENE

Cette dernière modification nous l'avons passée alors que nous avons déjà plusieurs centaines de scénarios qui comportaient des instances ce nouvel objet. Et, en vertu de la NOTE IMPORTANTE en fin de section A.10.1, nous avons perdu le paramétrage de de cet objet (central dans nos modèles) dans tous nos scénarios ! D'où l'intérêt d'avoir des *sauvegardes quotidiennes* à jour de notre travail!! Nous avons rapidement descendu la sauvegarde de la veille et avons abandonné cette modification trop tardive.

A.10.3 Création, suppression

La création d'un attribut suit les étapes suivantes :

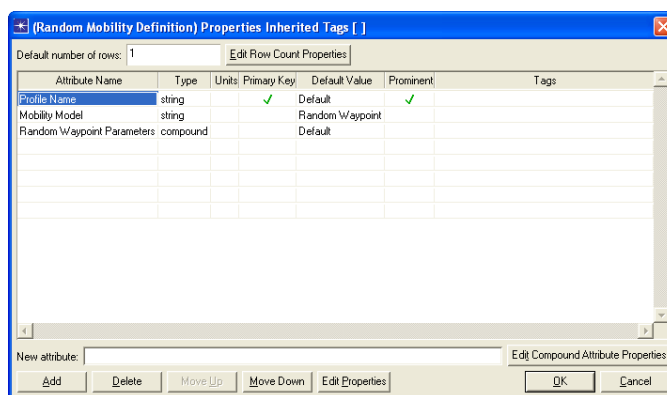


FIGURE A.32 – Boîte de dialogue Random Mobility Definition Properties

1. cliquer « Topology | Open Objet Palette » ;

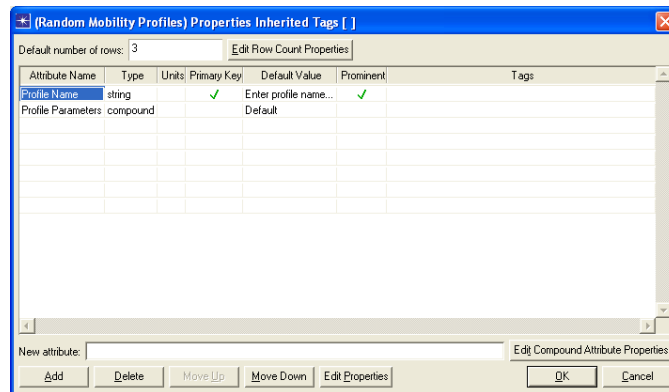


FIGURE A.33 – Boîte de dialogue Random Mobility Definition Properties

2. constater l’affichage de la palette d’objets ;
3. se déplacer jusqu’à la branche « MANET_EXT » et l’ouvrir puis ouvrir « Node Models » ;
4. cliquer droit sur notre nouvel objet « Mobility Config Ext » et choisir « View Model Details » ;
5. constater l’affichage de la boîte de dialogue « Node Model Description : Mobility Config Ext » ;
6. cliquer le bouton « View Parent » en haut à droite ;
7. cliquer le bouton « Edit... » ;
8. constater l’affichage de la fenêtre « Node Model : mobility_config_ext », voir Fig. A.25 ;
9. double-cliquer sur l’unique nœud de la zone cliente de la fenêtre ;
10. constater l’affichage de la fenêtre « Process Model : mobility_config_ext », voir Fig. A.26 ;
11. cliquer dans le menu « Interfaces | Model Attributes »
12. constater l’affichage de la boîte de dialogue « Model Attributes : mobility_config_ext », voir Fig. A.28.
13. cliquer l’attribut « Random Mobility Profiles » ;
14. cliquer le bouton « Edit properties » ;

15. constater l'affichage de la boîte de dialogue « (Random Mobility Profiles) Properties Inherited Tag[] », voir la Fig. A.32.

Observons la boîte de dialogue « (Random Mobility Profiles) Properties Inherited Tag[] », Fig. A.32 et formulons les remarques suivantes :

- l'attribut « Profile Name » est à sa place : c'est bien à ce niveau qu'il faut paramétrer le nom que l'on donne au modèle de mobilité en cours de paramétrage ;
- l'attribut « Random Waypoint Parameters » doit être renommé en « Profile Parameters » : en effet, nous avons plus d'un modèle de mobilité à proposer maintenant ; à noter que nous savons déjà réaliser une modification d'attribut, voir la section A.10.2 ;
- en conséquence, l'attribut « Mobility Model » n'a plus sa place à ce niveau mais au niveau immédiatement inférieur dans l'attribut « Profile Parameters » ;
- sélectionner cette attribut ;
- cliquer le bouton « Delete » ;

Afin d'introduire l'attribut « Mobility Model » à sa bonne place à l'intérieur des « Random Waypoint Parameter », nous créons ce nouvel attribut.

Les développements concernant le code proto-C sont hors du cadre de cette thèse.

A.11 Attribut étendu et XML ou limites de l'outil

Le but de cette section est de donner un exemple concret des limites d'un outil de simulation et de la confiance qu'on peut lui attribuer, notamment pour tout ce qui est relatif aux fonctionnalités peu usitées.

Dans les réseaux denses, 40 nœuds, en OLSR, scénarios 055_XXX à 108_XXX, et en AODV, 163_XXX à 216_XXX, soit donc 108 scénarios au total, nous découvrons à une phase très avancée de notre thèse, les 2 problèmes suivants.

Premier problème, les points d'attraction ont disparu. *Second problème*, l'attribut "Mobility Profile Name" des objets "station_manet_social" n'a plus sa liste déroulante empêchant de changer de modèle de mobilité pour les stations de ces scénarios.

Par exemple, afin de poursuivre nos investigations, nous devons changer environ 15 % des stations des scénarios (en les dupliquant bien sûr) pour les remettre en "Manhattan" de

base et relever l'influence que ces stations ont sur l'ensemble des performances du réseaux selon le modèle de mobilité social choisi : *SMN*, *WMN* ou *AMN*.

Après analyse, nous remontons à la source.

Afin de passer du réseau que nous qualifions de « normal », c'est à dire qui comporte 20 stations, au réseau que nous qualifions de « dense », c'est à dire qui comporte 40 stations, l'idée d'exporter les scénarios *réseau normal* en XML, les dupliquer et les transformer en *réseau dense*, dans ce langage de marquage, puis les importer, nous avait séduits. Après moult essais et nous être bien documentés, sans outil XML professionnel tel que *Altova Mapforce*, la charge de travail est aussi importante que de dupliquer directement les scénarios dans *Opnet Modeler*. Nous abandonnons donc cette voie et décidons de faire évoluer après duplication le premier des 54 scénarios *réseau normal*, puis de dupliquer celui-ci 53 fois afin d'en décliner l'ensemble des scénarios *réseau dense*.

Ce faisant, nous omettons, c'est l'origine du *premier problème*, mais là c'est bien notre faute, de déposer de nouveau, les points d'attractions à leur place. Nous ne nous en apercevons pas de suite. Mais, dès lors que nous en prenons conscience, nous entreprenons de suite la dépose des points d'attraction omis et cela nous coûte une bonne journée de travail d'intervenir dans les 90 scénarios concernés (108 - 18 Manhattan de base qui n'ont pas de point d'attraction).

Mais, c'est l'origine du *second problème*, et là nous ne sommes pas en cause, à notre insu, la fonctionnalité d'export/import XML, nous fait perdre 2 éléments :

- le cadre Mobility Domain ;
- l'éditeur d'attribut associé à "Mobility Profile Name" qui est `mobility_domain_name.ad.m` qui fait apparaitre la liste déroulante en question.

Le Mobility Domain étant notre façon de représenter le cadre de la zone de simulation, nous remarquons de suite la chose et y remédions en déposant de nouveau, cet objet, avant duplication des 53 autres scénarios.

En revanche, l'éditeur d'attribut nous échappe, loin de penser que l'import/export XML pourrait commettre une telle bévue!!

Afin de restituer cette liste déroulante, dans l'après coup, sur toutes les stations d'un

scénario, nous déroulons la procédure qui suit.

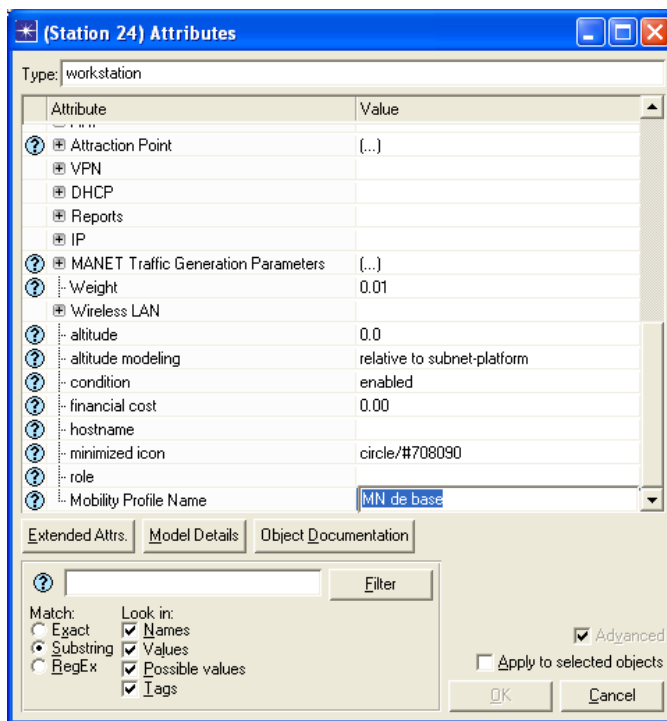
Nous cliquons une station : mettons la Station 24.

Nous cliquons droit puis "Select Similar Node".

Nous constatons que toutes les stations similaires sont sélectionnées.

Nous cliquons droit puis "Edit Attributes (advanced)".

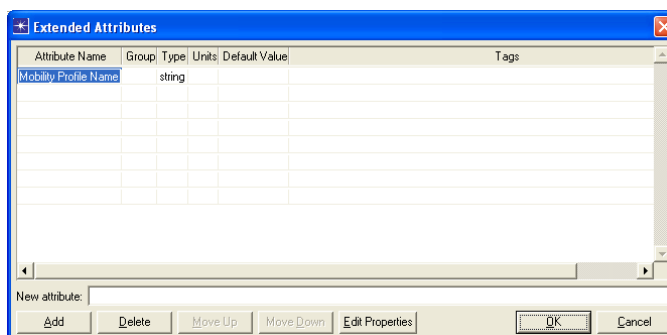
Nous constatons l'affichage de la boîte de dialogue : "(Station 24) Attributes" :



Nous cochons la case : Apply to selected objects.

Nous cliquons le bouton : "Extended Attrs.".

Nous constatons l'affichage de la boîte de dialogue : "Extended Attributes" :

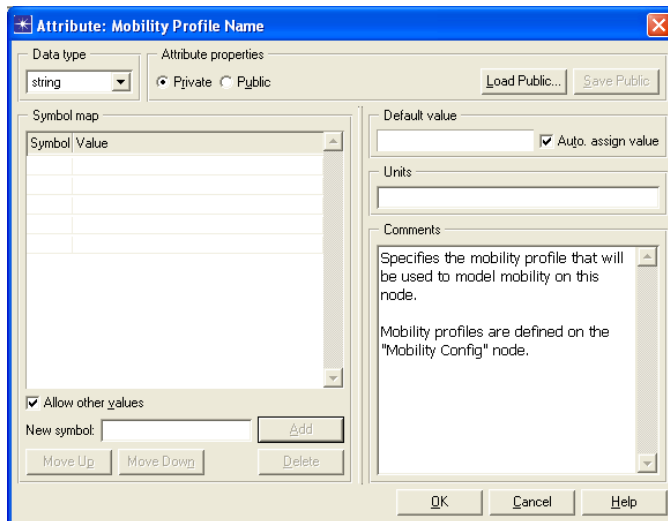


ANNEXE A. CHOIX RELATIFS À LA SIMULATION ET IMPLÉMENTATION DE NOS MODÈLES

Nous constatons que "Mobility Profile Name" est sélectionné ou le sélectionnons dans le cas contraire.

Nous cliquons le bouton : Edit Properties.

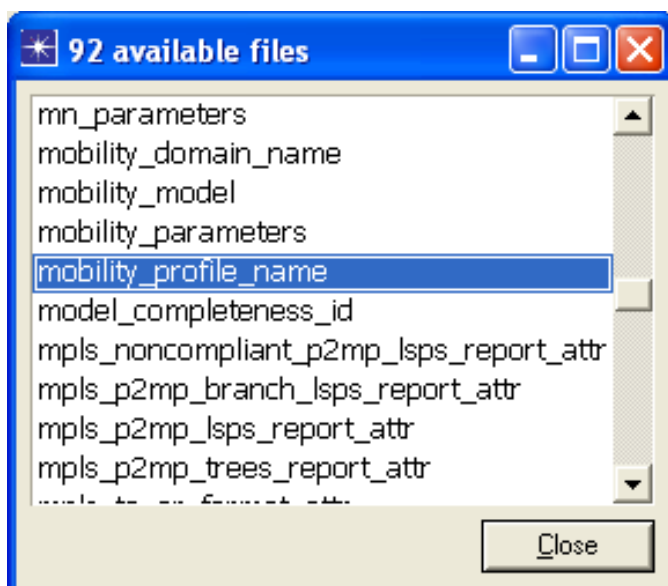
Nous constatons l'affichage de la boîte de dialogue : "Attribute : Mobility Profile Name" :



Nous cliquons la case : "Public" qui devient "Public unnamed".

Nous cliquons le bouton : "Load Public...".

Nous constatons l'affichage de la boîte de dialogue : 92 available files :



Nous cliquons la ligne : `mobility_profile_name`.

Nous constatons la fermeture de la boîte de dialogue : 92 available files.

Nous constatons le retour sur la boîte de dialogue : "Attribute : Mobility Profile Name".

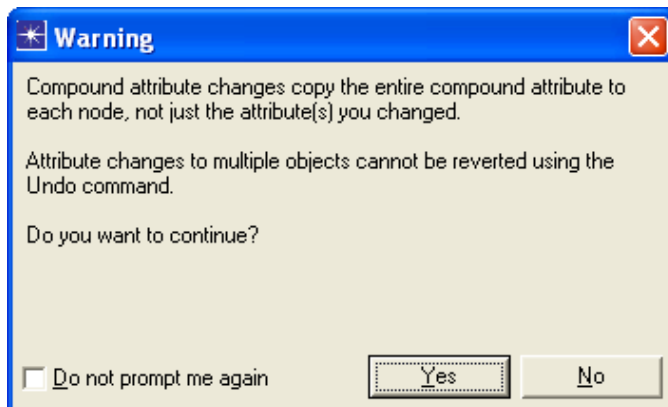
Nous constatons que la case à cocher "Public unnamed" est devenue "Public mobility_profile_name".

Nous cliquons OK pour fermer la boîte de dialogue : "Attribute : Mobility Profile Name".

Nous cliquons OK pour fermer la boîte de dialogue : "Extended Attributes".

Nous cliquons OK pour fermer la boîte de dialogue : "(Station 24) Attributes".

Nous constatons l'affichage de la boîte de dialogue Warning :



Nous cliquons Yes.

Nous rejouons donc cette procédure 108 fois !

C'est dire combien un petit dysfonctionnement d'*Opnet Modeler* non relevé et corrigé au départ peut prendre de l'ampleur avec l'avancement de la thèse : cela nous coûte 2 journées de travail, un travail fort peu passionnant, autant le dire !

Il s'agit d'un exemple assez intéressant pour avoir sa place dans notre thèse, car il est très représentatif des comportements surprenants, pourquoi ne pas dire des bogues, des outils utilisés qui nous causent des pertes de temps qui suivent un accroissement exponentiel avec l'avancement de la thèse.

Annexe B

Implémentation d'OLSR-S

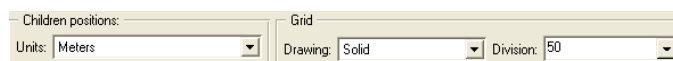
Nous comparons notre approche avec celle par défaut en utilisant dans les 2 cas notre modèle de mobilité *Ant Manhattan Mobility Model (AMN)*. Nous utilisons l'unique statistique suivante :

- *Node statistics, OLSR, MPR Status* qui relate l'attribut « *est MPR* » du nœud considéré.

Nous déposons une instance de notre composant *Mobility Config Ext* que nous paramétrons comme suit :

Attribute	Value
name	Mobility Config Ext Instance
model	Mobility Config Ext
Random Mobility Profiles	(...)
Number of Rows	1
Ant Social Manhattan de base	
Profile Name	Ant Social Manhattan de base
Profile Parameters	(...)
Mobility Model	Ant Social Manhattan
Ant MN Parameters	(...)
Mobility Domain Name	Not Used
x_min (meters)	0.0
y_min (meters)	0.0
x_max (meters)	1,000
y_max (meters)	1,000
Speed (meters/seconds)	uniform_int (0, 3)
Pause Time (seconds)	constant (1)
Start Time (seconds)	constant (10)
Stop Time (seconds)	End of Simulation
Animation Update Frequency (...)	1.0
Record Trajectory	Disabled
Safety Distance (meters)	30
Move Step (meters)	25
Block Length (meters)	50
Maximum Acceptable Factor	(...)
Function Type	Constant Function
Constant Function Paramet...	(...)
Parabola Function Paramet...	(...)
Parabola/Constant Parame...	(...)
Pheromone	(...)
Decay Factor	0.0
Distance Exponent	1.0

Dans View | Background | Set properties, nous paramétrons ceci :

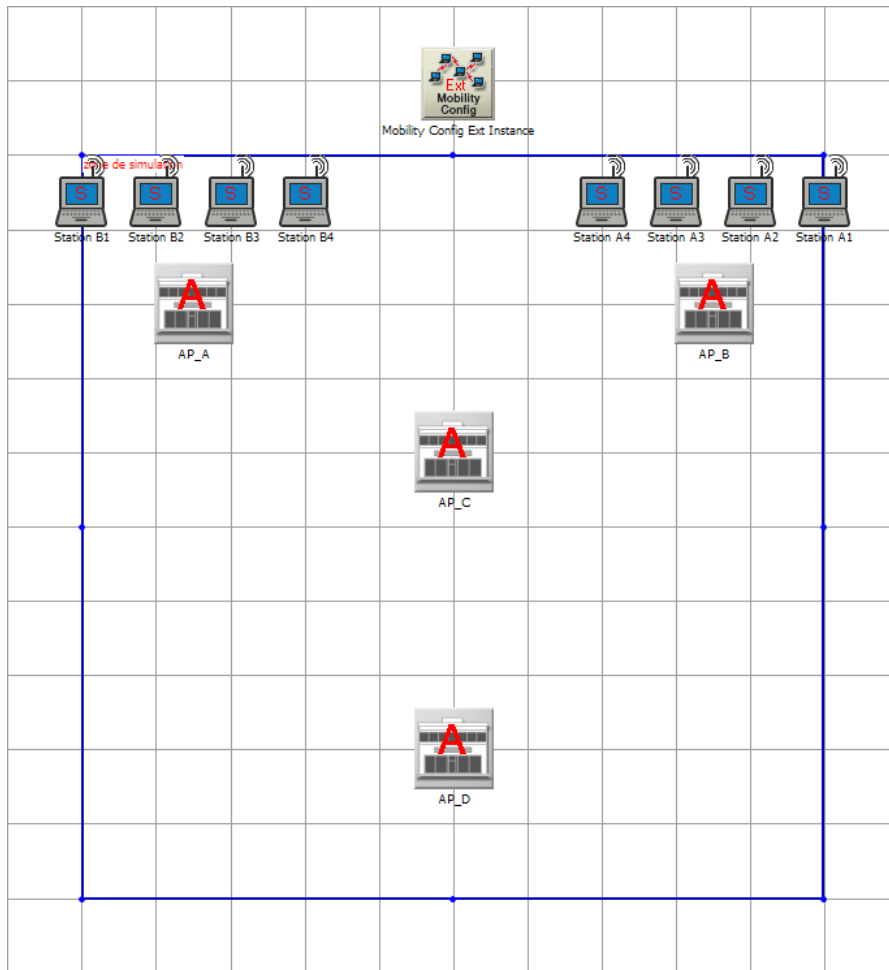


Nous déposons :

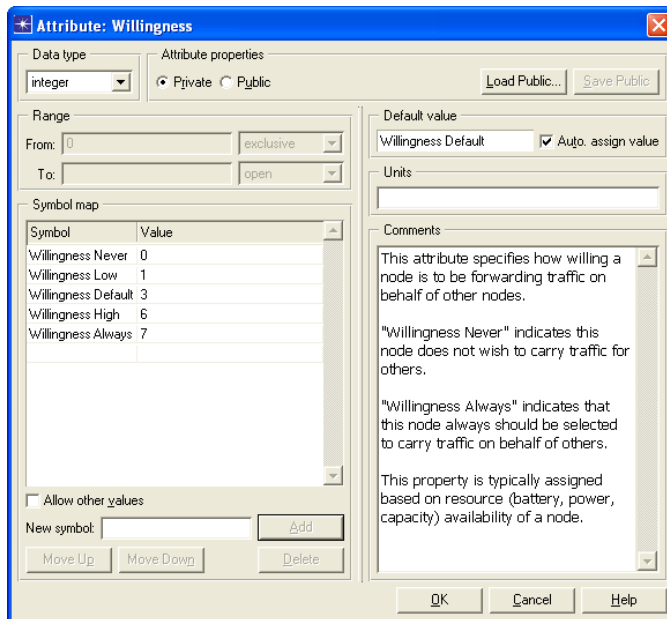
- 4 points d'attractions : AP_a où $a = A, B, C, D$;
- 4 machines nommées "Station An" où $n = 1, 2, 3, 4$ pour le groupe A ;
- 4 machines nommées "Station Bn" où $n = 1, 2, 3, 4$ pour le groupe B.

Cela nous donne le réseau suivant¹ :

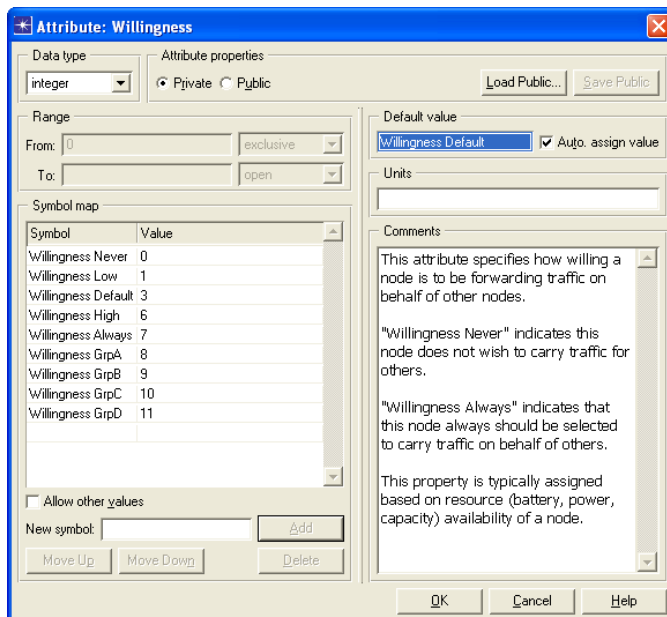
1. nous avons choisi la couleur blanche dans Edit | Preference | UI | Background color



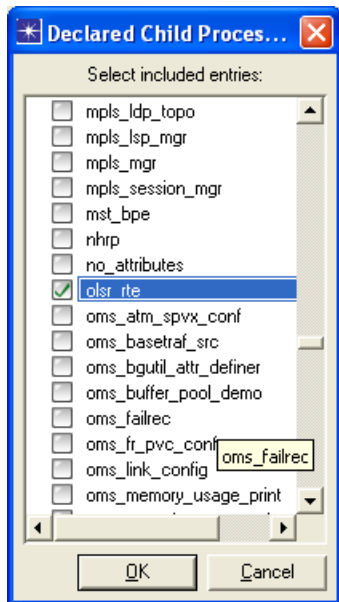
Afin d'ajouter nos nouvelles valeurs de *Willingness*, nous déroulons la procédure décrite à l'annexeA.10.3 en partant d'un de nos composants déjà déposés. Il peut s'agir aussi bien de notre *manet_station_social* que de notre composant *manet_station_attraction_point*. En effet, tous les deux ont même parent : *manet_station_adv_ext*. Or, c'est en éditant le modèle de nœuds de ce dernier que l'on trouve le nœud *manet_rte_mgr*. C'est en double-cliquant sur celui-ci qu'on ouvre son modèle de processus. Arrivé là, on clique Interfaces | Model Attributs puis en sélectionnant OLSR Parameters, on clique sur Edit Properties. Enfin, en sélectionnant Willingness, on clique de nouveau sur Edit Properties qui ouvre la boîte de dialogue suivante :



qui devient suite à nos ajouts, ceci :



Après une série (importante!) de OK, nous retrouvons le modèle de processus *manet_rte_mgr.pr.m*. Nous cliquons sur File | Declare Child Process Models... pour constater que *olsr_rte.pr.m* est bien processus fils de *manet_rte_mgr.pr.m*, comme il apparaît ci-dessous :



Nous ouvrons donc ce modèle de processus et découvrons son code.

Nous en établissons rapidement l'architecture :

- le processus s'initialise ;
- il passe en sommeil ;
- il est réveillé par l'un des événements suivants :
 - arrivée d'un paquet ;
 - échéance d'un temporisateur.

À l'arrivée d'un paquet, la fonction *olsr_rte_pkt_arrival_handle* est appelée. Elle appelle à son tour, suivant le type de message arrivé, *olsr_rte_neighborhood_topology_check_hello* ou *olsr_rte_neighborhood_topology_check_tc* lesquels appellent *olsr_rte_neighborhood_topology_check*.

De même à l'échéance d'un temporisateur, les fonctions *olsr_hello_processing_expiry_handle* ou *olsr_rte_symmetric_link_expiry_handle* sont appelées lesquelles appellent à leur tour *olsr_rte_neighborhood_topology_check*.

Ensuite, quelque soit sa fonction appelante, *olsr_rte_neighborhood_topology_check* appelle *olsr_rte_calculate_mpr_set* qui est l'algorithme de choix des *MPR*, le cœur d'*OLSR*, comme nous le dénommons au début de notre propos.

En résumé, à chaque changement de voisinage² ou de topologie³, le calcul⁴ de l'ensemble des *MPR* est lancé.

Le flot de contrôle de *olsr_rte_calculate_mpr_set* est le suivant :

```
// Construire l'ensemble des nœuds à 2 sauts
olsr_rte_strict_two_hop_set_create

// Déterminer le degré de centralité5 de chaque nœud à 1 saut
olsr_rte_calculate_degree

// Déclarer MPR tous les nœuds à 1 saut passage obligatoire vers un nœud à 2 sauts
olsr_rte_mpr_with_nbr_count_one_add

// TANT QUE tous les nœuds à 2 sauts n'ont pas leur MPR
while (num_strict_two_hops > 0)
    // Déterminer la portée6 des nœuds à 1 saut
    olsr_rte_calculate_reachability

    // POUR tous les nœuds à 1 saut
    for (i=0 ; i < num_keys ; i++)
        // SI nœud à 1 saut courant a un lien non symétrique OU
        // a son Willingness à Will_never OU
        // est déjà MPR (OLSRC_VALUE_MPR=-1) ALORS
        if ((nbr_set_entry_ptr->status == OLSRC_NOT_SYM_STATUS) ||
            (nbr_set_entry_ptr->willingness == OLSRC_WILL_NEVER) ||
            (nbr_set_entry_ptr->reachability == OLSRC_VALUE_MPR))
            // Passer au nœud à 1 saut suivant
            continue
```

2. apparition/disparition d'un nœud à 1 ou 2 sauts, ou bien nouveau lien asymétrique. Apparition \iff lien devient symétrique. Disparition \iff plus de lien du tout. Lien asymétrique \iff nœud potentiellement nœud à 1 saut.

3. création/suppression/expiration d'interfaces principales ou secondaires de nœud, consécutives à l'arrivée d'un *Topologie Control Message* ou message *TC* ou à l'échéance d'un temporisateur d'expiration de validité d'interface.

4. pour traduire au plus près du nom de fonction mais il s'agit plutôt d'une mise à jour

5. nous nommons degré de centralité ce qu'*Opnet* dénomme degree car il s'agit en fait du nombre de lien symétrique dont dispose le nœud.

6. nous traduisons *reachability* par portée car il s'agit en fait du nombre de nœuds à 2 sauts qu'on ne peut atteindre que par ce nœud à 1 saut. À noter, *Opnet* n'a pas hésité à utiliser un substantif exactement inverse à celui dont il s'agit...

```

// FIN SI
// SI nœud à 1 saut courant a meilleur7 Willingness ALORS
if (nbr_set_entry_ptr->willingness > willingness)
    // Le choisir comme candidat MPR
    mpr_candidate_ptr = nbr_set_entry_ptr...
// SINON SI nœud à 1 saut courant a même Willigness
// et meilleur Reachability ALORS
else if (... && nbr_set_entry_ptr->reachability > reachability))
    // Le choisir comme candidat MPR
    mpr_candidate_ptr = nbr_set_entry_ptr...
// SINON SI nœud à 1 saut courant a même Willigness
// et Reachability et meilleur Degree ALORS
else if (... && nbr_set_entry_ptr->degree) > degree))
    // Le choisir comme candidat MPR
    mpr_candidate_ptr = nbr_set_entry_ptr...
// FIN SI
// FIN POUR
// Déclarer ce nœud MPR
mpr_candidate_ptr->reachability = OLSRC_VALUE_MPR;
// Ré-évaluer les nœuds à 2 sauts restants
olsr_rte_update_strict_two_hop_set(&num_strict_two_hops, mpr_candidate_ptr)
// FIN TANT QUE
/* Optimization of MPR set should happen here. */
/* Not required if all the nodes in the network */
/* have same Willingness. MPR calculation */
/* itself gives optimized set. More optimization*/
/* will only be helpful when nodes in network */
/* have different level of WILLINGNESS */
/* Extra optimization currently not implemented.*/

```

7. meilleur que les nœuds à 1 saut qui l'ont précédé dans la boucle POUR.

```
// Construire l'ensemble des nœuds à 2 sauts
olsr_rte_build_new_mpr_set
/* Note: Here we are not doing an explicit check */
/* if the new MPR set is different from the */
/* previous one. RFC says : "if different then send */
/* out an Hello message to notify the changes". */
/* We will wait for next scheduled hello rather */
/* than explicit hello message. Lazy method. */
```

Nous mentionnons ci-dessus 2 commentaires d'*Opnet*. Quoique le second soit mentionné à titre humoristique (lazy method...), le premier est pertinent dans notre discussion. En effet, nous aurions bien profité d'une optimisation concernant Willingness. Mais, nous n'en disposons pas.

Quoiqu'il en soit, nous avons trouvé le lieu où Willingness est testé. Nous pouvons ici ajouter notre code. Nous remplaçons donc ceci :

```
// SI nœud à 1 saut courant a meilleur Willingness ALORS
if (nbr_set_entry_ptr->willingness > willingness)
```

par cela :

```
// Notons N : le nœud qui a lancé l'algorithme MPR
// SI N a son Willingness > Will_always ALORS
if (node_willingness > OLSRC_WILL_ALWAYS)
    // SI nœud à 1 saut courant a même Willingness que N ALORS
    if (nbr_set_entry_ptr->willingness == node_willingness)
        // Le choisir comme candidat MPR
        mpr_candidate_ptr = nbr_set_entry_ptr...
// SINON SI nœud à 1 saut courant a meilleur Willingness ALORS
if (nbr_set_entry_ptr->willingness > willingness)
etc.
```

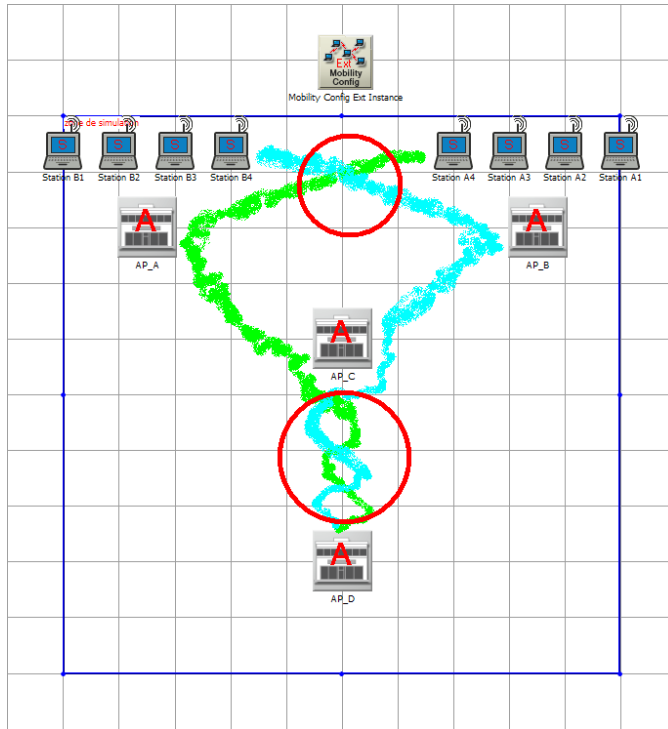
Notons la concision de l'intervention dans le code *Opnet* dont voici les lignes extraites sans aucune intervention supplémentaire :

```

/*****
/* OLSR SOCIAL --- DEBUT
/*
/* Remplacer ceci :
/*
/* if (nbr_set_entry_ptr->willingness > willingness)
/*
/* par cela :
/*
if (node_willingness > OLSRC_WILL_ALWAYS)
{
if (nbr_set_entry_ptr->willingness == node_willingness)
{
mpr_candidate_ptr = nbr_set_entry_ptr;
willingness = nbr_set_entry_ptr->willingness;
reachability = nbr_set_entry_ptr->reachability;
degree = nbr_set_entry_ptr->degree;
}
}
else if (nbr_set_entry_ptr->willingness > willingness)
/*
/* OLSR SOCIAL --- FIN
/*****

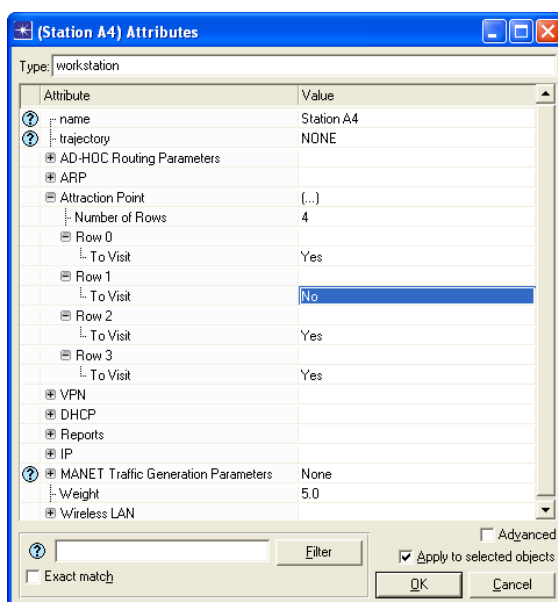
```

Ci-après les trajectoires des groupes *A* et *B* :

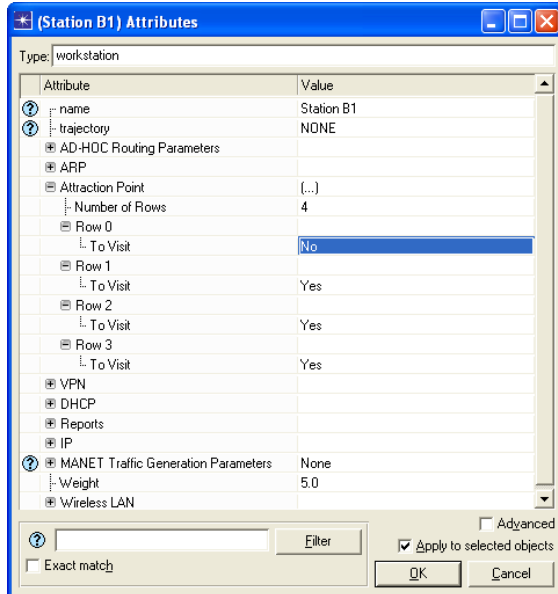


Les lieux où la probabilité est grande pour un nœud de changer ses *MPR* sont représentés par 2 cercles.

Afin de forcer les groupes à suivre ces trajectoires, il nous suffit de sélectionner les points d'attraction à visiter. Pour le groupe *A* voici la sélection réalisée dans les stations :

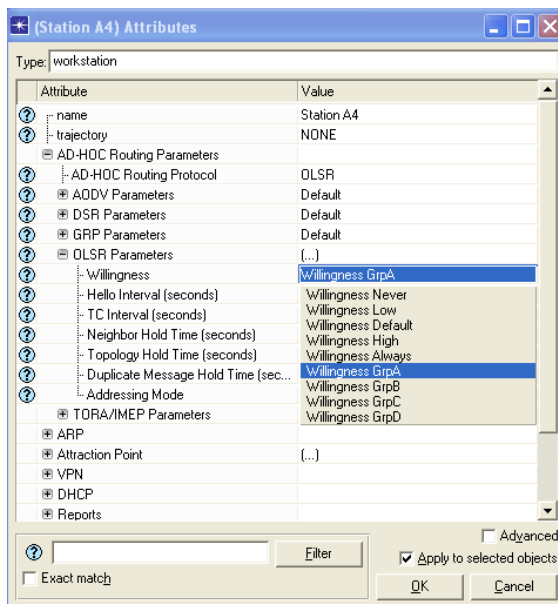


et voici celle du groupe *B* :

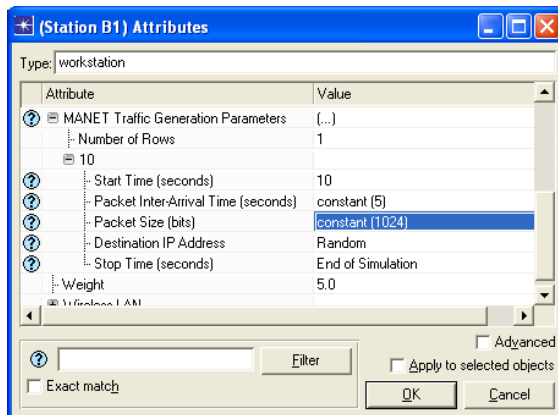


Nous réalisons 2 scénarios :

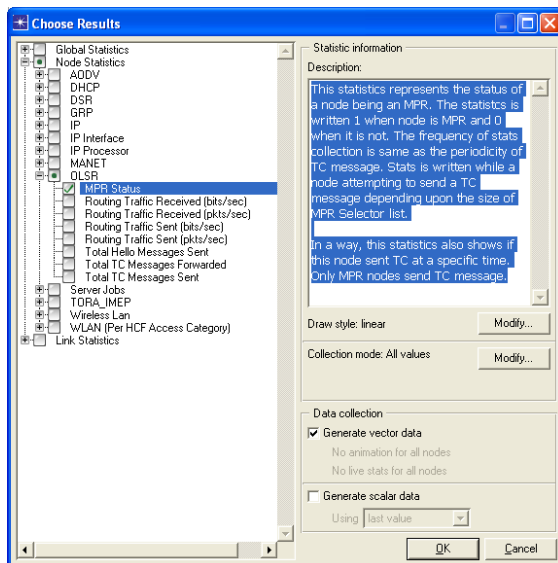
- « Sans OLSR Social » laisse à Will_default le Willingness ;
- « Avec OLSR Social » initialise le Willingness à :
 - Willingness_grpA pour le groupe *A* ;
 - Willingness_grpB pour le groupe *B*.



Il nous faut générer un trafic d'informations. Nous choisissons d'avoir des stations uniformes d'un point de vue fonctionnement (iso-fonctionnelles). Nous paramétrons donc le générateur de trafic uniformément sur toutes les stations comme suit :



Enfin, nous devons choisir nos statistiques *Opnet*. Nous optons pour *OLSR Status* dont voici la rubrique dans l'aide en ligne :



« This statistics represents the status of a node being an MPR. The statistics is written 1 when node is MPR and 0 when it is not. The frequency of stats collection is same as the periodicity of TC message. Stats is written while a node attempting to send a TC message depending upon the size of MPR Selector list. In a way, this statistics also shows if this node sent TC at a specific time. Only MPR nodes send TC message ».

Résumé :

À l'aire du *Multimedia Mobile*, l'essor des réseaux sans fil est fulgurant et la mobilité est devenue un sujet primordial exacerbé par l'augmentation significative du nombre d'utilisateurs mobiles.

Un nœud évoluant dans un *réseau mobile de base* se comporte de la même manière qu'un aveugle évoluant dans notre univers en élaborant sa propre représentation à l'aide de son bâton, mécanisme dénommé dans la littérature : *mobilité terminale*. Pour réduire cet aveuglement, plusieurs méthodes ont été élaborées qui prennent appui sur les services de localisation ou les modèles de mobilité.

Un modèle de mobilité est donc destiné à décrire en termes d'environnement, le mode de circulation des nœuds mobiles avec pour défi de trouver des modèles fidèles aux comportements des utilisateurs. Les modèles aléatoires sont biaisés car les appareils mobiles sont portés par des êtres sociaux. Cela nous a amené à inclure des éléments sociaux dans notre modélisation.

Nous présentons les modèles de mobilité existants et les classifions. Nous définissons nos modèles, les implémentons et mesurons leur impact sur le test des réseaux.

Enfin nous élargissons notre spectre en montrant comment le fait d'octroyer la perception de regroupement social à un protocole de routage réseau, peut en améliorer les performances.

Mots clés :

Réseaux sans fil, modèle de mobilité pour l'évaluation des performances, protocoles de routage.

Abstract :

In our area of *Mobile Multimedia*, the expansion of wireless networks is dazzling and mobility has become a major issue exacerbated by the significant increase in the number of mobile users.

A node operating in a *basic mobile network* behaves the same way a blind person moving in our universe by developing its own representation with his stick, a mechanism known in the literature as *terminal mobility*. To reduce this blindness, several methods have been developed that are based on location services and mobility models.

A mobility model is then intended to describe in terms of environment, the motion criteria of mobile nodes with the challenge to find models faithful to user behavior. Random models are biased because mobile devices are supported by social beings. This led us to include social elements in our models.

We present the existing mobility models and classify them. We define our models, implement them and measure their impact on the network testing.

Finally we expand our spectrum by showing that granting social grouping perception to a network routing protocol, can improve its performance.

Keywords :

Wireless networks, mobility model for performance evaluation, routing protocols.