

UNIVERSITÉ DU QUÉBEC À MONTRÉAL

UN SYSTÈME DE SURVEILLANCE DE MALADIES INFECTIEUSES  
UTILISANT LA TECHNOLOGIE SANS FILS ET RÉSEAU DE CAPTEURS

MÉMOIRE  
PRÉSENTÉ  
COMME EXIGENCE PARTIELLE  
DE LA MAÎTRISE EN INFORMATIQUE

PAR  
JEAN-BERNARD ORIVAL

FÉVRIER 2017

UNIVERSITÉ DU QUÉBEC À MONTRÉAL  
Service des bibliothèques

Avertissement

La diffusion de ce mémoire se fait dans le respect des droits de son auteur, qui a signé le formulaire *Autorisation de reproduire et de diffuser un travail de recherche de cycles supérieurs* (SDU-522 – Rév.10-2015). Cette autorisation stipule que «conformément à l'article 11 du Règlement no 8 des études de cycles supérieurs, [l'auteur] concède à l'Université du Québec à Montréal une licence non exclusive d'utilisation et de publication de la totalité ou d'une partie importante de [son] travail de recherche pour des fins pédagogiques et non commerciales. Plus précisément, [l'auteur] autorise l'Université du Québec à Montréal à reproduire, diffuser, prêter, distribuer ou vendre des copies de [son] travail de recherche à des fins non commerciales sur quelque support que ce soit, y compris l'Internet. Cette licence et cette autorisation n'entraînent pas une renonciation de [la] part [de l'auteur] à [ses] droits moraux ni à [ses] droits de propriété intellectuelle. Sauf entente contraire, [l'auteur] conserve la liberté de diffuser et de commercialiser ou non ce travail dont [il] possède un exemplaire.»

## REMERCIEMENTS

Je tiens à exprimer toute ma gratitude à toutes les personnes qui ont contribué à la réussite de mon cursus universitaire.

Tout d'abord, j'aimerais remercier la famille Sheehy, qui m'a financièrement aidé dans la réalisation de mes études de deuxième cycle. J'adresse également un remerciement spécial à la famille Filion, Sr. Marina, Mgr. Wildor Pierre qui ont cru en mon potentiel.

Je tiens aussi à remercier mon directeur de recherche, M. Abdelattif Obaid, pour son soutien, ses suggestions et ses commentaires les plus constructifs. Sa rigueur dans sa méthode d'encadrement et sa confiance en moi m'ont toujours incité à dépasser mes limites ainsi qu'à fournir le meilleur de moi-même. Je l'ai eu en tant que professeur en Haïti et m'a reçu en tant que stagiaire à l'UQAM et suivi mes pas et motivé pour que je continue dans la cadre de cette Maîtrise.

Mes remerciements les plus sincères vont à mon codirecteur M. Abdoulaye Baniré Diallo, par son expertise en Bioinformatique, il a grandement contribué à la réalisation de ce mémoire. Il s'est toujours mis à ma disposition et a su me guider tout au long de mes activités de recherche.

Un merci bien spécial à mes parents, ma mère Destinette Orival Jean-Louis et mon père Sauveur Orival. En dépit de la distance, ils n'ont jamais cessé de me soutenir tout au long de mon cheminement. Sans leur précieux support, leurs encouragements et leur inspiration, je ne serais probablement pas là où j'en suis aujourd'hui. Un merci spécial à mon frère Patrick Orival.

Je voudrais aussi remercier Manon Demers wice, M. Rony et toute sa famille qui ont su guider mes premiers pas à Montréal.

Je remercie aussi tous les étudiants du laboratoire TRIME pour leur aide et leur sympathie, spécialement Fabio, Anne-Marie, Faten, Imane.

Merci encore à tous.

## TABLE DES MATIÈRES

LISTE DES FIGURES.....	IX
RÉSUMÉ.....	XV
CHAPITRE	
INTRODUCTION.....	1
1.1 Introduction générale .....	1
1.2 Problématiques.....	3
1.2.1 La prédiction de l'impact réel de la maladie sur la population .....	3
1.2.2 Réduire la complexité des analyses et des calculs mathématiques. ....	3
1.2.3 Comprendre le déplacement des humains et trouver une approche similaire à leur déplacement.....	4
1.2.4 La surveillance de transition d'états de la maladie dans la population .....	4
1.3 Portée et Limite du projet.....	5
1.4 Objectifs .....	6
1.5 Méthodologie .....	6
1.6 Organisation du mémoire.....	7
CHAPITRE II	
REVUE DE LITTÉRATURE .....	9
2.1 Les maladies infectieuses.....	9
2.1.1 Propagation des maladies .....	11
2.1.2 Comparaison entre les modèles stochastiques et les modèles déterministes.....	13
2.1.3 Les modèles stochastiques .....	14
2.1.4 Modèle déterministe (Les modèles à compartiment : approche mathématique).....	17
2.1.5 Synthèse .....	22
2.1.6 Aspects importants dans la simulation d'une maladie infectieuse.....	23
2.2 Les Modèles de mobilités dans les réseaux MANET .....	24

2.2.1	Les modèles de mobilité.....	25
2.2.2	Les modèles de mobilité d'entité .....	26
2.2.3	Les modèles de mobilité de groupe.....	29
2.2.4	Classification des modèles de mobilité .....	30
2.2.5	Limitations des modèles de mobilité aléatoire .....	31
2.3	Crowdsourcing et crowdsensing .....	34
2.3.1	Typologie de crowdsourcing (approche).....	35
2.3.2	catégories de crowdsourcing mobiles .....	37
2.3.3	architecture de crowdsourcing .....	38
2.3.4	Le crowdsourcing dans le domaine médical .....	39
CHAPITRE III		
MISES EN ŒUVRE ET SIMULATION .....		
3.1	Méthodologie .....	41
3.1.1	Les maladies infectieuses simulées dans le cadre de notre projet.....	41
3.1.2	Gestion de la transmission de la maladie dans notre simulation .....	42
3.1.3	Particularité des maladies simulées.....	43
3.2	Simulation .....	45
3.3	Architecture de la simulation .....	45
3.4	Générateur et simulateur de modèle de mobilité.....	47
3.4.1	BonnMotion .....	48
3.4.2	Ns2 .....	49
3.4.3	Nam .....	51
3.5	Intégration de données .....	52
3.5.1	La base de données.....	54
3.6	Représentation des données à travers un service web.....	56
3.6.1	Description des différentes fenêtres du service web .....	56
3.6.2	Les différents sous-menus.....	58
CHAPITRE IV		
INTÉGRATION DE CROWDSOURCING ET CROWDSENSING À HEALTH ...		
4.1	Modèle proposé.....	67
4.1.1	Méthodologie .....	69

4.1.2 Simulation <i>HealthSim</i> .....	72
4.1.3 Module <i>crowdsourcing</i> .....	72
4.1.4 Module <i>crowdsensing</i> .....	74
CHAPITRE V .....	81
CONCLUSION .....	81
ANNEXE .....	85
BIBLIOGRAPHIE .....	101





## LISTE DES FIGURES

Figure	Page
Figure 2.1 Répartition des causes de décès selon (World Health Organization, 2008), (Stattner & Vidot, 2011) .....	10
Figure 2.2 Comparaison de la dynamique d'une infection d'un modèle à compartiment par rapport à un modèle réseau (Meyers, 2007) .....	12
Figure 2.3 Représentation du modèle SI (Hethcote, 2007).....	19
Figure 2.4 Représentation du modèle SIR (Hethcote, 2007) .....	20
Figure 2.5 catégories des modèles de mobilité dans les réseaux ad hoc (Bai & Helmy, 2004).....	31
Figure 2.6 Différentes approches de crowdsourcing.....	37
Figure 2.7 Architecture générale de crowdsourcing (Fuchs-kittowski & Faust, 2014) 38	
Figure 2.8 Architecture détaillée d'une application crowdsourcing (Fuchs-kittowski & Faust, 2014).....	39
Figure 3.1 Chronologie de la maladie Ebola dans notre simulation .....	44
Figure 3.2 Architecture de la simulation HealthSim.....	47
Figure 3.3 Génération des fichiers scénario du modèle de mobilité RandomWaypoint .....	48

Figure 3.4	Génération des fichiers scénario du modèle de mobilité Smooth.....	48
Figure 3.5	Commande <i>Interval format</i> .....	49
Figure 3.7	Format du fichier Ns2.....	50
Figure 3.8	Chemin pour téléverser le fichier mouvement ns2 et Informations sur certains paramètres de la simulation .....	50
Figure 3.9	Résultats de la simulation ns2.....	51
Figure 3.10	Visualisation de la propagation de l'épidémie pour les modèles de mobilité <i>RandomWaypoint</i> et <i>ManhattanGrid</i> .....	52
Figure 3.11	Première étape : processus d'insertion des nœuds (individus) dans la base de données.....	53
Figure 3.12	Deuxième étape : processus d'insertion des coordonnées dans la base de données.....	54
Figure 3.13	Troisième étape : processus de traitement et de filtrage de données dans la base .....	54
Figure 3.14	Schéma de la base de données <i>HealthSim</i> .....	55
Figure 3.15	Portail <i>HealthSim</i> .....	57
Figure 3.16	Portail <i>HealthSim</i> suite à une connexion valide. ....	57
Figure 3.17	Menu du système .....	58
Figure 3.18	Modèle de mobilité.....	59
Figure 3.19	Coordonnées .....	59

Figure 3.20	Liste des maladies.....	60
Figure 3.21	Information relative à la maladie Ebola.....	60
Figure 3.22	Déplacement .....	61
Figure 3.23	Statut de santé. ....	62
Figure 3.24	Statut de santé des individus en fonction d'un état de santé particulier.....	62
Figure 3.25	Liste des nœuds pour les différents scénarios.....	63
Figure 3.26	Informations sur les nœuds contaminés.....	64
Figure 3.27	État de santé des nœuds. ....	64
Figure 3.28	Information sur les noeuds infectés au cours d'une contamination.....	65
Figure 3.29	Liste des scénarios utilisant le modèle <i>Random Waypoint</i> .....	66
Figure 4.1	Architecture générale du modèle proposé. ....	68
Figure 4.2	Architecture détaillée du modèle proposé ( <i>Crowdsourcing/Crowdsensing</i> ) .....	70
Figure 4.3	Architecture d'intégration du modèle proposé à Healthsim .....	71



## LISTES DES ABRÉVIATIONS

aGPS : Assisted Global Positionning Service

AMT : Amazon Méchanical Turk

BdC : Base de données Capteurs

BdH : Base de données HealthSim

BDS : Base de données Serveur

BSAMM : Boundless Simulation Area Mobility Model

CMM : Column Mobility Model

CSMM : City Section Mobility Model

DHIT : Distributed Human Intelligence Tasking approach

ECRMM : Exponential Correlated Random Mobility Model

KDM : Knowledge Discovery and Management approach

MANET : Mobile Adhoc Network

MIT : Massachusset Institue of Technology

NCMM : Nomadic Community Mobility Model

OMS : Organisation Mondiale pour la Santé

PCS : Personal communication System

PMM : Purse Mobility Model

PRWMM : Probabilistic Random waypoint mobility

RDMM : Random Direction Mobility Model

RWMM : Random walk mobility model

RWpMM : Random Waypoint Mobility model

SEIR : Succceptible, Exposed, Infected, Recovered

SI : Succceptible, Infected

SIR : Succceptible, Infected, Recovered

WHO : World Health Organization

## RÉSUMÉ

La surveillance des maladies infectieuses est l'un des moyens efficaces permettant de prendre des mesures en vue de limiter l'expansion d'une épidémie.

Des outils technologiques pour collecter des données de santé sur les individus, ainsi que des mécanismes pour capter et gérer ces informations sont essentiels à la surveillance épidémiologique.

Avec l'avènement des capteurs intégrés dans les appareils mobiles intelligents combinés avec les objets connectés à l'Internet, une nouvelle forme de collection de données a pris naissance, dont le *crowdsensing mobile*, qui permet d'utiliser les téléphones munis de capteurs comme sources de données pour accomplir des tâches telles que la collecte d'informations biologiques et médicales.

L'accès aux différents services offerts par ces systèmes de surveillance se fait par l'intermédiaire d'une plateforme Web, communément appelé *crowdsourcing*, à travers laquelle les tâches et les critères de ce processus de collection sont publiés.

En vue de comprendre le phénomène de propagation des maladies infectieuses, nous avons simulé la progression de certaines maladies. Nous avons utilisé des modèles de mobilité employés dans les réseaux ad-hoc comme approche pour simuler le déplacement des individus dans une région.

Les résultats de la simulation sont accessibles via un service web qui permet d'interpréter les données simulées.

En vue de faire une surveillance en temps réel afin de prendre les mesures qui s'imposent, nous proposons une architecture de crowdsourcing/crowdsensing permettant de suivre en temps réel l'évolution d'une épidémie dans une région à l'aide des capteurs de localisation intégrés dans les téléphones et d'autres capteurs médicaux permettant d'avoir des informations sur l'état de santé des individus. Les résultats permettront aux institutions de santé de prendre à temps les mesures préventives nécessaires pour limiter la propagation de la maladie.

**Mots clés :** maladies infectieuses, crowdsourcing, crowdsensing, réseaux MANETS, modèles de mobilité ad-hoc, capteurs médicaux, gestion de propagation de maladies contagieuses.





# CHAPITRE I

## INTRODUCTION

### **1.1 Introduction générale**

Les nouvelles opportunités technologiques apportées par les objets connectés à l'Internet en très grand nombre et l'émergence des appareils mobiles intelligents permettent de contribuer de façon significative à la façon dont les systèmes informatiques offrent leurs services pour résoudre certains problèmes dans divers domaines, tels que la santé, l'environnement, etc. L'avènement de cette nouvelle ère dans l'informatique permet une meilleure gestion des données, la surveillance médicale, l'analyse des interactions des usagers avec les services, etc. Cela a également facilité la surveillance à distance des maladies infectieuses (Liao et al., 2012).

Par exemple, certains systèmes permettent aux utilisateurs de suivre l'état de santé d'un patient en temps réel ou de prendre des mesures préventives pour l'améliorer. D'autres aident à favoriser la désaccoutumance au tabac, à diminuer la douleur ou à améliorer la santé en général (Brabham et al., 2014). Ces exemples utilisent largement ces nouvelles technologies, soit pour l'amélioration, soit pour le contrôle de la santé tant au niveau individuel que global.

En plus, de nombreux travaux sur l'intégration des applications mobiles et les capteurs médicaux permettraient de suivre en temps réel l'évolution de la propagation de certaines maladies infectieuses à caractère épidémique.

L'un des paradigmes qui a été développé ces dernières années grâce à cette évolution technologique est le *crowdsourcing* (Brabham et al., 2014). D'après D. C. Brabham, (2008), le *crowdsourcing* est une approche Web de résolution de problèmes de type ODPS (*online distributed problem-solving*) qui tire parti de l'intelligence collective des communautés en ligne pour résoudre un problème complexe dans un but spécifique. Il peut aider à renforcer les liens entre les organisations, les communautés et les populations en facilitant la résolution de problèmes de manière collaborative (Howe, 2008).

Nous pensons que l'approche *crowdsourcing* peut aider dans la collecte d'informations pour le suivi de l'état de propagation des maladies infectieuses.

La présence de plusieurs capteurs intégrés dans les téléphones intelligents suscite l'intérêt d'un nouveau paradigme qui est le *Crowdsensing*. Il exploite les réseaux de capteurs sans-fil par lequel les données qui proviennent des capteurs sont collectées dans un cadre spécifique tel que les analyses médicales ou le suivi de l'état de santé des usagers (Moo-RyongRa et al., 2012).

Ces paradigmes sont avantageux tant pour les institutions responsables de veiller sur la santé de la population que pour les individus. Ils permettent aux institutions de santé publique de mieux cerner les problèmes en cas d'une propagation infectieuse et offrent la possibilité à chaque personne d'apprendre et de modifier leur comportement en tenant compte des informations recueillies grâce à ces systèmes et de les stocker dans leur "journal de santé" (Brabham et al., 2014).

Dans le cadre de ce projet, nous proposons une méthodologie de conception et de gestion des applications mobiles réparties en exploitant les paradigmes de *crowdsourcing* et de *crowdsensing* qui permettent de suivre l'évolution de certaines maladies infectieuses.

## **1.2 Problématiques**

La surveillance des maladies contagieuses et infectieuses lors d'une épidémie est un domaine qui comporte plusieurs défis majeurs (Gonçalves et al., 2013).

### **1.2.1 La prédiction de l'impact réel de la maladie sur la population**

La complexité des interactions entre les malades et les individus en bonne santé rend généralement difficile la prédiction de l'impact réel d'une maladie infectieuse et contagieuse (Nistal et al., 2013).

Le déplacement des personnes est l'un des facteurs qui influencent la propagation des maladies contagieuses ou infectieuses qui se transmettent par contact direct ou indirect. La collecte d'informations sur la localisation de ces personnes ainsi que la distance entre personnes infectées et personnes non infectées et leur état de santé en fonction du temps sont primordiaux pour la surveillance des maladies infectieuses. Il est donc important d'établir une méthode qui permette de faire une estimation de l'impact de la maladie sur la population lors d'une épidémie, par exemple.

### **1.2.2 Réduire la complexité des analyses et des calculs mathématiques.**

Diverses analyses et modèles mathématiques ont permis d'expliquer les phénomènes qui se produisent dans le cas de la propagation des épidémies et ont également joué un rôle central dans les stratégies de santé publique dans de nombreux pays. Cependant, en l'absence de données fiables et réelles, certains modèles mathématiques ont tendance à intégrer de nombreux paramètres permettant de se rapprocher le plus possible de la réalité. Cependant, cela rend les calculs intensifs (Liao et al., 2012).

### **1.2.3 Comprendre le déplacement des humains et trouver une approche similaire à leur déplacement**

Afin de mieux comprendre comment les maladies se propagent, il est important de comprendre le mode de déplacement des humains (Belik et al., 2009; González et al., 2008) et de trouver certains modèles mathématiques qui permettent de simuler leur déplacement.

Certains modèles de mobilités utilisés dans les réseaux ad hoc (Mobile Ad-Hoc Networks ou MANET) présentent certaines caractéristiques similaires aux déplacements des humains, d'une part, par leurs modes de déplacement aléatoires, et d'autre part, par leur contexte spatio-temporel (Bai & Helmy, 2004; Tsui et al., 2013). D'ailleurs, plusieurs travaux ont utilisé l'approche spatio-temporelle pour comprendre le déplacement des humains (Gao, 2015). L'utilisation de ces modèles pour simuler la propagation des maladies infectieuses présente des avantages pour améliorer la surveillance des maladies infectieuses.

### **1.2.4 La surveillance de transition d'états de la maladie dans la population**

Les informations sur la maladie à différents stades de l'état de santé d'un individu au cours d'une épidémie sont d'une importance capitale pour le contrôle de cette épidémie. Cependant, la diversité des maladies et la variabilité dans le changement de phase de la maladie rendent difficile l'obtention d'informations exactes sur l'état de santé des personnes et l'évolution de la propagation de cette maladie (Mikler et al., 2009). De ce fait, il est important de trouver une façon de contrôler et de comparer la transition des différentes étapes possibles de la maladie au fil du temps, et en conséquence, cela implique de classer la population entière en différentes sous-populations sur la base de l'état de chaque individu (Brauer, 2008; Sontag, 2006).

Dans certains cas, si la population est grande, un modèle mathématique représentant les différentes étapes possibles de la maladie permet de faire une analyse quantitative de son évolution au fil du temps (Chapman et Evans, 2009; Nistal et al., 2013).

### 1.3 Portée et Limite du projet

La surveillance des maladies contagieuses et infectieuses est un domaine très vaste à couvrir, d'une part, par les différents modes de propagation de ces maladies et d'autre part, par la diversité des maladies (Mikler et al., 2009).

Dans le cadre de ce projet, nous nous intéressons à certaines maladies contagieuses qui se propagent principalement par contact direct ou indirect.

Généralement, les modèles épidémiologiques existants sont classés en deux catégories : les modèles descendants (*Top-Down models*) et les modèles ascendants (*bottom-up models*) (Liao et al., 2012):

- La première catégorie se base sur des modèles mathématiques déterministes utilisant les équations différentielles et les chaînes de Markov (Kleczkowski et Grenfell, 1999; O'Neill, 2002). Ces modèles sont complexes et nécessitent l'intégration de nombreux paramètres leur permettant de se rapprocher de la réalité.
- La deuxième catégorie contient des modèles stochastiques qui sont basés sur des simulations informatiques, des automates cellulaires, des agents à base de méthodes et des petits réseaux mondiaux (Perez et Dragicevic, 2009).

Dans notre projet, nous utiliserons l'approche ascendante, plus particulièrement la simulation informatique basée sur l'approche stochastique (Mikler et al., 2009) .

La répartition de la population en groupes selon leurs états de santé est une méthode mathématique très utilisée comme modèle permettant d'étudier la propagation des maladies depuis plusieurs années (Sontag, 2006). Plusieurs modèles mathématiques épidémiologiques découlent du modèle de base appelé SIR (*Susceptible, Exposed, Infected*) (Hethcote, 2000) (Chapman & Evans, 2009). À titre d'exemple, on peut citer des modèles tels que, SEIR (*Susceptible, Exposed, Infected,*

*Removed/Recovered*) (Nistal et al., 2013), SEPAIR (*Succeptible, Exposed, Infected, Asymptomatic, Symptomatic, Recovered*) (Lee et al., 2010) etc. Dans le cadre de notre projet, nous utiliserons le modèle SEIR.

L'étude de la propagation des maladies contagieuses peut se faire à un niveau local ou global. (Lee et al., 2010). Dans notre projet, nous nous intéressons à la propagation de maladies dans un cadre régional.

#### **1.4 Objectifs**

Un système de surveillance des maladies infectieuses est essentiel pour diminuer les risques de propagation. Par conséquent, nous visons les objectifs suivants :

- La mise en oeuvre d'un ensemble de services de type *crowdsourcing* et *crowdsensing* qui permet de lire les données captées, les partager et les rendre accessibles à distance, notamment pour les institutions de santé publique afin de surveiller les maladies contagieuses.
- La mise en place d'un système permettant de simuler une épidémie à partir des modèles de mobilités utilisés dans les réseaux ad hoc sans-fils en utilisant l'approche stochastique.
- La mise sur pied d'un prototype de visualisation et de prise de décision en se basant sur la comparaison des données réelles et simulées.

#### **1.5 Méthodologie**

La méthodologie appliquée est la suivante :

- L'utilisation des modèles de mobilité dans les réseaux ad hoc en vue de simuler la propagation des épidémies et les comparer avec les données réelles.
- L'utilisation des objets connectés, notamment des capteurs pour la collecte des données biologiques (entre autre la température corporelle) et les données de géolocalisation.

- La conception d'une architecture de type *crowdsourcing* et *crowdsensing* dans le but de lire les données captées, les partager et les mettre à la disposition d'une partie tierce en vue de les exploiter.
- La mise en œuvre d'un prototype de visualisation et de prise de décision inspirée des données réelles et simulées.
- L'utilisation de la méthode stochastique pour simuler une épidémie en vue de subdiviser la population en différents groupes suivant leur état de santé.
- La conception d'une base de données pour sauvegarder les données sur l'état de santé des individus en vue de suivre la variation dans le changement d'état de leur santé et l'évolution de l'épidémie dans le temps.

## 1.6 Organisation du mémoire

La structure du mémoire est organisée selon les chapitres suivants:

- Le chapitre 2 présente une revue critique de la littérature sur la propagation des maladies contagieuses ou infectieuses (épidémie), les modèles de mobilités utilisés dans les réseaux ad hoc ainsi que l'approche de *crowdsourcing* et de *crowdsensing* pour collecter et partager les données. Nous abordons également, les techniques utilisées pour faire le contrôle de la propagation des maladies contagieuses ou infectieuses.
- Le chapitre 3 présente les méthodes que nous avons appliquées afin d'utiliser les modèles de mobilité comme approche pour la simulation d'une épidémie. Nous présentons ensuite l'environnement de simulation, les outils utilisés pour réaliser ce travail ainsi que les résultats de la simulation. Nous présentons également un prototype de visualisation et de prise de décision tiré des données réelles et simulées.
- Le chapitre 4 présente le modèle proposé pour surveiller les maladies infectieuses. Nous présentons par la suite une architecture de *crowdsourcing* ainsi qu'une architecture d'interconnexion. Nous abordons aussi la méthode

proposée pour faire l'intégration des données réelles qui proviennent des capteurs, notamment les données de géolocalisation, de température corporelle et de rythme cardiaque. Nous abordons ensuite les techniques proposées pour collecter les données provenant des capteurs.

- Le chapitre 5 présente les travaux futurs à réaliser dans ce domaine. Nous terminons ce chapitre en présentant une conclusion sur le travail qu'on a réalisé.



## CHAPITRE II

### REVUE DE LITTÉRATURE

Dans ce chapitre, nous présentons une revue de littérature sur les différentes techniques liées à la propagation des maladies infectieuses. Nous montrons ensuite les différents modèles épidémiologiques utilisés dans ce domaine pour simuler ou contrôler la propagation des maladies contagieuses. De plus, nous abordons les modèles de mobilités dans les *MANET (Mobile Ad hoc Network)* et nous utilisons certains d'entre eux comme méthode pour simuler la propagation des maladies contagieuses.

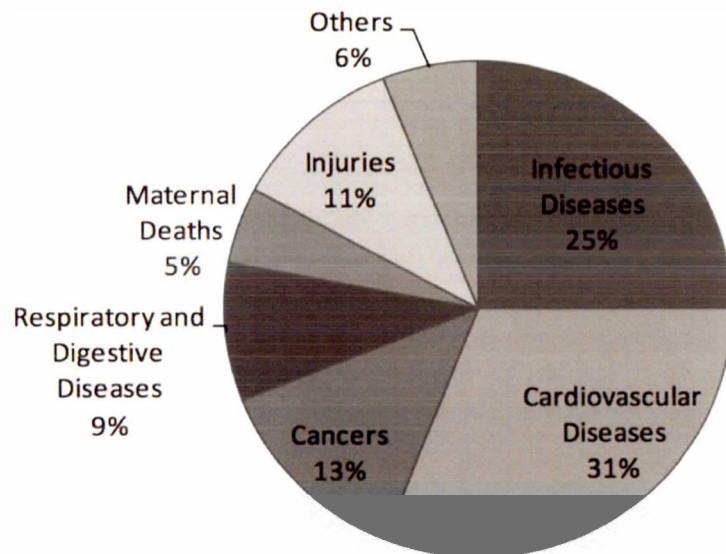
#### **2.7 Les maladies infectieuses**

Selon un rapport publié par l'OMS (l'Organisation Mondiale pour la Santé ou World Health Organization, WHO<sup>1</sup>), les maladies infectieuses représentent le deuxième facteur qui cause le plus grand nombre de décès dans le monde après les maladies cardiovasculaires (World Health Organization, 2008), (fig.2.1), (Stattner & Vidot, 2011). Ces maladies peuvent causer des dommages potentiels aux patients si elles ne sont pas détectées et traitées rapidement. Dans certains cas, lorsque la maladie est très contagieuse et se propage rapidement au niveau de la population, une intervention rapide et adéquate peut largement aider à limiter les dégâts. Donc, il s'avère nécessaire de trouver un moyen permettant de simuler la propagation des maladies infectieuses afin de comprendre comment elle se propage en vue de la contrôler. C'est dans ce contexte que plusieurs études ont été menées dans le but de trouver des

---

<sup>1</sup> WHO : Institution spécialisée des Nations Unies qui s'occupe de la santé publique internationale.

méthodes qui permettent de mieux comprendre l'évolution des maladies infectieuses et d'évaluer leur impact sur une population (Hethcote, 2000),(Belik et al., 2009),(Perez & Dragicevic, 2009). Parmi les techniques les plus utilisées pour étudier la propagation des maladies contagieuses, des formulations à l'aide des méthodes analytiques ont été considérées (Kermack & McKendrick, 1927) (Hethcote, 2000). Cependant, elles sont limitées dans certains cas et ne suffisent pas pour simuler les épidémies dans toutes leurs intégralités. En effet, certains aspects microscopiques ne sont pas pris en compte tels que les aspects spatio-temporels (Gao, 2015), (Filipe, Otten, Gibson, & Gilligan, 2004). Avec l'utilisation de l'informatique, des techniques plus systématiques ont été mises en œuvre afin d'étudier la propagation des épidémies. Nous abordons certaines techniques dans ce chapitre .



**Figure 2.1** Répartition des causes de décès selon (World Health Organization, 2008), (Stattner & Vidot, 2011)

### 2.7.5 Propagation des maladies

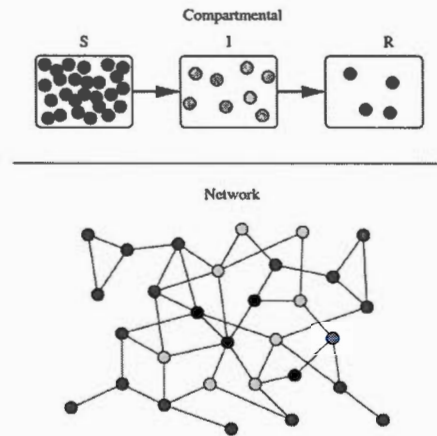
Pour étudier la propagation des maladies infectieuses, les épidémiologistes ont eu recours généralement à trois grandes approches de calcul (Bisset, Feng, Marathe, & Yardi, 2009), (Connell, Dawson, & Skvortsov, 2009):

- *Les modèles à compartiment* : qui permettent à partir de certaines équations mathématiques de décrire la transmission d'une épidémie entre les différents compartiments (Brauer, n.d.) (Hethcote, 2000; Sontag & Sontag, 2006).
- *Les modèles de percolation en réseau* : est une approche décrivant comment la transmission se propage dans un réseau à travers des liens qui relient les noeuds entre eux (Easley & Kleinberg, 2010).
- *Les modèles à base d'agents* : permettent de modéliser la propagation d'une épidémie à partir des agents qui sont des composants logiciels autonomes, hétérogènes et interagissant (Perez & Dragicevic, 2009).

Selon (Bisset et al., 2009), les deux premiers modèles sont très efficaces du point de vue du calcul. Cependant, ils se penchent sur des hypothèses irréalistes ou ils ignorent certaines informations essentielles à la planification du contrôle épidémique. Dans le cas des hypothèses irréalistes, nous avons, par exemple, les modèles à compartiments qui supposent que la population est mélangée et que tout individu *sensible* a une probabilité uniforme d'être infecté par une personne contaminée (fig. 2.2)<sup>2</sup>. Quant aux cas d'absence d'informations, nous retrouvons, par exemple, les modèles de percolation qui ne donnent pas la taille finale de l'épidémie, et informe peu sur les temps de variations de la maladie lors de la progression de l'épidémie.

---

<sup>2</sup> La figure 2.2 de l'article (Meyers, 2007), illustre la répartition d'une infection d'un modèle à compartiment vers un modèle réseau par contact. on peut voir que pour le modèle à compartiment tous les individus d'un même groupe ont une probabilité égale d'être infecté.



**Figure 2.2** Comparaison de la dynamique d'une infection d'un modèle à compartiment par rapport à un modèle réseau (Meyers, 2007). Dans cette figure on peut voir que pour le modèle à compartiment tous les individus d'un même groupe ont une probabilité égale d'être infecté.

Une classification beaucoup plus raffinée dans la littérature regroupe ces modèles épidémiologiques en deux grandes catégories (Liao et al., 2012):

- a) **les modèles stochastiques** ou **les modèles ascendants** : Ces modèles se basent sur des probabilités et des statistiques. Ils comprennent *les modèles à base d'agents informatiques* et *les modèles de la théorie du réseau* (Liao et al., 2012).
- b) **Les modèles déterministes** ou **les modèles descendants** : Ces modèles se basent sur des équations différentielles. Ils comprennent *les modèles à compartiment* (Brauer, 2012).

### **2.7.6 Comparaison entre les modèles stochastiques et les modèles déterministes.**

Les *modèles stochastiques* se basent notamment sur les probabilités de variation de risques d'exposition lors d'une épidémie. Ils fournissent beaucoup plus de possibilités pour une modélisation à l'échelle individuelle en tenant compte des populations de petite taille où chaque individu joue un rôle important dans le modèle. Par conséquent, ils sont utilisés lorsque les hétérogénéités connues sont importantes dans les populations de petite taille (Hethcote, 2007).

Les *modèles stochastiques* présentent plusieurs avantages selon (Hethcote, 2007), elles permettent plus précisément une observation étroite de chaque individu de la population sur un modèle probabiliste. Ils peuvent cependant être coûteux à mettre en place car ils nécessitent de nombreuses simulations pour obtenir des prévisions utiles. Ces modèles peuvent être mathématiquement très complexes et ne contribuent pas à une explication exhaustive de la réalité.

Les *modèles déterministes* permettent de classer les individus en différents sous-groupes (Hethcote, 2000). Ils se retrouvent parmi les modèles les plus utilisés pour décrire le comportement des maladies infectieuses (Kermack & McKendrick, 1927). Le modèle SEIR, par exemple, comprend quatre compartiments représentés par les sujets sensibles, exposés, infectieux et récupérés. Entre ces compartiments, il existe des taux de transitions (force d'infection, taux de contact) qui indiquent comment la taille d'un compartiment change par rapport à l'autre (Hethcote, 2007).

Selon les recherches réalisées par (Hethcote, 2007), les modèles déterministes tentent de décrire et d'expliquer ce qui se passe sur la moyenne à l'échelle de la population. Ils intègrent des grandes populations. Ils en déduisent que ces modèles sont relativement faciles à mettre en place, car ils nécessitent moins de données. L'autre

aspect important vient du fait que les logiciels informatiques exploitant ces modèles sont largement disponibles et conviviaux. Par ailleurs, de nombreux autres modèles plus complexes existent et peuvent incorporer des éléments stochastiques. Par exemple, certains modèles à bases d'agent sont agrégés à un modèle déterministe comme *SEIR* pour simuler la propagation d'une épidémie.

## 2.7.7 Les modèles stochastiques

### 2.7.7.1 Les modèles de percolation en réseau

Selon les travaux de (Kleczkowski & Grenfell, 1999), il y a eu à un intérêt accru pour modéliser les interactions entre les agents qui se trouvent dans un réseau avec un mélange d'interactions locales et globales. Lorsque les agents communiquent avec leurs voisins les plus proches seulement, la progression d'une maladie est lente. Ceci est causé par des niveaux de corrélations élevées au niveau des interactions locales; ce qui conduit à une perte de contacts fréquents. Cependant, les agents peuvent aussi faire des contacts couvrant de grandes distances. Avec les moyens de transport de nos jours, les maladies peuvent se propager rapidement à travers différents pays ou continents. Donc, ils voulaient à partir de leur modèle *mean field type equation* qui se comporte comme un *modèle réseau petit monde*, de démontrer que même en présence de corrélation locale élevée, leur modèle peuvent réussir. Ils ont fait référence aux travaux de (Watts & Strogatz, 1998) qui ont analysé certaines propriétés des réseaux maillés (*mesh network*) et ont montré qu'ils se comportent comme un modèle de *réseau petit monde*, même pour un paramètre de maillage relativement faible. Un modèle similaire a été analysé par (Boccaro & Cheong, 1999), où les individus ont été autorisés à changer de place sur un réseau à deux dimensions. Ainsi, ils ont prouvé lorsque les interactions entre les agents se font sur une longue portée, cela provoque une propagation très rapide de l'épidémie dans la population.

Dans le travail de (Kleczkowski & Grenfell, 1999), les auteurs ont étudié les effets de corrélations spatiales locales d'une propagation épidémique temporelle en fonction de la proportion croissante de contacts mondiaux (modèle «petit monde») d'un modèle automate cellulaire (CA). Ils en déduisent que même en présence de corrélations locales élevées, le modèle (*Mean field types equations*) peut être tout à fait adapté si le taux de contact est traité comme un paramètre libre. Ils en concluent que le taux de contact reflète non seulement une situation microscopique et épidémiologique, mais aussi une structure sociale complexe, y compris les contacts à court et à long terme suivant une structure hiérarchique de la société. Parmi les travaux qui se basent sur le principe de réseaux *petit monde*, nous pouvons citer le travail de (Kleczkowski & Grenfell, 1999) et de (Watts & Strogatz, 1998).

De nos jours, les réseaux jouent un grand rôle dans la simulation des épidémies. Ils sont considérés comme un système qui comprend un ensemble d'entités interconnectées qui permet de modéliser des liens entourant certains phénomènes physiques, biologiques et sociaux conduisant à des modèles prédictifs (Matt J Keeling & Eames, 2005).

Il existe en effet une certaine similarité entre la façon dont les objets connectés interagissent entre eux avec la manière dont les individus évoluent ou sont interconnectés (relations sociales) dans une communauté (Bisset et al., 2009). Dans le but d'exploiter ces avantages, plusieurs travaux se sont intéressés à l'idée d'utiliser l'approche réseau pour modéliser la propagation des maladies infectieuses (Stephen Eubank, 2005) ,(Stattner & Vidot, 2011), (Meyers, 2007), (Matt J Keeling & Eames, 2005). Ces réseaux vont de modèle simple en réseaux complexes par la façon dont les éléments qui les compose sont interconnectés (topologie physique) et aussi par la façon dont les informations transitent à travers les différents noeuds (topologie logique).

De ce fait, il est important de bien comprendre leur structure afin de bien les modéliser. C'est en ce sens que certains travaux de recherche sont axés sur la description formelle des réseaux, dont la théorie des graphes (Easley & Kleinberg, 2010), (Matt J Keeling & Eames, 2005). En effet, un réseau peut être représenté par un graphe  $(G=V,E)$ .  $V$  représente l'ensemble des noeuds et  $E$  représente l'ensemble des liens.

L'autre aspect important dans la modélisation des réseaux c'est de comprendre les relations qui existent entre les différentes entités qui les composent. Donc, dans l'optique de la propagation des épidémies, certains travaux s'intéressent plutôt au fondement des liens (aspects sociaux) qui relient les entités (individus) dans un réseau (Stattner & Vidot, 2011)(Stattner, 2013) (Kleczkowski & Grenfell, 1999).

#### **2.7.7.2 Modèles à base d'agent**

Étant une approche relativement nouvelle, les modèles à base d'agents sont plus puissants et plus réalistes dans la simulation de la propagation des maladies dans la prédiction des impacts sur la population et surtout dans l'efficacité des interventions. Ce modèle est classé comme une solution pour modéliser les interactions sociales complexes dans un monde réel (Stattner & Vidot, 2011).

Selon (Perez & Dragicevic, 2009), les modèles à base d'agents permettent de surmonter les limites des différentes approches, telles que les automates cellulaires (Margolus & Toffoli, 1987; Sun et al., 2012) et des modèles épidémiques classiques (Hethcote, 2000). Ils permettent également de modéliser les interactions entre les individus, ce qui permet d'étudier les aspects spatiaux spécifiques de la propagation des épidémies et de traiter la nature stochastique du processus épidémique (Perez & Dragicevic, 2009).

Parmi les nombreuses méthodes de simulation à base d'agents, *EpiSimdemics* permet de simuler la propagation des maladies contagieuses.



D'après le travail de (Bisset et al., 2009), *EpiSimdemics* est différent des autres systèmes sur au moins trois aspects. Premièrement, il est une plateforme assez riche qui permet la modélisation de nombreux processus de propagation de maladie, outre les modèles *Sensible-Infectieuses-Récupérées* (SIR) et (SEIR) *Sensibles-Exposés-Infectieuses-Récupérées* utilisés par les épidémiologistes (Kermack & McKendrick, 1927). Deuxièmement, *EpiSimdemics* est construit sur la plateforme de *Simdemics* (Barrett, Eubank, et Marathe 2008). Il utilise un modèle de population synthétique qui est statistiquement identique à la population du monde réel. Troisièmement, *EpiSimdemics* intègre certaines modélisations d'interventions dans le domaine de la santé publique, les comportements individuels et les réseaux d'interaction.

Selon (Bisset et al., 2009), tous ces simulateurs ont été largement évalués et utilisés dans des dizaines d'études de santé publique à grande échelle (Atkins et al. 2006, Atkins et al. 2007, Halloran et al., 2008), *EpiSimdemics* semble être le meilleur choix pour simuler la propagation des épidémies.

### **2.7.8 Modèle déterministe (Les modèles à compartiment : approche mathématique)**

L'approche la plus commune à la modélisation déterministe de la progression de l'infection au niveau d'une population est à travers une représentation sous forme de compartiments des différentes étapes de l'infection Elle est souvent décrite comme un système d'équations différentielles ordinaires (Brauer, 2012). Cette méthode a été développée par (Kermack & McKendrick, 1927) et avait pour but de comprendre l'évolution de l'épidémie au sein des grandes populations de taille constante. L'application de cette technique de modélisation mathématique a été motivée par la nécessité d'avoir des prédictions plus précises de la propagation des infections au niveau de la population. Cela contribue à une meilleure compréhension des mécanismes de propagation permettant d'aider au développement de stratégies optimales de contrôle par les interventions de santé publique. Nous retrouvons

plusieurs Modèles à Compartiment dans la littérature, parmi ces modèles nous citons le modèle SI, SIR, et SEIR (Hethcote, 2007).

### 2.7.8.3 Modèle SI

Le modèle SI est la plus simple parmi les modèles épidémiques. Dans ce modèle, la population est divisée en compartiments : susceptible  $S(t)$  et infectieuse  $I(t)$  (Hethcote, 2007). Ils considèrent que la maladie est très contagieuse, mais pas grave, ce qui signifie que les personnes infectées restent en contact avec les sujets sensibles pendant le temps  $t \geq 0$ . Ils supposent également que les personnes infectées continuent de propager la maladie jusqu'à la fin de l'épidémie. Dans ce cas, la taille de la population est constante c.-à-d. ( $S(t) + I(t) = N$ ) et le mélange au niveau de la population est homogène. Dans ce modèle, le taux d'infection est proportionnel au nombre de personnes infectées, d'où  $\beta = r\lambda I$

Avec :

$r =$  Nombre d'individus récupérés avec l'immunité acquise.

$I =$  Nombre d'individus infectés.

$\lambda =$  Force de l'infection.

Le modèle SI est décrit par une paire d'équations différentielles ordinaires (Hethcote, 2007):

$$\frac{dS(t)}{dt} = -r\lambda I(t)S(t) \quad (2.1)$$

$$\frac{dI(t)}{dt} = r\lambda I(t)S(t) \quad (2.2)$$

D'où :

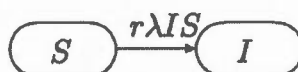
$$N = S(t) + I(t)$$

$$S(t) = N - I(t)$$

Donc, nous avons:

$$\frac{dI(t)}{dt} = r\lambda I(t)[N - I(t)]$$

Le modèle SI est représenté par la figure 2.3



**Figure 2.3** Représentation du modèle SI (Hethcote, 2007)

#### 2.7.8.4 Modèle SIR

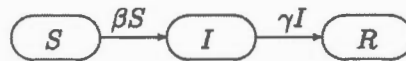
Le modèle SIR est généralement connu sous le nom de modèle à compartiment ou modèle dit de Kermack-McKendrick (Hethcote, 2007) (Sontag & Sontag, 2006), (Brauer, 2012).

Selon (Hethcote, 2000), il existe deux types de modèles SIR. Ils décrivent soit une épidémie, c.à.d une propagation rapide d'une maladie infectieuse, soit une endémie, c.-à-d. une maladie qui demeure dans la population pendant une longue période de temps où la classe de sujets sensibles est alimentée par les nouveau-nés ou les individus récupérés qui ont perdu leur immunité temporelle. Ces deux modèles sont les bases de l'épidémiologie mathématique moderne et sont encore largement utilisés dans la pratique.

Le *framework* du modèle SIR peut être utilisé pour caractériser les systèmes épidémiques où les étapes de l'infection peuvent être représentées en trois étapes distinctes. La population totale est donc divisée en trois classes sans distinctes (sensibles, infectieux et récupérés) représentant les sous-populations de personnes ayant un état spécifique de la maladie. La classe sensible comprend toutes les

personnes qui sont en mesure de contracter la maladie et deviennent infectées, la classe “infectée” et la classe “récupérée” qui contient tous les individus qui sont guéris, c’est-à-dire qui ont acquis une forme d’immunité.

Le modèle SIR peut être représenté par la figure 2.4:



**Figure 2.4** Représentation du modèle SIR (Hethcote, 2007)

Le modèle de SIR de base peut être représenté par un système dynamique non linéaire de trois équations différentielles (Connell et al., 2009).

$$\frac{dS}{dt} = -\alpha SI \quad (2.1)$$

$$\frac{dI}{dt} = \alpha SI - \beta I \quad (2.2)$$

$$\frac{dR}{dt} = \beta I \quad (2.3)$$

Avec :

$S$  = Nombre d'individus sensibles.

$I$  = Nombre d'individus infectés.

- $R$  = Nombre d'individus récupérés.
- $\alpha$  = Probabilité pour qu'un membre spécifique de  $I$  infecte un membre spécifique de  $S$ , multiplié par la chance, pour qu'ils se rencontrent par unité de temps.
- $\beta$  = la probabilité qu'une personne infectée se récupère par unité de temps.

Les équations 2.1, 2.2 et 2.3 décrivent l'évolution temporelle d'une population entre états pathologiques sensibles, infectés et récupérés.

Selon (Connell et al., 2009), le système *SIR* a été initialement développé sur la base d'un modèle très simplifié d'interaction sociale. Donc, l'hypothèse d'un mélange uniforme de la population a été implicitement utilisée pour déduire les termes non linéaires du système *SIR*. Cela met en évidence les faiblesses du modèle *SIR* et représente une base pour les critiques. Par conséquent, une généralisation importante a été faite qui permet d'étendre le modèle de *SIR* standard à un ensemble non uniforme des individus. Ce qui est exprimé dans, les équations suivantes (Connell et al., 2009):

$$\frac{dS}{dt} = -\alpha S^\delta I \quad (2.1)$$

$$\frac{dI}{dt} = \alpha S^\delta I - \beta I \quad (2.2)$$

$$\frac{dR}{dt} = \beta I \quad (2.3)$$

Avec

$\delta = 2$  : est un paramètre décrivant une efficacité de mélange social.

$\delta = 1$  : correspond à un mélange uniforme, comme dans le modèle de *SIR* standard.

Le paramètre d'ajustement supplémentaire ( $\delta$ ) du modèle étendu *SIR* permet de mieux concilier avec les observations réelles et de mettre en évidence deux principales valeurs observables dans une simulation d'une épidémie: la durée de l'épidémie et de son amplitude (Barthélemy, Barrat, Pastor-Satorras, & Vespignani, 2005).

### 2.7.8.5 Modèle SEIR

Le modèle *SEIR* contient un élément supplémentaire, dit exposé (E) (Brauer, 2012). Ce sont les gens qui sont infectés, mais les symptômes de la maladie ne sont pas encore visibles. Ils ne peuvent pas transmettre la maladie car elles sont dans une période de latence. Pour certaines maladies, il faut un certain temps pour qu'un agent infectieux se multiplie à l'intérieur de l'hôte jusqu'au niveau critique de telle sorte que la maladie se manifeste dans le corps de l'hôte. Ceci est appelé une période d'incubation. Dans ce modèle, la taille de la population est constante et les taux de variation d'un compartiment à l'autre suivent le système ci-dessous (Hethcote, 2000):

$$\frac{dS}{dt} = \mu N - \beta SI - \mu S \quad (2.1)$$

$$\frac{dE}{dt} = \beta SI - \mu E - \theta E \quad (2.2)$$

$$\frac{dI}{dt} = -\mu I + \theta E - \gamma I \quad (2.3)$$

$$\frac{dR}{dt} = -\mu R + \gamma I \quad (2.4)$$

Pour ce modèle, la probabilité de survivre à la période de latence et d'entrer dans la période de contagiosité est égale à  $\frac{\theta}{\theta + \mu}$ . Par conséquent, le nombre de reproductions de base dans ce cas sera  $R_0 = \frac{\theta}{\theta + \mu} \frac{\beta N}{\gamma + \mu}$

### 2.7.9 Synthèse

Selon (Connell et al., 2009), l'approche mathématique fournit des résultats rigoureux et il est le plus simple à mettre en œuvre, Cependant, elle comporte une lacune dans

le fait que les scénarios simplifiés ne peuvent être traités que de manière analytique. En ce qui concerne l'approche de simulation à base d'agents (Perez & Dragicevic, 2009), elle est plus flexible en termes d'évaluation de scénarios réalistes et est devenue de plus en plus populaire.

Avec la disponibilité croissante des ressources informatiques, les méthodes à base d'agents permettent une modélisation de fidèle des épidémies tant au niveau mondial, national que communautaire. Leurs principal inconvénient est la validation d'un modèle en particulier (Connell et al., 2009). Par conséquent, l'une des étapes importantes de la validation des modèles à base d'agents est d'appliquer la méthode dite "alignement de modèles" (Carley et al, 2006).

#### **2.7.10 Aspects importants dans la simulation d'une maladie infectieuse**

Certains facteurs sont importants lors de la simulation de la propagation d'une épidémie. Pour qu'un agent infectieux propage la maladie à travers la population, il est nécessaire d'avoir, par exemple, un contact avec les sujets exposés. La possibilité pour que cette maladie se transmette aux personnes saines peuvent donc dépendre de plusieurs facteurs, comme la virulence du micro-organisme (agent infectieux), l'étendue à laquelle ils sont déchargés, le degré de proximité (emplacement géographique), les groupements scolaires, la taille des familles, les habitudes sociales, etc. (Grassly & Fraser, 2008). Dans certains cas, il pourrait également être nécessaire d'avoir des informations sur la taille de la population ou le taux de reproduction de base (Hethcote, 2007),(M. J. Keeling & Danon, 2009). Il y a beaucoup d'autres aspects qui peuvent être importants dans la propagation d'une maladie, par exemple, sa vitesse, son impact et la distribution non uniforme de la population dans l'espace (Bisset et al., 2009), (Filipe et al., 2004). En conséquence, le taux de transmission d'une maladie infectieuse peut être plus élevé dans certains endroits que d'autres. Un autre facteur essentiel dans le contrôle de la propagation de l'épidémie sont les personnes qui ne présentent pas de symptômes de la maladie,

c'est-à-dire, les personnes dans le stade asymptomatique. Elles se soustraient à la vigilance sanitaire, et peuvent compliquer le contrôle de la maladie et augmenter le risque d'infection dans la population (Beaglehole, Bonita, & Kjellström, 2000).

Dans notre travail, nous prendrons en considération certains aspects essentiels à la simulation des maladies infectieuses. Nous utiliserons les modèles de mobilités appropriés dans les réseaux MANET (Mobile Ad-hoc Networks) en incorporant certains éléments stochastiques. Nous allons nous inspirer des techniques de compartiment, plus précisément le modèle SEIR permettant de subdiviser la population.

## **2.8 Les Modèles de mobilités dans les réseaux MANET**

Un réseau MANET (Mobile Ad hoc NETWORK) est une collection de nœuds mobiles sans fil formant un réseau auto configurable sans infrastructure (Bhupendra Suman, 2011). Ces nœuds sont capables de communiquer entre eux, mais leur mobilité dans le réseau rend le routage complexe, c'est-à-dire l'acheminement d'informations entre les nœuds dépend de cette mobilité. Un tel problème a suscité l'intérêt des recherches, d'une part, dans le but de comprendre le mode de déplacement des nœuds mobiles et d'autre part, de trouver une façon d'améliorer la performance des protocoles de routage (Zhang, Low, & Ng, 2011).

Divers chercheurs ont proposé différents types de modèles de mobilité permettant d'étudier, d'analyser et d'évaluer l'impact de la mobilité sur les protocoles de routage (Bai & Helmy, 2004)

Les modèles de mobilité sont conçus pour décrire le schéma de déplacement des utilisateurs mobiles. Ils permettent de mieux comprendre comment la localisation, l'accélération et la vitesse changent au fil du temps (Camp, Boleng, & Davies, 2002). Il existe un besoin réel pour les modèles de mobilité de capturer les différentes caractéristiques de la mobilité et de permettre de la représenter de façon réaliste de



manière à ce que les observations faites et les conclusions tirées lors des simulations ne soient pas faussées.

Lors de l'évaluation de la performance des protocoles dans les Manet, il est important de bien choisir le modèle de mobilité approprié. Vu que ces modèles de mobilité ne se comportent pas de la même façon, le modèle choisi permettra de déterminer si le protocole proposé sera utile une fois mis en œuvre. L'utilisation d'un modèle inapproprié pourrait fausser les résultats de la simulation. Il en est de même lors des simulations dans le cas des épidémies. Nous visons à utiliser des modèles de mobilité qui se rapprochent le plus de la réalité que nous voulons simuler.

Dans les prochaines sections, nous présenterons les différents modèles de mobilité (Bai & Helmy, 2004) (Bhupendra Suman, 2011), leurs limites, ainsi que les groupes de modèles les mieux appropriés pour simuler une épidémie. Nous devons également prendre en considération certaines contraintes dans l'application de ces modèles. Par exemple, un nœud mobile ne doit pas se déplacer sur des lignes droites à vitesse constante au cours de la simulation tout entière, car les individus ne se déplaceraient pas d'une façon si restreinte.

### **2.8.11 Les modèles de mobilité**

Il existe deux types de modèles de mobilité utilisés dans la simulation des réseaux: *les modèles de traces et les modèles synthétiques* (Bai & Helmy, 2004).

*Les modèles de traces* permettent d'avoir des informations précises sur les traces de mobilité des utilisateurs. Ils sont très utiles quand ils impliquent un nombre important d'utilisateurs pendant de très longues périodes d'observations. Cette méthode permet de créer des modèles de mobilité réalistes (Minkyong, Kotz, & Songkuk, 2006). Cependant, l'obtention des traces de mobilité réelle forme un grand défi vu que les *MANETS* n'ont pas été déployés sur une grande échelle. Par conséquent, divers chercheurs ont proposé d'autres modèles de mobilité qui essaient de représenter les

comportements des nœuds mobiles sans l'utilisation des traces. Ces modèles de mobilité, dit *synthétiques* (Roy, 2010) sont divisées en deux groupes : *Les modèles de mobilités d'entités* et *les modèles de mobilités de groupes* (Camp et al., 2002). Les modèles de mobilités synthétiques modélisent le mouvement en utilisant des processus aléatoires.

Dans certains scénarios, les deux modèles sont combinés; ce qui donne un modèle dit *hybride*. Donc, dans ce cas particulier, les données réelles (tels que le plan d'une ville) sont jumelées à un modèle statistique, par exemple, le mouvement des nœuds.

### **2.8.12 Les modèles de mobilité d'entité**

Dans les Modèles de mobilités synthétiques d'entités, les déplacements des nœuds sont indépendants, alors que dans les modèles de groupes les déplacements des nœuds dépend des autres nœuds du même groupe (Roy, 2010). Parmi les modèles de mobilités synthétiques d'entités (Camp et al., 2002), nous citons :

#### **1)- *Les Modèles de mobilité de promenade aléatoire (Random walk mobility model RWMM):***

*Le modèle de promenade aléatoire* a été proposé dans le but d'imiter le mouvement imprévisible des particules en physique. Il utilise un mouvement Brownien parce que certains nœuds mobiles sont soupçonnés de se déplacer d'une manière inattendue (Groenevelt, Altman, & Nain, 2006). Les nœuds choisissent les directions et les vitesses de façon aléatoire et ne font pas de pause lors des changements de vitesse et de direction (Roy, 2010).

#### **2)- *Les Modèles de mobilité de promenade aléatoire avec pause (Random Waypoint Mobility model RW<sub>p</sub>MM):***

*Le modèle de Random waypoint* (Bhatia, 2013), présente des similitudes avec

le modèle de promenade aléatoire. La différence réside dans le fait que les nœuds font des pauses entre les changements de direction et de vitesse. Les nœuds modifient leur vitesse et la direction à chaque intervalle de temps. L'un des problèmes majeurs de ce modèle est la distribution non uniforme des nœuds et le problème d'onde de densité (Zheng, Hong, & Ray, 2004).

3)- *Modèles de mobilité de direction aléatoire (Random Direction Mobility Model RDMM):*

*Le modèle de direction aléatoire* (Royer, 2001), permet de résoudre les problèmes de distribution spatiale non uniforme et d'ondes de densité du *modèle Random Waypoint* (Bhupendra Suman, 2011). Ce modèle force les nœuds mobiles à se diriger vers les limites de l'aire de simulation et une fois arrivé sur la frontière, les nœuds font une pause et choisissent une autre direction dans un angle entre 0 et 360 degrés avant de changer de vitesse et de position. Cela permet d'éviter une concentration des nœuds au centre de l'aires de simulation (C. Bettstetter, Resta, & Santi, 2003).

4) *Le modèle de mobilité dans une région de simulation illimitée (Boundless Simulation Area Mobility Model BSAMM) :*

Ce modèle convertit un secteur 2D rectangulaire de simulation en un secteur de simulation en forme de tore (Camp et al., 2002). Ce modèle présente une caractéristique de mobilité importante car il prend en considération l'aspect de dépendance temporelle. La direction et la vitesse du nœud courant dépendent des directions et des vitesses précédentes de ce nœud. Ce modèle donne l'impression que le secteur de simulation est illimité du fait que lorsque les nœuds atteignent la frontière, ils continuent à se déplacer et apparaissent de l'autre côté de la surface de simulation (Haas & Pearlman, 1998; Johnson & Maltz, 1996). Ce modèle est utile pour simuler les réseaux de grande

envergure comme le WMAN (*Wireless Metropolitan Area Network*) ou WWAN (*Wireless Wide Area Network*).

**5) Le modèle de mobilité de Gauss-Markov:**

Ce modèle a été proposé en vue de s'adapter aux différents niveaux d'aspects aléatoires (Camp et al., 2002). Dans ce modèle, un protocole de coordination est utilisé afin de changer l'aspect aléatoire de mobilité. Cela permet d'éviter les arrêts soudains des nœuds au cours de leurs déplacements. Cela a permis à ce modèle de se rapprocher des conditions réalistes.

**6) La version probabiliste du modèle de promenade aléatoire (PRWMM):**

Dans ce modèle une matrice de probabilité est utilisée dans le but de déterminer la prochaine position d'un nœud mobile (Camp et al., 2002). Il utilise trois états pour coordonner le déplacement du nœud. L'état 1 représente la position initiale, l'état 2 représente la position actuelle et l'état 3 représente la prochaine position d'un nœud dans l'axe des X ou l'axe des Y..

**7) Le modèle de mobilité des sections de ville (City Section Mobility Model CSMM):**

Ce modèle utilise les sections d'une ville comme aire de simulation (Camp et al., 2002). Il respecte l'aspect de restriction géographique puisqu'il limite fortement le comportement de déplacement des nœuds mobiles. Les rues et les limites de vitesse varient d'une ville à une autre. Ce modèle utilise un algorithme qui permet à chaque nœud mobile de commencer la simulation à un point défini sur une certaine rue et de choisir aléatoirement une destination sur une rue quelconque qui est également représentée par un point (Camp et al., 2002). Lors du déplacement des nœuds d'un point à l'autre, l'algorithme

localise un chemin correspondant au temps de voyage le plus court. Nous remarquons que ce modèle fournit des mouvements réalistes, puisque dans la réalité, les nœuds n'ont pas toujours la possibilité de se déplacer librement si nous tenons compte des obstacles.

### **2.8.13 Les modèles de mobilité de groupe**

Les modèles de *mobilité de groupe* (Roy, 2010)(Bhupendra Suman, 2011) permettent de simuler des situations dans lesquelles les décisions des nœuds dépendent des autres nœuds d'un même groupe. Parmi les modèles de mobilité de groupes, nous citons :

#### **1)– *Le modèle exponentiel aléatoire corrélé (Exponential Correlated Random Mobility Model ECRMM):***

Ce modèle est l'un des premiers modèles de mobilité de groupe proposé (Camp et al., 2002) Il permet de générer les mouvements de façon corrélée à l'aide d'une fonction mathématique.

#### **2)– *Le modèle de mobilité de colonne (column mobility Model CMM)***

Dans ce modèle, les nœuds forment une colonne et se déplacent uniformément suivant un axe donné (Bhupendra Suman, 2011) (Kim, 2005). Ce modèle est très utile pour simuler les recherches sur le terrain. Un exemple couramment utilisé dans la littérature pour illustrer ce modèle est une ligne de soldat marchant vers leur ennemi ou des opérations de recherches pour désactiver une mine antipersonnel.

#### **3)– *Le modèle de mobilité de communauté nomade (NCMM):***

Dans ce modèle , les nœuds se déplacent ensemble d'une place à une autre. Ils

se regroupent autour d'un point de référence donné à une certaine distance définie par les paramètres du modèle d'entité utilisé. Lorsque le point de référence change, les nœuds mobiles se déplacent vers le nouveau secteur défini par ce point. Donc tous les nœuds partagent un point de référence commun (Roy, 2010).

#### 4)– *Le modèle de mobilité de poursuite (Purse Mobility Model PMM):*

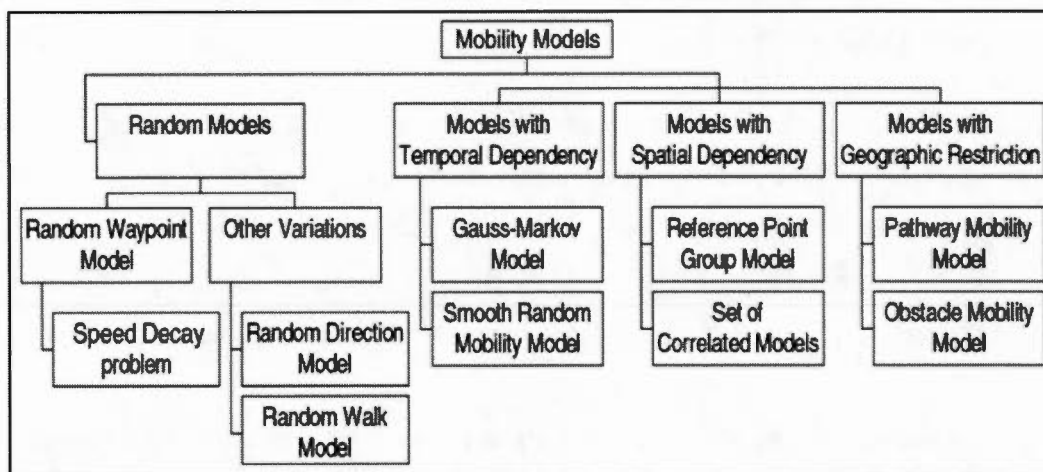
Dans ce modèle, les nœuds se déplacent ensemble en ayant pour cible un nœud particulier. Une équation de mise à jour est utilisée afin de permettre aux nœuds de suivre la cible (Bai & Helmy, 2003) .

#### 2.8.14 Classification des modèles de mobilité

Cette classification revient à grouper les nœuds suivant certaines caractéristiques particulières telles que les métriques de vitesse ou le choix des chemins (Bhupendra Suman, 2011). Certaines applications utilisant des modèles de mobilité nécessitent l'utilisation des schémas de mobilité complexes, mais très spécifiques. Ainsi, certains travaux de recherches ont commencé à se concentrer sur les modèles de mobilité avec des caractéristiques de mobilité différentes (Bai & Helmy, 2004). Dans ces modèles, le mouvement d'un nœud est plus ou moins limité par son historique (Trajectoire), ou d'autres nœuds voisins. C'est en fonction de ces caractéristiques spécifiques de mobilité que les modèles de mobilité sont classés en plusieurs catégories (Bhupendra Suman, 2011). Pour certains modèles, le mouvement d'un nœud est susceptible d'être affecté par sa trajectoire. Ce type de modèle fait partie de *modèle de mobilité avec dépendance temporelle (Models with Temporal dependency)* (Bai, Sadagopan, & Helmy, 2003). Dans certains cas de figure de mobilité, les nœuds ont tendance à voyager de manière corrélée. Nous nous référons à ce type de modèle de mobilité comme *modèle de mobilité avec dépendance spatiale (Models With spatial Dependency)* (Roy, 2010). Une autre classe est le *modèle de mobilité avec*

*restriction géographique (Models with Geographic Restriction)* (N. Aschenbruck, M. Frank, P. Martini, 2004) ( a. P. Jardosh, Belding-Royer, Almeroth, & Suri, 2005) où le mouvement des nœuds est délimité par les rues, des autoroutes ou des obstacles. Les modèles de mobilité avec obstacle ont été implémentés pour modéliser le mouvement des nœuds mobiles dans des terrains qui ressemblent à des topographies réelles.

La figure 2.5 donne une vue globale de la classification des modèles de mobilité



**Figure 2.5** catégories des modèles de mobilité dans les réseaux ad hoc (Bai & Helmy, 2004)

### 2.8.15 Limitations des modèles de mobilité aléatoire

Selon le scénario à simuler, le modèle de mobilité choisi doit permettre de se rapprocher le plus que possible de la réalité que nous voulons simuler. Cependant, certains scénarios réalistes exigent qu'un modèle doive permettre de saisir adéquatement certaines caractéristiques de mobilité, y compris *la dépendance temporelle, spatiale et la dépendance restriction géographique* (S Eubank et al., 2004), (A. Jardosh, Belding-Royer, Almeroth, & Suri, 2003). Cela permettrait d'avoir

des résultats fiables. Dans le cas de simulation d'une épidémie, il est important d'utiliser des modèles capables d'imiter à un certain niveau, le comportement des humains c.-à-d., leur déplacement et l'environnement dans lequel ils se trouvent comme les obstacles par exemple (S Eubank et al., 2004).

Dans cette partie, nous allons parler de certaines limites des modèles de mobilité et déduire par la suite les catégories de modèles de mobilités qui répondent mieux à notre situation (Roy, 2010), à savoir : *modèles de mobilité aléatoire, modèles de mobilité avec dépendance temporelle, modèles de mobilité avec dépendance spatiale et modèles de mobilité avec restriction géographique.*

Dans les études de simulation, *les modèles de mobilité aléatoire* sont les plus utilisés vue leur simplicité (Bai & Helmy, 2004). Dans ces modèles, les nœuds se déplacent aléatoirement et librement, sans aucune restriction. Par conséquent, la vitesse, la destination et la direction sont toutes choisies au hasard et indépendamment des autres nœuds. Si nous faisons référence aux déplacements des humains, il est évident que dans certains cas, ils se déplacent de manière aléatoire et indépendante. Cependant, ils sont quand même limités par certains obstacles dans leur parcours comme les bâtiments.

### ***RandomWalk model***

Le *modèle de mobilité de promenade aléatoire* est un processus de mobilité sans mémoire où les informations sur l'état précédent d'un nœud ne sont pas utilisées pour définir son état futur. En d'autres termes, la vitesse du nœud courant est indépendante de sa vitesse précédente et sa vitesse future est également indépendante de sa vitesse actuelle. La notion de dépendance temporelle n'est pas prise en considération par ce modèle (Bhupendra Suman, 2011). Cependant, nous observons que ce n'est pas le cas de nœuds mobiles dans de nombreuses applications réelles et il l'est encore moins dans le cas du déplacement des humains.



### ***Random waypoint model***

Dans la littérature, *le modèle de mobilité Random waypoint* est limité par deux problèmes majeurs : *la distribution spatiale non uniforme (non-uniform spatial distribution) et le phénomène d'onde de densité (density wave)* (Blough, Resta, & Santi, 2002) (Royer, 2001) (Bai & Helmy, 2004). les travaux de (Christian Bettstetter, 2001; Blough et al., 2002) ont montré que la distribution spatiale des nœuds du modèle aléatoire *Waypoint* est transformée d'une distribution uniforme à une distribution non uniforme après que la simulation ait commencé. Ce déséquilibre devient encore pire à la fin de la simulation car la densité de nœuds converge au centre de la région, tandis qu'elle est presque nulle autour de la limite de la zone de simulation.

### ***Random direction model***

Ce modèle a été proposé pour surmonter le problème de distribution spatiale non uniforme et d'ondes de densité du modèle aléatoire *waypoint* (Bai & Helmy, 2004). Au lieu de choisir une direction aléatoire dans l'espace de simulation, les nœuds choisissent plutôt de façon aléatoire et uniformément une direction après un temps de pause  $t$ , et procèdent de la même manière pour choisir une autre direction. Cependant, *le modèle Random direction* ne corrige pas la non-dépendance spatiale temporelle et géographique que requièrent certains scénarios de simulation.

### **2.8.16 Synthèse et Conclusions**

Nous pouvons conclure que *le modèle aléatoire Waypoint* et ses variantes ne respectent pas les trois principes clés que nous rencontrons dans les scénarios réalistes qui utilisent les modèles de mobilité (Roy, 2010) ( a. P. Jardosh et al., 2005); à savoir:

1. *Dépendance temporelle*: Dans le modèle aléatoire Waypoint et d'autres modèles aléatoires, la vitesse du nœud mobile est un processus aléatoire sans mémoire, c'est à dire que, la vitesse à la période actuelle est indépendante de l'époque précédente.

2. *Dépendance spatiale*: Dans le modèle aléatoire Waypoint et d'autres modèles aléatoires, le nœud mobile est considéré comme une entité qui se déplace indépendamment des autres nœuds. Ce genre de modèle de mobilité est classé comme modèle de mobilité individuelle.

3. *Restriction géographique* : Dans le modèle aléatoire Waypoint et d'autres modèles aléatoires, les nœuds mobiles peuvent se déplacer librement dans le champ de la simulation sans aucune restriction. Cependant, dans de nombreux cas réalistes, en particulier pour les applications utilisées dans les zones urbaines, le mouvement d'un nœud mobile peut être délimité par des obstacles, des bâtiments, des rues ou des autoroutes (Roy, 2010).

### ***Conclusion***

En résumé, les points énumérés ci-dessus nous permettent de souligner certaines limitations des modèles aléatoires qui pourraient avoir un impact sur la qualité des résultats dans notre simulation. De ce fait, les modèles de mobilité qui prennent en considération les critères spatiaux, temporels et géographiques font partie des modèles les mieux classés pour simuler, dans notre cas, la propagation des maladies.

### **2.9 Crowdsourcing et crowdsensing**

Le terme Crowdsourcing est considéré comme un nouveau paradigme qui compte sur l'intelligence collective pour résoudre un problème spécifique ou pour accomplir une tâche. Il a été introduit pendant les dernières décennies comme une forme de modèle de production pour les affaires selon laquelle certaines tâches sont distribuées à une communauté en ligne dans le but d'avoir des solutions plus efficaces qui coutent

moins cher pour chaque individu car chacun contribue à une petite partie de la tâche (Yuen, King, & Leung, 2011).

Le crowdsourcing consiste à prendre une tâche qu'exécute traditionnellement un individu et la distribuer à une large groupe de personnes sous la forme d'un appel ouvert (*open call*) (Howe, 2006).

Ils existent différentes définitions du crowdsourcing. Selon (Estellés-Arolas & González-Ladrón-de-Guevara, 2012), "*Le crowdsourcing se présente comme étant un type d'activité participative en ligne dans laquelle un individu, une institution, une organisation sans but lucratif, ou une compagnie propose à un groupe d'individus de connaissances hétérogènes par l'intermédiaire d'un appel ouvert et flexible, l'engagement volontaire d'une tâche. Les participants qui exécutent une tâche sont récompensés soit de manière économique ou par satisfaction de reconnaissance sociale ou d'estime de soi*".

Avec l'avènement des équipements mobiles intelligents (tablettes, montres, téléphones portables), le phénomène crowdsourcing a connu un nouvel essor. Les participants complètent normalement leurs tâches à travers des ordinateurs, désormais ils peuvent le faire à travers des équipements mobiles.. Ces appareils mobiles, étant équipés de plusieurs capteurs intégrés, permettent de collecter plus de données pertinentes à un contexte donné. Cette façon de recueillir les données a donné naissance au terme de *Crowdsensing*, plus précisément, le crowdsensing mobile (Ganti, Ye, & Lei, 2011). Les applications mobiles de crowdsourcing tirent en principe l'avantage de la présence des capteurs intégrés pour collecter des informations.

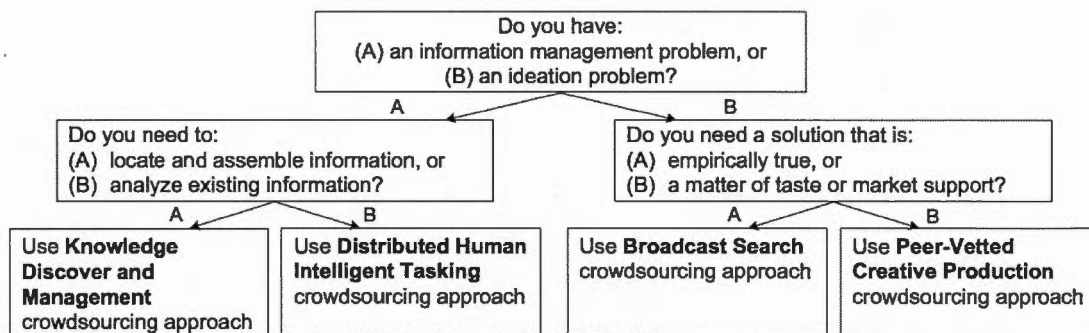
#### **2.9.17 Typologie de crowdsourcing (approche)**

Le *Crowdsourcing* a été utilisé pour une variété de problèmes dans différents domaines. Suivant le type de problème et le domaine en question, il est important de

déterminer quelles approches de crowdsourcing utiliser. Dans (Brabham , 2014) on décrit quatre typologies de crowdsourcing (voir fig 2.6) :

1. *The knowledge discovery and Management approach (KDM)* : qui est idéale pour des problèmes de gestion d'informations nécessitant la collecte, l'organisation, ou des rapports d'informations tels que la création de ressources collectives. Cette approche est très utile, lorsque les organisations rencontrent des problèmes dont l'information qu'ils cherchent existe en dehors de l'organisation dans des endroits dispersés. Ils ont besoin que les informations découvertes soient rassemblées dans des formats spécifiques à un seul endroit.
2. *The distributed Human Intelligence tasking approach* : qui est idéale pour les problèmes de gestion d'informations qui exigent l'analyse des données à grande échelle où l'intelligence humaine est plus efficace.
3. *The broadcast search approach* : Cette approche est idéale pour des problèmes qui exigent des solutions empiriquement démontrables, tels que les problèmes scientifiques. Le problème est diffusé à une communauté en ligne dans l'espoir de trouver quelqu'un qui connaît la solution.
4. *The Peer-Vetted Creative Production Approach* : Cette approche est idéale pour les problèmes qui exigent la contribution de nouvelles idées créatives comme solutions.

Afin de déterminer quel type d'approche à utiliser dans le cadre de notre projet , on a appuyé sur le modèle présenté par Brabham, (2014) sous forme d'arbre décisionnel qui permet de déterminer quel type de crowdsourcing utiliser suivant le problème considéré (voir fig. 2.6).



**Figure 2.6** Différentes approches de crowdsourcing.

L'approche appropriée dans notre cas est l'approche KDM car dans notre cas, nous nous intéressons à la collecte d'information.

### 2.9.18 catégories de crowdsourcing mobiles

Plusieurs types d'applications de crowdsourcing mobile existent dans la littérature. (Korthaus & Dai, 2012) en présente certains. Ces applications peuvent être classées en deux catégories (Christin et al 2011):

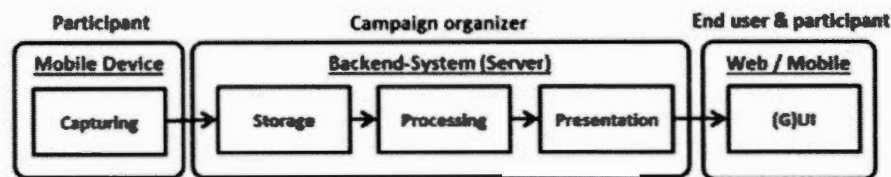
- *people centric (centré sur les personnes)*: qui collecte des données à propos des utilisateurs. Ces informations peuvent être des données sur l'état de santé des utilisateurs, leurs activités et leurs comportements.
- *Environnement centric (centré sur l'environnement)*: capture des informations sur l'environnement de l'utilisateur telles que la qualité de l'air, la pression de l'air, la pollution sonore, ou d'autres événements comme les accidents, les dommages ainsi que les désastres.

Dans notre projet nous adoptons l'approche *people centric* qui permet de collecter des informations sur l'état de santé des individus.

### 2.9.19 architecture de crowdsourcing

Les applications mobiles de crowdsourcing sont généralement définies en termes d'architecture par deux éléments principaux (Fuchs-kittowski & Faust, 2014): la capture des données et traitement des données.

L'interaction entre ces composants est organisée comme un système client-serveur où l'équipement mobile (client) est utilisé pour capter les données et le serveur (backend system) pour stocker, traiter et visualiser les informations captées (fig 2.7)



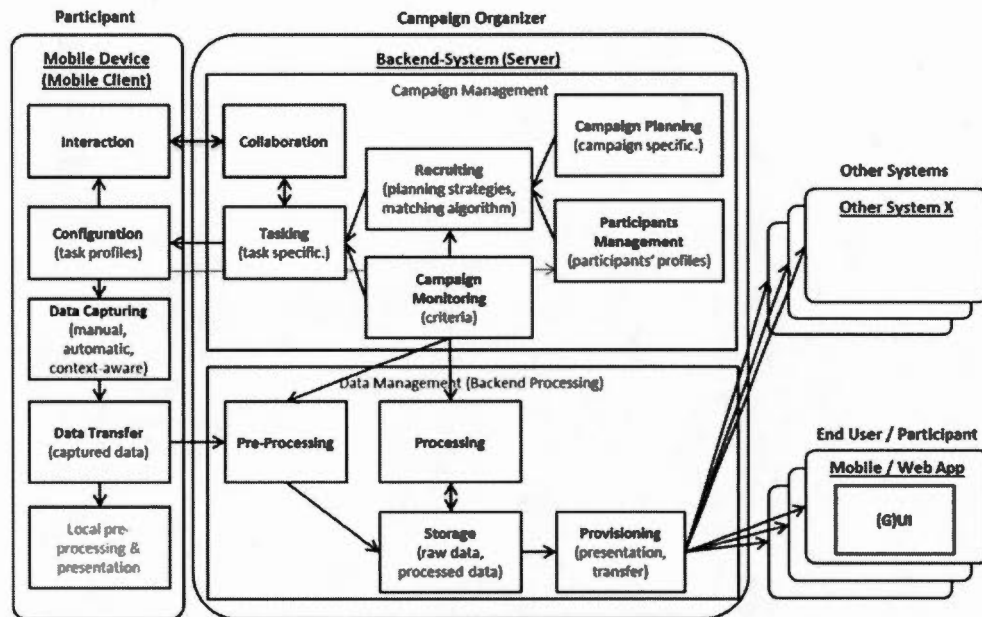
**Figure 2.7** Architecture générale de crowdsourcing (Fuchs-kittowski & Faust, 2014)

D'autres travaux redéfinissent certains composants et se focalisent sur des fonctions ou sous systèmes spécifiques au niveau des architectures tels que le traitement de données, la collecte des données, le recrutement des participants, etc. L'architecture présentée dans (Christin et al., 2011; De Cristofaro & Soriente, 2011) se focalise sur des aspects non fonctionnels comme la confidentialité des données (privacy).

D'autres travaux se penchent plutôt sur l'aspect fonctionnel au niveau des architectures tels que le recrutement des participants (Reddy et al., 2010), la capture et le traitement des données (Reddy et al., 2010), la gestion des tâches, etc.

Selon (Fuchs-kittowski & Faust, 2014), l'architecture proposée dans certains cadres d'applications crowdsourcing comme MoCoMapps (Hupfer, 2012), Medusa (Ra et al., 2012), PRISM (Das et al, 2010) se sont penchés sur des services génériques tels que la sécurité, la protection des données, etc.

La figure 2.8 illustre certains composants dans une architecture crowdsourcing.



**Figure 2.8** Architecture détaillée d'une application crowdsourcing (Fuchskittowski & Faust, 2014)

### 2.9.20 Le crowdsourcing dans le domaine médical

Dans le secteur médical, le crowdsourcing a été utilisé dans plusieurs cas tels que la recherche médicale pour observer la fréquence de certaines maladies (Ranard et al., 2014) et pour examiner et analyser certains de maladies en utilisant des données provenant de sources d'imagerie médicale. Dans l'article (Mavandadi et al., 2012), il a été utilisé pour examiner des échantillons de sang dans le but de déterminer la présence ou l'absence d'infections paléudiennes.





## CHAPITRE III

### MISES EN ŒUVRE ET SIMULATION

#### **3.10 Méthodologie**

Pour réaliser la simulation, nous avons utilisé les modèles de mobilités utilisés dans les réseaux ad-hoc comme approche pour imiter le déplacement des individus. Nous avons utilisé l'outil de génération de mobilité appelé *Bonmotion* (BonnMotion, 2008). Nous avons intégré ces modèles dans un simulateur (en l'occurrence *Ns2*), en vue d'analyser et de visualiser le déplacement des noeuds à travers l'outil de visualisation *Nam* (Nam, 2002; Ns2, 2005). Nous avons utilisé un outil d'intégration et de traitements de données (*Pentaho, spoon*) pour traiter et intégrer les données de mobilité dans une base de données *MySQL* (MySQL, 2016; pentaho, 2005). Nous avons ensuite, utilisé un service Web pour manipuler et interpréter les résultats.

Par ailleurs, nous avons proposé une architecture exploitant le paradigme de crowdsourcing et de crowdsensing en vue de surveiller la propagation des maladies contagieuses.

#### **3.10.21 Les maladies infectieuses simulées dans le cadre de notre projet.**

La transmission des maladies infectieuses peut se faire de diverses manières :

- par le biais de contacts avec les individus infectés,
- par l'eau, l'air,
- par propagation à transmission vectorielle.

Dans le cadre de ce projet nous mettons plus l'accent sur certaines maladies infectieuses et contagieuses qui peuvent contaminer les individus lorsqu'ils sont rentrés en contact direct avec une personne infectée ou lorsque la personne infectée est capable de contaminer son environnement et présente des risques pour les autres se trouvant à une distance assez proche de lui. C'est le cas, par exemple, de maladies comme pour *l'Ebola*, le *choléra*.

### 3.10.22 Gestion de la transmission de la maladie dans notre simulation

Pour reproduire la transmission des maladies dans notre simulation, nous partons du principe SEIR selon laquelle la population est divisée en plusieurs états (*Sains*, *Exposés*, *Infectés* et *Guéris*). On suppose que tous les noeuds (individus) sont sains au début de la simulation. On introduit par la suite un noeud Infecté dans notre modèle. Ce noeud transmet l'infection aux autres noeuds suivant les caractéristiques de la maladie considérée pendant le déroulement de la simulation. Les informations relatives au noeud infecté et la maladie choisie sont passées en paramètres dans l'outil d'intégration de donnée qui fait le traitement et l'intégration des données dans la base. Le processus de traitement permet de déterminer si un éventuel cas de contamination ou un changement de statut est possible. Si c'est le cas, le statut du noeud en question est changé dans l'un des états suivants : *contaminé*, *infecté*, *guéri*.

Étant donné que les noeuds changent de position constamment durant le déroulement la simulation, pour chaque intervalle de temps ( $t = 0, t \leq t_{sim}^3$ ) on calcule la distance entre les noeuds en vue de déterminer si la position d'un noeud infecté influence l'état de santé des autres noeuds.

La distance à elle seule ne suffit pas pour déterminer si l'infection aura lieu ou pas. Par contre, on a également pris en considération les différents stades dans le processus d'infection.

---

<sup>3</sup>  $t_{sim}$  : représente la durée de la simulation.

### 3.10.23 Particularité des maladies simulées

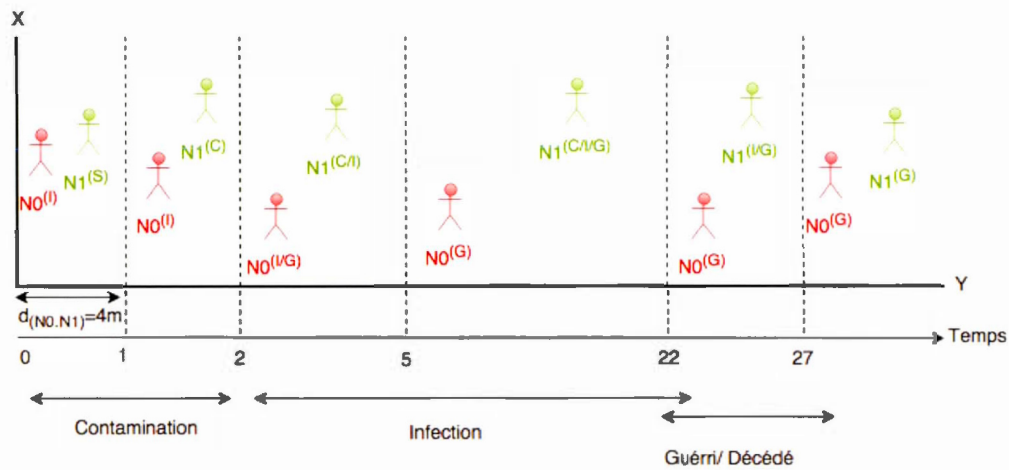
Nous avons intégré des paramètres à travers un calculateur de l'outil Pentaho pour chaque maladie simulée. Ces paramètres sont entre autres :

- a) *La Période d'incubation* : Elle détermine la durée qu'on peut passer avec le virus sans pouvoir transmettre l'infection. Elle dure deux (2) à vingt-deux (22) jours dans le cas de la maladie d'Ebola et de un (1) à trois (3) jours dans le cas du choléra par exemple.
- b) *La distance de contamination* : Elle détermine la portée pour qu'une contamination puisse être possible entre un individu infecté et sain. Elle est de un (1) à cinq (5) mètres dans le cas de la maladie d'Ebola et de un (1) à dix (10) mètres dans le cas de choléra.
- c) *La durée de survie* : Elle détermine le nombre de jours qu'on peut passer avec la maladie après le stade d'infection. Elle est de deux (2) à cinq (5) jours dans le cas de la maladie d'Ebola et de un à trois jours dans le cas de la maladie de choléra.

Le calculateur de l'outil Pentaho interagit avec la base de données *healthSim* via des requêtes SQL à travers des modules de Pentaho. Il fait le traitement des données et les mises à jour selon le changement de statut de santé des individus en fonction du temps.

Pour illustrer le cas de la maladie Ebola par exemple, considérons le cas de  $N = 2$  individus illustrés par la figure 3.1, dont il y a une personne ( $N0^I$ ) infectée et une autre ( $N1^{E/S}$ ) exposée ou saine à l'instant  $t = 0$ . La personne  $N0$  va contaminer la personne  $N1$  si elle se trouve à une distance inférieure ou égale à 5 mètres de lui dans ce même intervalle de temps. Nous avons incorporé certains paramètres à travers l'outil Pentaho pour calculer la distance entre les deux personnes ( $N0$  et  $N1$ ). Le

module qui calcule la distance est connecté à la base de données en vue de faire des calculs sur les coordonnées des noeuds.



**Figure 3.9** Chronologie de la maladie Ebola dans notre simulation

Après la phase de contamination, il va passer entre 2 à 22 jours avant que la personne N1 soit infectée. C'est-à-dire, qu'à l'instant  $t = (t_0 + t_{(2-22)})$  N1 va pouvoir retransmettre la maladie à d'autres personnes. Par ailleurs, nous avons utilisé des paramètres qui permettent de choisir aléatoirement un nombre dans l'intervalle d'incubation (2-22).

Une fois la personne N1 est infectée, elle va être guérie ou décédée suivant la durée de survie de la maladie après le stade d'infection, soit une période allant de 2 à 5 jours dans le cas de la maladie d'Ebola. Ceci est aussi valable pour la personne N0 et toutes les autres personnes dans la simulation. À l'instant  $t = (t_0 + t_{(2-5)})$  N0 va être guérie et pour la personne N1 ça va être à l'instant  $t = t_0 + t_{(2-22)} + t_{(2-5)}$ . De ce fait, nous avons paramétré certains modules de l'outil Pentaho tout en interagissant avec la

base de données à l'aide des requêtes, en vue de déterminer quand la personne va être guérie ou décédée.

Dans notre simulation, nous avons analysé en fonction du temps, le statut de chaque noeud par rapport aux différents stades de l'infection pour la maladie simulée. Cela nous permet de suivre la propagation de l'infection dans toute son intégralité et nous garantit que la transmission a respecté les différentes phases possibles dans le processus de transmission d'une infection.

On a découpé les tâches au niveau de l'outil d'intégration et de traitements de données, de manière à ce qu'un traitement ne soit pas trop lourd.

### 3.11 Simulation

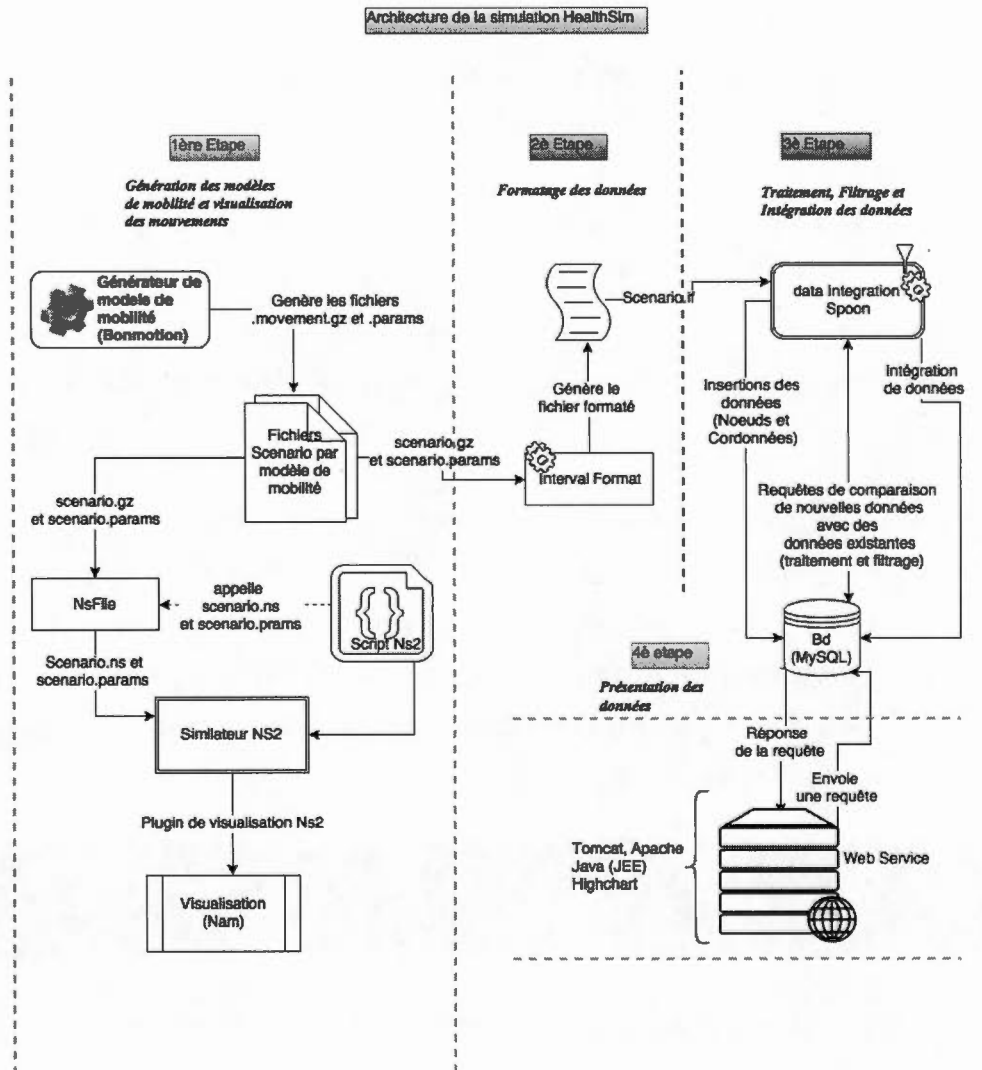
Dans cette section, nous détaillons notre procédure de simulation. Nous expliquons le modèle ainsi que les résultats obtenus. Ces résultats portent principalement sur 4 parties importantes: le générateur de modèle de mobilité, l'outil de simulation et de visualisation, l'outil d'intégration de données et le service Web utilisé.

### 3.12 Architecture de la simulation

L'architecture représentée par la figure 3.2 illustre l'interaction entre les différentes parties de notre système. Elle comprend 4 parties:

- 1) **Génération des modèles de mobilité et visualisation des mouvements** : Cette étape permet, grâce à l'outil *BonnMotion* de créer les fichiers de scénarios qui contiennent les informations sur les modèles de mobilité que nous voulons visualiser. *BonnMotion* est un outil qui permet de générer certains modèles de mobilité qu'on retrouve dans les réseaux *ad hoc*. Les fichiers générés sont ensuite utilisés par *Ns2* et *Nam* qui permettent respectivement de simuler et visionner le déplacement des nœuds.

- 2) **Formatage de données** : Cette étape consiste, à l'aide du module *Interval Format de Bonmotion* de formater les fichiers qu'il génère sous un format interprétable par l'outil d'intégration de donnée Pentaho (*Spoon*) (Pentaho, 2005) qui intègre de données. De plus, *Interval Format* permet de spécifier l'intervalle de temps pour afficher l'information sur les nœuds.
- 3) **Traitement, filtrage et intégration des données** : Cette étape permet de faire le traitement, le filtrage et l'intégration des données dans la base, à l'aide de l'outil d'intégration de données *Pentaho*. Le filtrage
- 4) **Présentation des données** : Cette étape consiste, de présenter à travers un *service web* les données enregistrées. Cela permet d'analyser et d'interpréter plus facilement les résultats obtenues lors des simulations.



**Figure 3.10** Architecture de la simulation HealthSim

### 3.13 Générateur et simulateur de modèle de mobilité

*BonnMotion* nous a permis de générer les modèles de mobilité. Cet outil nous permet d'obtenir les fichiers scénarios des modèles de mobilité que nous analysons avec NS2. Nous obtenons plus d'informations sur le comportement du modèle en question,

c'est-à-dire la position des nœuds, leur vitesse de déplacement ainsi que leur mode de déplacement à travers un simulateur de réseau.

### 3.13.24 BonnMotion

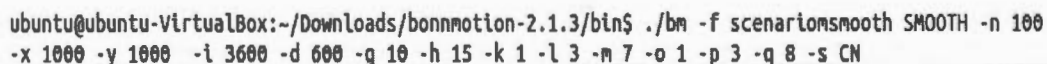
*BonnMotion* est un logiciel qui permet de générer et d'analyser certains modèles de mobilité. Il permet également l'exportation des fichiers scénarios vers de nombreux simulateurs de réseau. La figure 3.3 illustre la commande permettant de générer les fichiers scénarios pour le modèle de mobilité *RandomWaypoint* (Bai & Helmy, 2004).

Les paramètres utilisés permettent de définir le nombre de nœuds à prendre en considération (-n 150), la topologie de la simulation (-x 600 -y 600), la durée enlevé de la simulation (i - 3600) et la durée de la simulation (-d 600). D'autres paramètres peuvent s'ajouter aux paramètres précédents suivant le modèle de mobilité utilisé ou le type de scénario que nous voulons réaliser. La figure 3.4 est un autre exemple pour le modèle de mobilité *smooth* (Munjal & Navidi, 2011).



```
ubuntu@ubuntu-VirtualBox: ~
ubuntu@ubuntu-VirtualBox:~$ ./bm -f ScenarioRandomWaypoint_HealthSim RandomWaypoint -n 150 -x 600
-y 600 -i 3600 -d 600
```

**Figure 3.11** Génération des fichiers scénario du modèle de mobilité  
RandomWaypoint



```
ubuntu@ubuntu-VirtualBox:~/Downloads/bonnmotion-2.1.3/bin$ ./bm -f scenariomsmooth SMOOTH -n 100
-x 1000 -y 1000 -i 3600 -d 600 -g 10 -h 15 -k 1 -l 3 -m 7 -o 1 -p 3 -q 8 -s CN
```

**Figure 3.12** Génération des fichiers scénario du modèle de mobilité Smooth

#### 3.13.24.6 Interval Format

*Interval Format* est une librairie de *BonnMotion* qui permet de spécifier l'intervalle de temps affichant la position des nœuds. Par exemple, nous pouvons demander au



système d'afficher les informations sur les nœuds toutes les 2 secondes ou suivant un intervalle de notre choix. La figure 3.5 illustre la commande pour lancer la librairie *Interval Format* dans le cadre d'un intervalle de 1 seconde. Un autre aspect de cet outil est qu'il permet de formater les fichiers scénarios générés par *BonMotion* sous un format interprétable pour l'outil d'intégration de données Pentaho, plus précisément le module *spoon* de cet outil utilisé pour alimenter la base de données *MySQL*. Nous détaillons l'utilisation de l'outil *spoon* dans la section 3.5.

```
ubuntu@ubuntu-VirtualBox:~$ ./bm IntervalFormat -f ScenarioRandomwaypointHealthSim -l 1 -s|
```

**Figure 3.13** Commande *Interval format*

### 3.13.25 Ns2

*Ns2* est un simulateur de réseaux informatiques permettant d'étudier et d'analyser la performance des protocoles de routage dans les réseaux Manet (*Ns2*, 2005). Il utilise les informations relatives aux nœuds tels que leurs coordonnées et leurs vitesses dans le but d'étudier leurs déplacements dans l'espace de simulation. Ces informations peuvent être rentrées dans le script *NS2* de manière statique s'ils ne sont pas nombreuses. Sinon, il faut les passer en paramètre en mentionnant le chemin du fichier contenant ces informations tel qu'illustré dans la figure 3.8. Dans notre cas, nous avons utilisé la librairie *NSFile* de *BonnMotion* qui permet de générer des fichiers pour *NS2* à partir des fichiers de scénarios générés par *BonMotion*. Fig. 3.6

La figure 3.7 illustre le format de données qui se trouve dans le fichier généré par *NSFile* pour *Ns2*. Par exemple, nous retrouvons les informations telles que le nom des nœuds, les coordonnées et la vitesse de déplacement des nœuds.

```
NSFile done.  
Runtime: 2 sec  
ubuntu@ubuntu-VirtualBox:~/Downloads/bonnmotion-2.1.3/bin$ ./bm NSFile -f ScenarioRandomwalk 2016
```

**Figure 3.6** Commande permettant de générer les fichiers pour *Ns2*

```

Scenariorandomwalk_2016.ns_movements
1 $node_(0) set X_ 95.09802266642308
2 $node_(0) set Y_ 144.15688360498922
3 $ns_ at 0.0 "$node_(0) setdest 98.77607256557313
  134.5607813500823 1.0276829742058953"
4 $ns_ at 10.0 "$node_(0) setdest 107.28711108164175
  146.5847033165917 1.4731343322274157"
5 $ns_ at 20.0 "$node_(0) setdest 112.28971757292724
  157.05404215761055 1.1603151618190526"
6 $ns_ at 30.0 "$node_(0) setdest 117.55572787402119
  156.414881775929 0.5304657433306965"
7 $ns_ at 40.0 "$node_(0) setdest 116.68314719104022
  158.82749067782319 0.566005616595133"

```

**Figure 3.14** Format du fichier Ns2

La figure 3.7 présente les informations sur les paramètres de la simulation tels que le nombre de noeuds, la durée de simulation, la topographie utilisée et la distance de contamination.

```

set opt(nn)          50                ;# number of mobile nodes
set opt(x)          400               ;# x-dimension of the topography
set opt(y)          400               ;# y-dimension of the topography
set opt(tr)         temp.rands
set opt(stop)       600.0             ;# simulation time
set opt(seed)       1
set opt(movements) /home/ubuntu/Downloads/bonnmotion-2.1.3/bin/Scenariorandomwalk_2016.ns_movements
set opt(dCont)      50                ;# Distance de Contamination

```

**Figure 3.15** Chemin pour téléverser le fichier mouvement ns2 et Informations sur certains paramètres de la simulation

### 3.13.25.7 Résultats

La figure 3.9 illustre les résultats obtenus lors de la simulation. L'ensemble de nœuds qui sont infectés au cours de la simulation de la propagation des maladies est affiché. D'autres informations sont également indiquées comme l'identité des nœuds contaminés et la distance de contamination en fonction du temps.

```

ubuntu@ubuntu-VirtualBox:~/Downloads/bonnmotion-2.1.3/bin$ ns ScenarioRandomwalk_2016.tcl
Seeding RNG with 1
num_nodes is set 5
num_nodes is set 5
INITIALIZE THE LIST xListHead
Loading movement file /home/ubuntu/Downloads/bonnmotion-2.1.3/bin/trime5.ns_movements...
Starting Simulation...
les noeuds qui vont etre contamines sont:

T Na Nb dst:
0 0 1 44
0 0 3 49
1 0 1 44
1 0 3 49
SORTING LISTS ...DONE!
2 0 1 44
3 0 1 44
4 0 1 45
5 0 1 45
6 0 1 45
7 0 1 45
8 0 1 47
9 0 1 47
10 0 1 48
11 0 1 48
12 0 1 50
13 0 1 50
ubuntu@ubuntu-VirtualBox:~/Downloads/bonnmotion-2.1.3/bin$

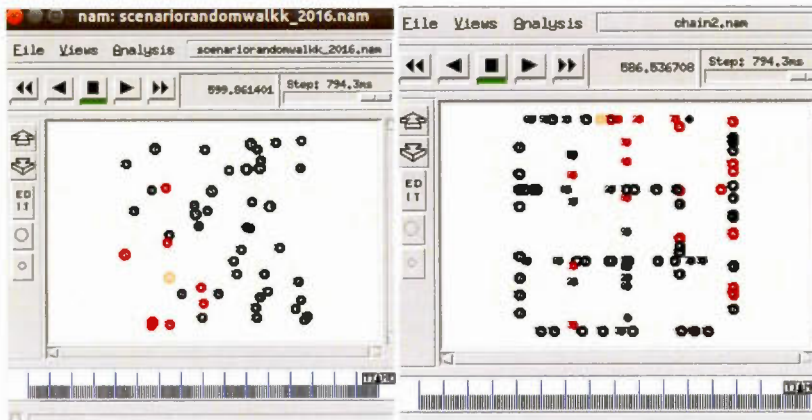
```

Figure 3.16 Résultats de la simulation ns2.

la figure 3.9 affiche les noeuds infectés pour une distance de contamination inférieure à 51 m. La première colonne affiche le temps, la deuxième affiche le noeud infecté, la troisième affiche les noeuds contaminés en fonction du temps et de la distance de contamination et la quatrième affiche la distance de contamination.

### 3.13.26 Nam

L'outil *Nam* permet de visualiser en temps réel le déplacement des nœuds et de suivre l'évolution de la propagation de l'épidémie dans notre projet. La figure 3.10 illustre les résultats de simulation réalisée pour deux scénarios différents, soit : *RandomWaypoint* et *ManhattanGrid* (Roy, 2010).



**Figure 3.17** Visualisation de la propagation de l'épidémie pour les modèles de mobilité *RandomWaypoint* et *ManhattanGrid*.

### 3.14 Intégration de données

Pour alimenter la base de données, nous avons utilisé le module *spoon* de l'outil Pentaho dans le but d'intégrer les données. Les fichiers obtenus lors de la simulation contiennent plusieurs données. En conséquence, l'entrée manuelle de ces informations dans la base est compliquée. Étant donné que le volume de données à traiter est élevé, l'utilisation de l'outil *Pentaho* effectue l'intégration de données automatiquement. Cet outil nous a permis de filtrer, de traiter et d'intégrer les données avant de les rendre accessibles à partir du service web. En revanche, le temps de réponse a diminué considérablement du côté du serveur parce que le traitement des données se fait plutôt du côté de l'outil d'intégration de données, Pentaho.

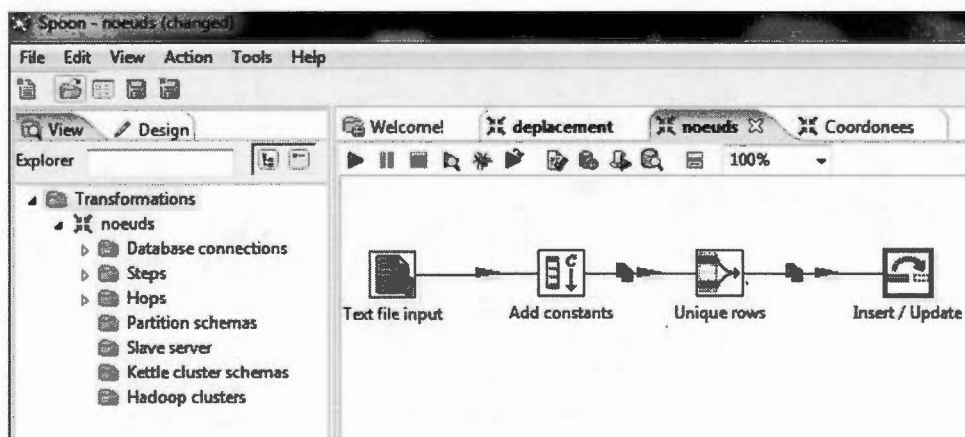
Afin de simplifier la tâche d'intégration des données, nous avons réalisé le processus en trois étapes :

- *La première étape* consiste à stocker dans la base de données des informations sur les nœuds, c'est-à-dire les individus tels que le nom des nœuds et leur temps de déplacement. Certaines de ces informations sont obtenues à partir du

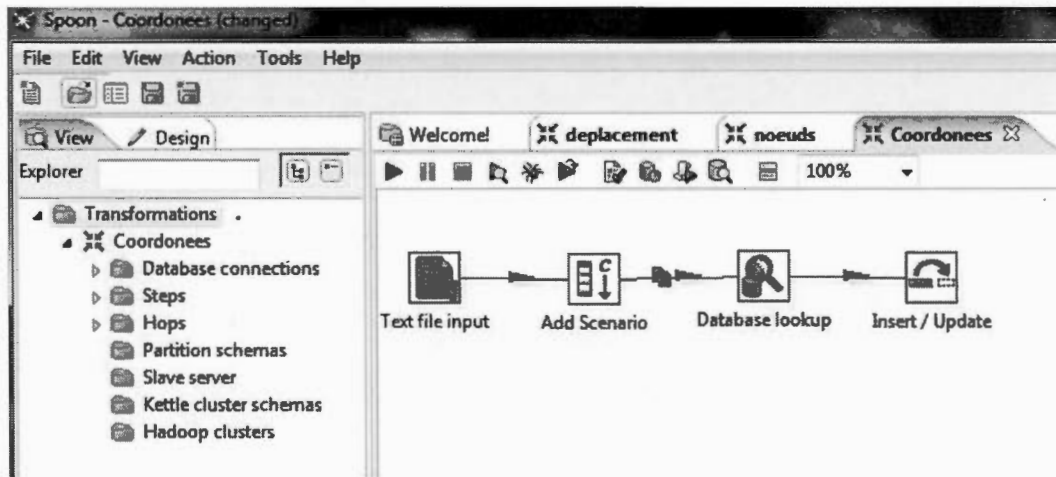
fichier de formatage *scenario.if* qui est créé par la librairie *Interval Format* tel qu'illustré dans l'architecture par la figure 3.2.

- La deuxième étape consiste à stocker les coordonnées des nœuds dans la base de données.
- La troisième étape consiste à utiliser certaines informations que nous passons en paramètre dans l'outil d'intégration de données dans le but d'intégrer de nouvelles informations telles que les *nœuds infectés*, *guéris* et *sains*. Ce processus est possible en fonction des résultats obtenus lors du filtrage et des calculs des données existant dans la base de données avec celles passées en paramètre au niveau de l'outil d'intégration de données.

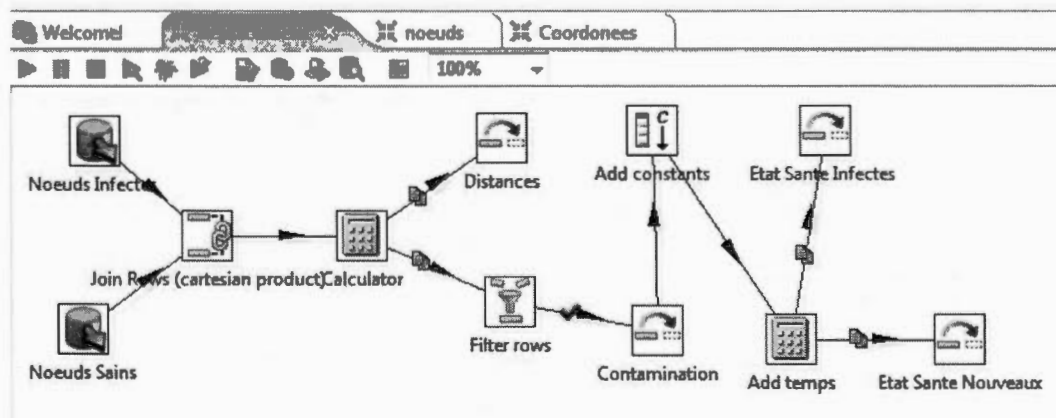
Les figures 3.11, 3.12 et 3.13 illustrent le processus d'insertion, de traitement et de filtrage des données dans la base.



**Figure 3.18** Première étape : processus d'insertion des nœuds (individus) dans la base de données



**Figure 3.19** Deuxième étape : processus d'insertion des coordonnées dans la base de données



**Figure 3.20** Troisième étape : processus de traitement et de filtrage de données dans la base

### 3.14.27 La base de données

La figure 3.14 illustre la conception de la base de données *HealthSim*, plus précisément les relations entre les tables. La base de données est composée de 11 tables, soit :

- *sst\_statut\_sante*: contient les informations sur l'état de santé des individus.
- *trs\_transition*: contient les informations reliées aux changements d'état de santé des individus
- *est\_etat\_sante*: contient les informations sur les différents états de santé possibles pour un individu.
- *cap\_capteur*: contient les informations sur les capteurs utilisés.
- *sce\_scenario*: contient les informations sur les différents scénarios du modèle de mobilités.
- *nod\_noeud*: contient l'identité des noeuds d'un modèle de mobilité spécifique.
- *coo\_cooronnees*: contient les informations sur les coordonnées des noeuds.
- *mdl\_modele*: contient les informations sur les modèles de mobilité.
- *mld\_maladie*: contient les informations sur les différents types de maladies.
- *dpc\_deplacement*: contient les informations sur la distance entre les noeuds.
- *con\_contamination*: contient les informations sur la transmission des maladies.

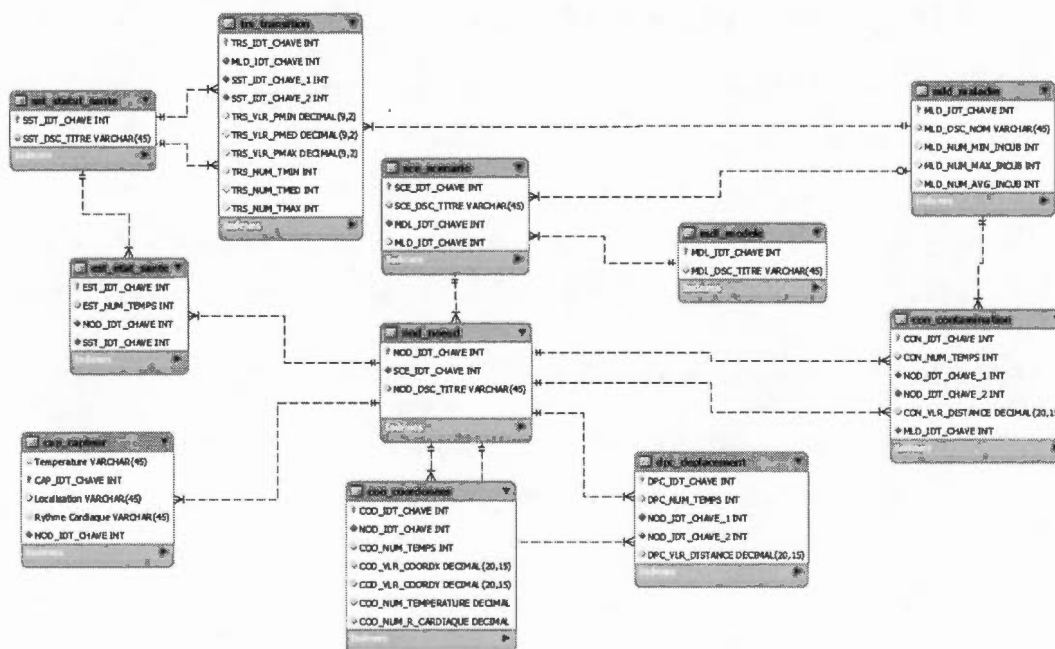


Figure 3.21 Schéma de la base de données *HealthSim*.

### **3.15 Représentation des données à travers un service web**

Nous avons utilisé un service web dans le but d'exploiter et de représenter les données obtenues lors des simulations. Il nous permet surtout, d'interpréter les différents tendances et fluctuations au niveau des individus infectés pour un scénario<sup>4</sup> choisi et de faire une comparaison pour différents scénarios en fonction des résultats obtenus lors des simulations.

Nous avons utilisé le serveur Web *Tomcat Apache* (Foundation, 2016). Nous avons implémenté le service web en *Java* (Java, 2016) et utilisé *Highchart* (Highchart, 2016) pour représenter les résultats graphiquement sous un format plus représentatif.

#### **3.15.28 Description des différentes fenêtres du service web**

Notre service web est connecté à une base de données qui contient les données résultantes suite aux processus de traitement, de filtrage et d'insertion. Dans le but de faciliter la compréhension des informations présentées dans le système *HealthSim*, une description des différentes parties du système est donnée dans les sections qui suivent.

##### **3.15.28.8 Portail *HealthSim***

Pour y accéder au portail *HealthSim*, l'utilisateur doit avoir un nom d'utilisateur et un mot de passe. La connexion au portail est illustrée par la figure 3.15.

---

<sup>4</sup> Un scénario comprend : un modèle de mobilité, une maladie, une topographie, des individus et une durée simulation.





Figure 3.22 Portail HealthSim.

Une fois connecté au système, une brève description sur le système est présentée. Le menu se trouve à gauche dans la partie supérieure tel qu'illustrée par la figure 3.16.

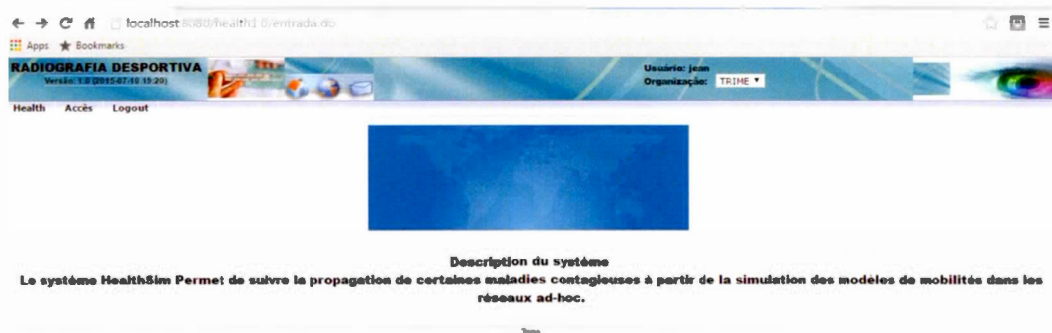


Figure 3.23 Portail *HealthSim* suite à une connexion valide.

### 3.15.28.9 Menu du Système

Le système est composé principalement de trois menus : *Health*, *Accès* et *Logout*. Le menu *Health* est la partie la plus importante du système. Il liste les différents sous-menus du système permettant d'exploiter le comportement des résultats de la simulation. Le menu *Accès* permet d'attribuer les différents droits à un utilisateur. Nous pouvons restreindre l'accès ou en garantir à un utilisateur. Nous pouvons également ajouter ou supprimer un utilisateur. Le menu *Logout* permet de sortir du système. La figure 3.17 illustre ces différents menus.

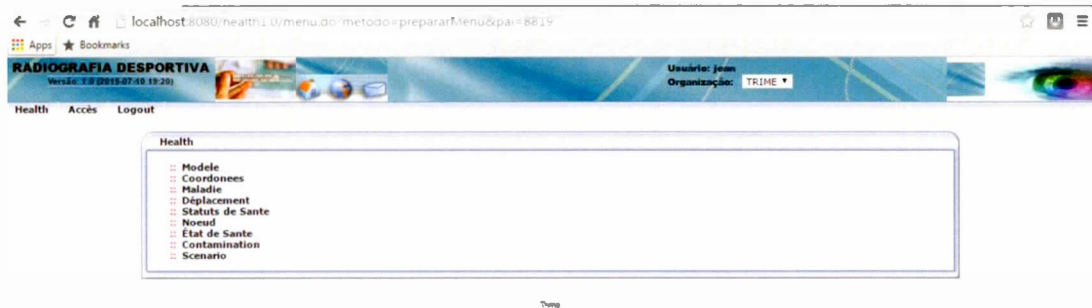


Figure 3.24 Menu du système

### 3.15.29 Les différents sous-menus

Le système est composé de 9 sous-menus que nous détaillons dans cette section.

#### 3.15.29.10 Modèle

Le sous-menu *modèle* permet de lister les différents modèles de mobilité utilisés pour la simulation. De plus, elle permet de supprimer, de créer ou de rechercher un modèle de mobilité spécifique et de consulter les informations relatives au modèle choisi en cliquant sur l'icône de recherche. La figure 3.18 illustre le sous-menu *modèle*.

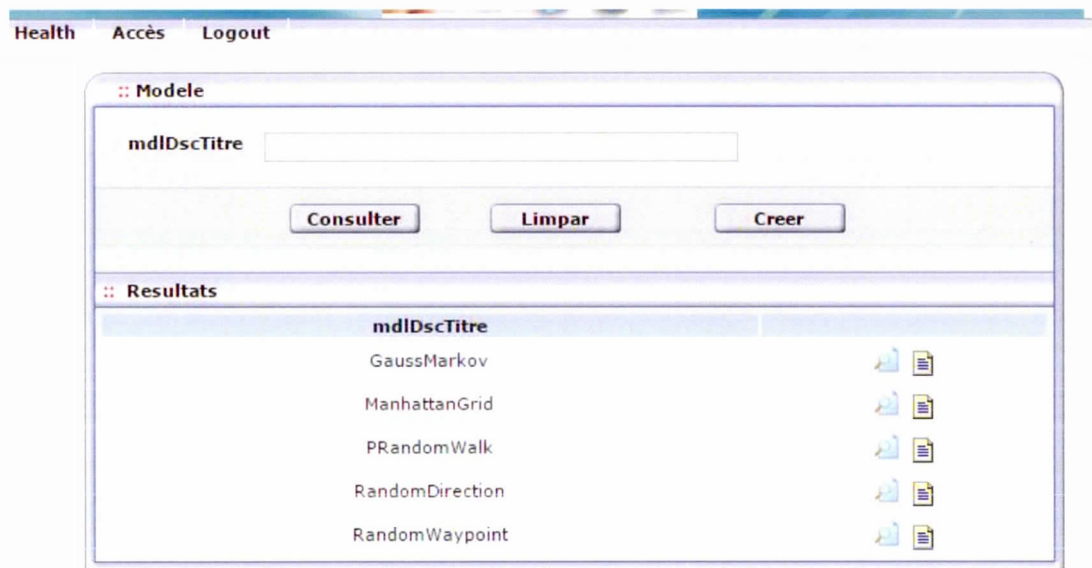


Figure 3.25 Modèle de mobilité

### 3.15.29.11 Coordonnées

Le sous-menu *coordonnées* permet de visualiser les coordonnées de chaque individu dans le temps. Il permet aussi de visualiser les informations relatives à chaque nœud ou à un nœud spécifique. La figure 3.19 présente un exemple d'utilisation du sous-menu *Coordonnées* pour le scénario *RandomWaypoint*.

nodIdtChave	cooNumTemps	codVirCoordx	codVirCoordy
0	0	106.91	182.38
0	1	106.91	182.38
0	2	106.91	182.38

Figure 3.26 Coordonnées

### 3.15.29.12 Maladie

Le sous-menu *Maladie* permet d'afficher la liste des maladies utilisées dans la simulation. Nous pouvons également visualiser dans quel scénario nous l'avons utilisé, quels sont les individus qui ont été infectés par cette maladie et à quelle

distance la contamination a été faite. Les figures 3.20 et 3.21 listent les maladies et affichent les informations relatives à la maladie Ebola.

Health Accès Logout

**:: Maladie**

mIdDscNom

---

**:: Resultats**

mIdDscNom
Ebola

Figure 3.27 Liste des maladies.

Health Accès Logout

**:: Maladie**

mIdDscNom: Ebola

**:: maladie x contamination**

conNumTemps	nodIdtChave1	nodIdtChave2	conVirDistance	
0	Q	Z	30.75	📄 📄
1	Q	8	27.96	📄 📄
1	Q	58	49.36	📄 📄
1	Q	95	43.75	📄 📄
1	Z	8	9.31	📄 📄
1	Z	25	41.34	📄 📄
1	Z	53	25.24	📄 📄

Figure 3.28 Information relative à la maladie Ebola.

### 3.15.29.13 Déplacement

Le sous-menu *Déplacement* permet d'afficher la distance entre deux individus dans le temps. Nous pouvons choisir d'afficher la distance entre tous les individus dans la simulation en spécifiant les individus pour lesquels nous voulons afficher la distance telle qu'illustrée par la figure 3.22.

dpcNumTemps	nodIdtChave1	nodIdtChave2	dpcVirDistance
1	0	1	50.16

Figure 3.29 Déplacement

### 3.15.29.14 Statut de santé

Le sous-menu *Statut de santé* donne la possibilité d'afficher les différents états de santé possibles, à savoir : sains, infecté et guéri. Nous pouvons à travers l'icône de recherche afficher l'information pour un état de santé spécifique, c'est-à-dire l'ensemble des individus qui sont concernés par ce statut de santé. Les figures 3.23 et 3.24 illustrent les informations pour un individu dont son statut de santé est infecté.

Figure 3.30 Statut de santé.

estNumTemps	nodIdtChave	
1	0	
1	2	
0	0	
2	8	
2	0	
2	58	
2	95	
2	2	
2	53	

Figure 3.31 Statut de santé des individus en fonction d'un état de santé particulier

### 3.15.29.15 Les Nœuds

Le sous-menu *Noeud* permet d'afficher l'ensemble des nœuds de la simulation pour tous les scénarios existants. Il permet de choisir un scénario particulier et d'afficher les informations relatives aux nœuds de ce scénario telles que le temps, la maladie et la distance d'un nœud par rapport à un autre. La figure 3.25 et 3.26 illustrent la liste des nœuds pour les différents scénarios et les informations concernant les nœuds contaminés, respectivement.

The screenshot shows a web application interface. At the top, there is a navigation bar with the text "Health", "Accès", and "Logout". Below this, the main content area is titled "Noeud". It contains a form with a "Scenario" dropdown menu set to "ScenarioPRandomwalk" and a "Nom" text input field. Below the form are three buttons: "Consulter", "Limpar", and "Creer". Underneath the buttons is a section titled "Resultats" which contains a table. The table has two columns: "Scenario" and "Nom". The rows of the table are as follows:

Scenario	Nom
ScenarioPRandomwalk	0
ScenarioPRandomwalk	1
ScenarioPRandomwalk	10
ScenarioPRandomwalk	11
ScenarioPRandomwalk	12
ScenarioPRandomwalk	13
ScenarioPRandomwalk	14

Figure 3.32 Liste des nœuds pour les différents scénarios.

Health Accès Logout

:: Noeud

Scenario: Scenario\_RandomWaypoint

Nom : 0

---

:: noeud x contamination 1

conNumTemps	nodIdtChave1	nodIdtChave2	conVlrDistance	midIdtChave		
0	0	7	30.75	Ebola		
1	0	8	27.96	Ebola		
1	0	58	49.36	Ebola		
1	0	95	43.75	Ebola		

---

:: noeud x contamination 2

Figure 3.33 Informations sur les nœuds contaminés.

### 3.15.29.16 États de santé

Le sous-menu *Etats de santé* permet d'afficher l'état de santé de tous les individus lors de la simulation. Il permet également d'afficher l'état de santé des individus en fonction d'un statut de santé particulier, soit : sain, infecté ou guéri. La figure 3.27 illustre l'état de santé des nœuds.

Health Accès Logout

:: EtatSante

estNumTemps

nodIdtChave

sstIdtChave

---

:: Resultats

estNumTemps	nodIdtChave	sstIdtChave		
1	0	Infecté		
2	0	Infecté		

Figure 3.34 État de santé des nœuds.



### 3.15.29.17 Contamination

Le sous-menu *Contamination* permet d'afficher un ensemble d'informations sur les individus qui sont infectés comme le nœud qui a provoqué la contamination, le temps où la contamination a été faite, la distance de contact pendant la transmission de l'infection et la maladie utilisée pour simuler la propagation de l'épidémie. Il est également possible d'avoir de plus amples informations sur le nœud infecté ou le nœud qui a provoqué l'infection en cliquant sur ce dernier. Ces informations pourraient être les coordonnées, le temps ou le scénario utilisé pour le nœud en question. La figure 3.28 présente les informations sur les noeuds infectés au cours d'une contamination de la maladie Ebola.

Healthi Accès Logout

**:: Contamination**

conNumTemps

nodIdtChave1

nodIdtChave2

conVirDistance

mldIdtChave

---

**:: Resultats**











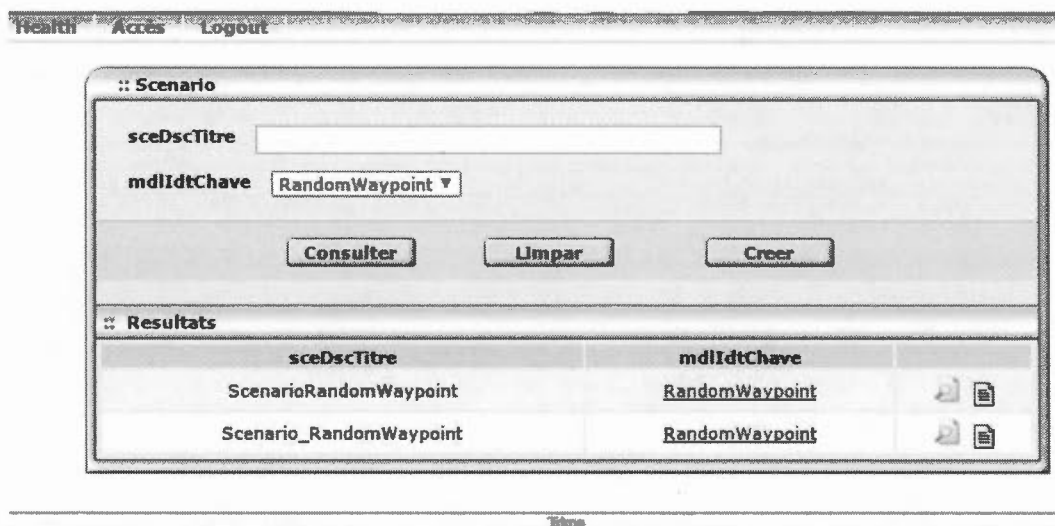
conNumTemps	nodIdtChave1	nodIdtChave2	conVirDistance	mldIdtChave	
0	0	7	30.75	Ebola	 
1	0	58	49.36	Ebola	 
1	0	8	27.96	Ebola	 
1	0	95	43.75	Ebola	 
1	7	53	25.24	Ebola	 

Figure 3.35 Information sur les noeuds infectés au cours d'une contamination.

### 3.15.29.18 Scenario

Le sous-menu *Scénario* permet d'afficher les différents scénarios utilisés dans la simulation. Nous rappelons qu'un scénario est composé d'un modèle de mobilité (*RandomWaypoint*, *Randomwalk*, etc.), d'un ensemble d'individus ( $n= 50, 100\dots$ ) et d'une maladie spécifique (Ebola, etc.). Donc, un scénario peut utiliser plusieurs modèles de mobilité ou des maladies différentes. En d'autres termes, un même modèle de mobilité peut être utilisé dans plusieurs scénarios différents et il en est de même pour une maladie. Le comportement de la propagation de la maladie dans deux scénarios différents utilisant les mêmes modèles de mobilité pourrait avoir des différences majeures si nous changeons le nombre de nœuds. La figure 3.29 présente la liste des scénarios utilisant le modèle *Random Waypoint*.



**Figure 3.36** Liste des scénarios utilisant le modèle *Random Waypoint*.

## CHAPITRE IV

### MODÈLE D'INTÉGRATION DE CROWDSOURCING ET CROWDSENSING À HEALTHSIM

#### 4.16 Modèle proposé

Nous proposons un modèle qui intègre les concepts de *crowdsourcing* et *crowdsensing* au modèle de simulation *HealthSim* (Orival, 2017) que nous avons présenté dans le chapitre 3.

Le crowdsourcing est un service Web qui permet de tirer profit de la participation en ligne d'un grand nombre d'individus pour résoudre un problème complexe. Le problème à résoudre est parfois divisé en plusieurs tâches dans le but de profiter de la compétence de chaque individu. Les individus choisissent leurs tâches et fournissent leur réponse en s'inscrivant sur une plateforme crowdsourcing qui publie les tâches à compléter. Le crowdsensing quant à lui est une technique qui permet de capter les informations à partir des téléphones intelligents ou par l'intermédiaire des capteurs internes ou externes connectés à un réseau (An, Gui, Wang, Yang, & He, 2015; Ganti et al., 2011).

L'objectif est de présenter un modèle qui permet de surveiller la propagation des maladies infectieuses à l'aide des modèles de mobilité simulés dans le cadre de ce travail en intégrant les paradigmes de *crowdsourcing* et *crowdsensing* à travers des dispositifs mobiles.

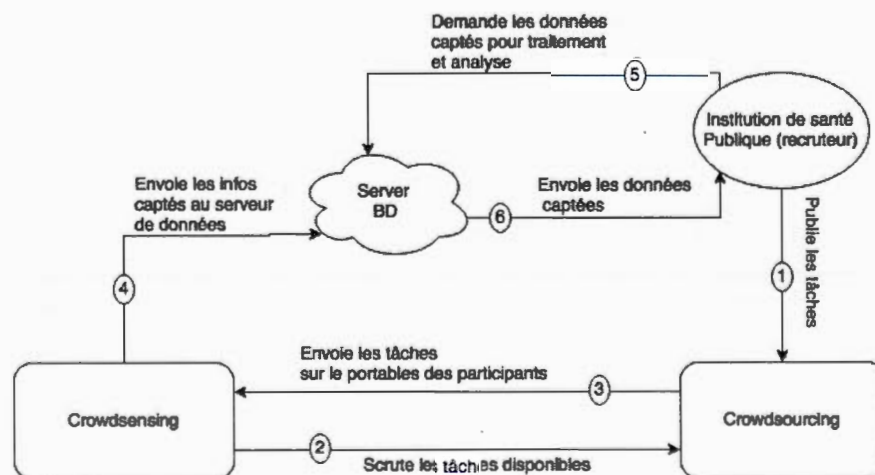
Avec le mécanisme de simulation que nous avons présenté au chapitre 3, nous obtenons les tendances de la propagation de la maladie dans une région donnée.

Les paradigmes de *crowdsourcing* et *crowdsensing*, nous permettent d'obtenir en temps réel des informations de santé des patients.

L'intégration du paradigme de *crowdsensing* à notre simulation, nous permet de capter les informations (*température corporelle, rythme cardiaque, lieu géographique, pression artérielle*) en temps réel sur l'état de santé des patients à l'aide, entre autres, de capteurs intégrés dans les mobiles.

Le paradigme de *crowdsourcing* nous permet de recruter des patients qui doivent soumettre les informations de leurs santé à une partie tierce telle qu'une institution médicale responsable de surveiller l'évolution des maladies contagieuses des patients. L'intégration de ces paradigmes permet de réduire considérablement le coût de collecte des données.

La figure 4.1 illustre de manière globale le modèle proposé.



**Figure 4.37** Architecture générale du modèle proposé.

#### 4.16.30 Méthodologie

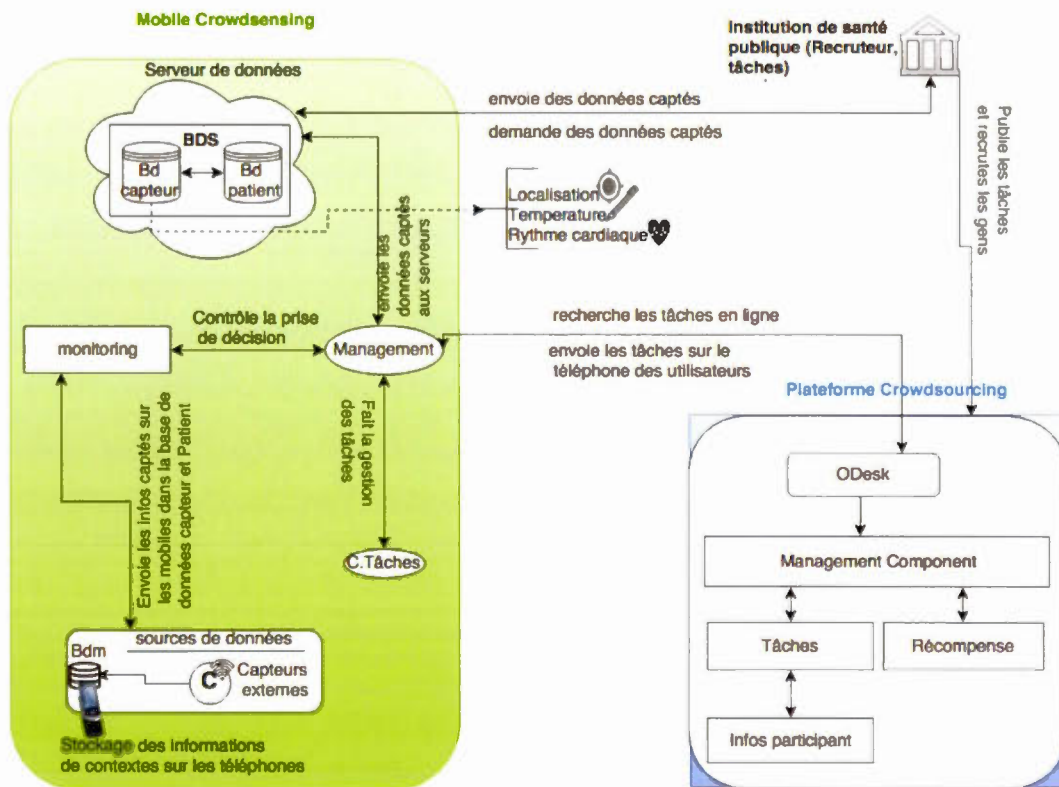
Nous avons choisi d'utiliser *ODesk* (Odesk, 2016), (Elance, 2016) comme plateforme *crowdsourcing* pour recruter les participants. Ces plateformes ont été également utilisées dans le travail de (Welbourne et al, 2014).

Lorsqu'un risque de contamination de maladie infectieuse est présent dans une région, une institution de santé publique définit les tâches à compléter et les affiche à travers la plateforme *ODesk*. Seules les personnes habitant dans la région suspecte peuvent voir l'affichage de la tâche à effectuer, car la plateforme *ODesk* (Elance, 2016; Odesk, 2016) permet de définir la région pour laquelle la publication sera visible. La description du travail, les termes du contrat, ainsi que le montant pour effectuer le travail, sont inclus dans cette publication. Le montant pour effectuer le travail est identique pour tous les participants.

Une fois qu'un participant a accepté l'offre d'emploi et qu'il est employé par le recruteur, il est invité à installer l'application mobile *HealthSense* sur son téléphone portable dont une partie de cette application a été créée à partir du *framework Funf* (Behavio, 2016), (Madan et al. 2012). Via l'application mobile *Healthsense*, les participants collectent les données à partir des capteurs intégrés dans leur téléphone mobile ainsi que d'autres capteurs externes (capteurs de pression artérielle, température corporelle, rythme cardiaque) qui communiquent les données par Bluetooth au téléphone. Les données collectées sont enregistrées dans la base de données *Sqlite* du téléphone, pour ensuite être envoyées au serveur lorsqu'il y a une connexion wifi ou 3G. Les participants doivent effectuer les tâches de captage 3 fois par jour et doivent être connectés pour une période d'au moins de 20 minutes.

La fréquence de captage peut être augmentée avec l'intensité de l'épidémie. Les participants qui ont déjà complété une tâche dans le passé reçoivent les nouvelles tâches via la méthode *push* ou ils peuvent à volonté choisir les tâches à compléter s'ils n'ont pas choisi de recevoir les tâches de cette manière-là.

La *figure 4.2* illustre les différents composants du modèle proposé et la *figure 4.3* illustre l'intégration du modèle proposé à l'architecture *Healthsim*.



**Figure 4.38** Architecture détaillée du modèle proposé  
(Crowdsourcing/Crowdsensing)

Architecture d'intégration Crowdsourcing/Crowdsensing mobile à notre simulation

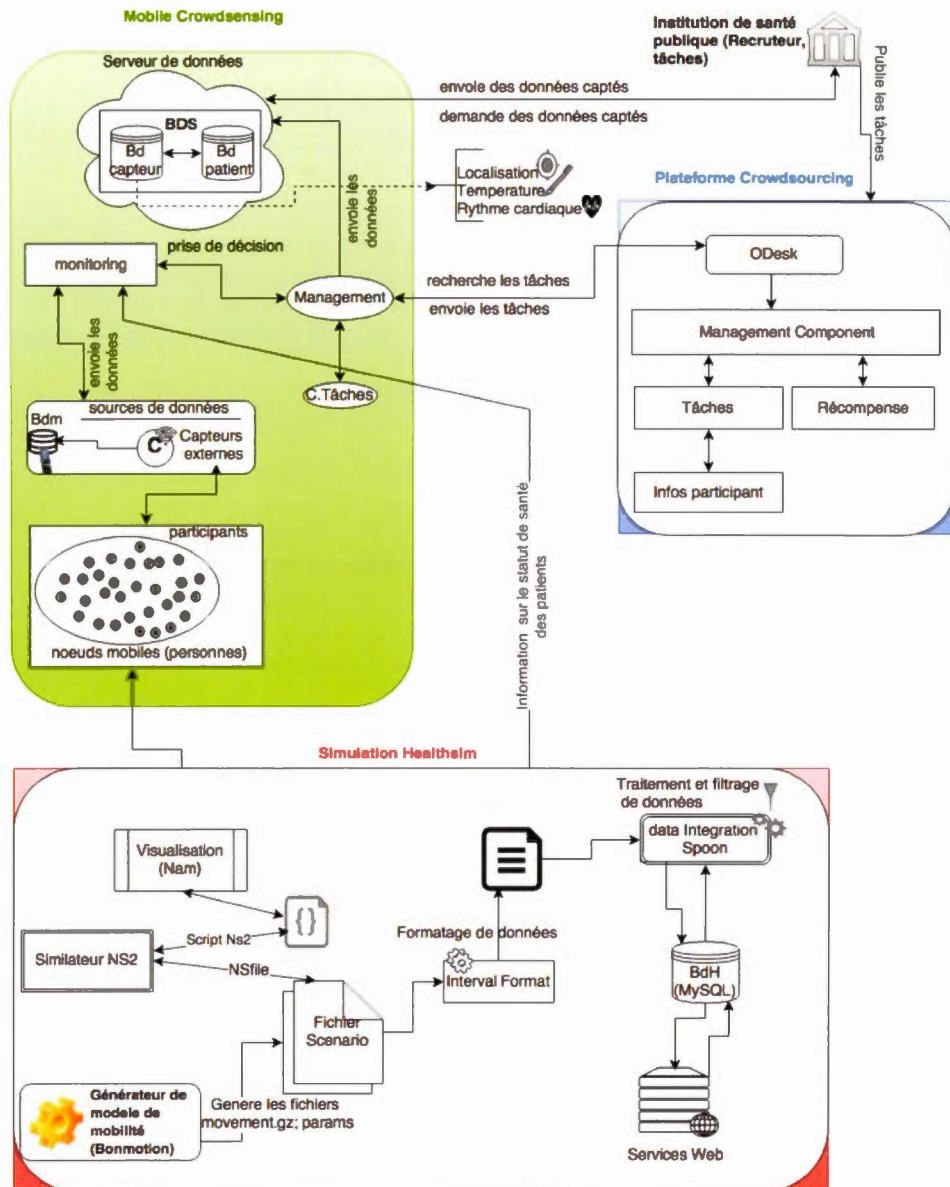


Figure 4.39 Architecture d'intégration du modèle proposé à Healthsim

Le modèle proposé se compose de différentes parties que nous expliquons dans les sous-sections qui suivent.

#### **4.16.31 Simulation *HealthSim***

L'architecture détaillée de *HealthSim* se trouve à la *figure 3.2*. Dans le modèle proposé, nous illustrons les composants principaux de cette architecture qui se trouve dans l'entité *HealthSim* de la *figure 4.3*.

Nous rappelons les éléments essentiels de la simulation *HealthSim* comme suit :

- 1) *Modèle de mobilité* : permet de simuler le déplacement des noeuds mobiles dans un réseau MANET. Nous avons identifié trois 3 types de noeuds mobiles, soit : infectés, sains et guéris. Ces noeuds se déplacent dans un espace de simulation en fonction du modèle de mobilité choisi.
- 2) *Service Web* : permet la communication et l'échange de données avec la base de données. Il interroge la base de données sur l'évolution d'épidémie pour le modèle de mobilité choisie et le type de maladie simulé ainsi que l'état de santé des patients concernant la propagation de l'épidémie.
- 3) *Base de données* : stocke les informations sur l'état de santé des noeuds mobiles, les paramètres de modèles de mobilités et les informations sur les maladies simulées.

#### **4.16.32 Module *crowdsourcing***

Le module *crowdsourcing* permet à l'instance *requérante* (comme une institution de santé publique, par exemple) de publier en ligne les tâches à effectuer ainsi que les termes de ce contrat (temps requis, processus de paiement, les accords légaux). Les différentes tâches à exécuter sont accessibles à travers la plateforme *ODesk*.



Dans notre architecture, la partie crowdsourcing est reliée à la partie *crowdsensing* à l'aide du module gestion de *crowdsensing*. Outre les différentes fonctionnalités (gérer les tâches, etc.) de ce dernier, il est également utilisé comme intermédiaire pour connecter les deux plateformes (*crowdsensing et crowdsourcing*), et permet aussi la communication avec l'application mobile *Healthsense* en vue de capturer certaines informations sur les individus qui veulent exécuter une tâche.

La plateforme *Odesk* fournit un service en ligne permettant à un large groupe d'individus de visualiser et de compléter certaines tâches créées par une tierce partie.

L'avantage d'utiliser *Odesk* par rapport aux autres plateformes (Amazon mechanical turk) se résume aux points suivants (Welbourne et al. , 2014) :

- a) Elle est mieux adaptée pour les tâches qui doivent s'exécuter à long terme sur les plateformes mobiles.
- b) Elle permet d'assigner les tâches aux participants dans un contexte spatial, c'est-à-dire qu'elle permet de limiter la visibilité des publications pour des régions particulières. Ce qui est très important dans notre cas, car la surveillance de la propagation des maladies contagieuses dans notre modèle se fait dans un contexte régional. Donc les participants qui auront à compléter une tâche doivent obligatoirement se trouver dans la région dans laquelle on fait la surveillance.

#### **4.16.32.19 Mode fonctionnement de la plateforme ODesk.**

La plateforme *Crowdsourcing* est facile à utiliser. Le recruteur affiche la description du travail à faire, le temps requis pour le compléter et les processus légaux. Il ajoute ensuite le travail dans une catégorie sur la plateforme avec un tag qui décrit le type de travail. Le recruteur publie ensuite l'annonce du travail et invite les participants à y appliquer. Le participant qui désire participer, répond à l'annonce en précisant une

enchère pour compléter le travail. Dans notre modèle proposé, le participant aura à télécharger l'application mobile leur permettant de compléter la tâche qui est celle de collecter les données.

#### 4.16.33 Module *crowdsensing*

Le module *crowdsensing* consiste à profiter de la coopération des utilisateurs ainsi que des propriétés de détection des appareils mobiles et autres dispositifs portables, en vue d'acquérir des connaissances locales du milieu environnant de l'individu ou sur l'individu lui même. Ce module permettra de tirer avantage de la disponibilité des individus pour effectuer des tâches de détection à travers leurs appareils mobiles intelligents en vue de mettre les données captées à l'aide de capteurs portés à la disposition d'une institution responsable de surveiller la propagation des maladies contagieuses . Dans le module *crowdsensing*, nous utilisons les entités suivantes :

##### a) *Application Healthsense*

L'application *Healthsense* est créée en partie à partir du *framework Funf in a box* (Behavio, 2016). Funf est un framework de captage de données développé par le groupe *Behavio* au *MIT*. Il permet de créer des applications mobiles pour capter des données sur les téléphones intelligents. Étant un framework libre, il est possible de modifier l'application pour capter certaines données provenant des capteurs externes. Dans notre modèle, certaines données contextuelles doivent être captées à l'aide des capteurs externes telles que le données de rythme cardiaque, de pression artérielle et de température corporelle, car les téléphones n'en disposent pas. Donc, un module permettant de récupérer les données des capteurs externes doit être rajouté à l'application *HealthSense* afin de les intégrer avec les données captées sur les téléphones (localisation, connexion wifi). Ces données seront également envoyées sur nos serveurs.

b) *Les capteurs*

Les capteurs permettant de capturer la valeur de certaines informations de l'environnement de l'utilisateur. Ceci peut être la température corporelle, la position géographique, etc. Les capteurs peuvent être intégrés dans le mobile ou pas. Dans notre modèle nous proposons d'utiliser des capteurs externes pour capter les données biologiques (pression artérielle, rythme cardiaque, température corporelle, etc.). Les capteurs externes envoient les données captées dans la base de données du téléphone via un module Bluetooth. Les données brutes captées sont stockées sur le téléphone puis envoyées dans la base de données.

#### 4.16.33.20 Intégration du module crowdsourcing aux autres composants

Nous intégrons le module *crowdsensing* à la simulation *HealthSim* en interconnectant le module *monitoring* de crowdsensing à la base de données de la simulation *Healthsim (BdH)* et à la partie *source de données*. L'interconnexion de la base de données (*BdH*) avec le module *monitoring* permet de transférer les informations sur l'état de santé des individus de la population dans la base de données *patient* qui se trouve dans le module *BDS (Base de données Serveur)* qui est sur le serveur. L'intégration de la partie *source de données* au module *monitor* permet de transférer les données contextuelles captées et enregistrées sur les équipements mobiles dans la base de données contextuelle *Bdcapteur* se trouvant dans le module *BDS*.

Les informations contextuelles captées sont stockées dans un premier temps sur la base de données SQLite (*Bdm*) qui se trouve dans le portable des utilisateurs. Cela permet d'éviter les pertes de données en cas de rupture de connexion avec le serveur. Même si l'utilisateur ne dispose pas de connexion Internet pendant un certain temps, il peut continuer à exécuter sa tâche. Les données vont être stockées sur le téléphone

et envoyées au module *BDS* via le module *monitoring* responsable dans ce cas-ci de vérifier si l'utilisateur dispose d'une connexion Internet.

Dans le module *crowdsensing*, nous utilisons les composants suivants :

- a) *Management* : Le module management joue trois rôles essentiels dans le module *crowdsensing*. Il permet de fournir au plateforme *crowdsourcing* certaines informations sur l'identité des participants et l'historique des différentes tâches qu'ils ont complétés dans le passé. Il détient ces informations via le module *monitoring* qui est relié au module *BDS* contenant ces informations. Il permet également au module *crowdsourcing* de choisir les participants à qui envoyer les tâches sur leurs téléphones en utilisant la méthode *Push*.
- b) *Monitoring* : Le module *monitoring* joue le rôle de chef d'orchestre. Il permet de gérer et de coordonner les différentes interactions entre le module *source de données* : les appareils et le module *management*. Il informe le module management lorsqu'une tâche est complétée et envoie certaines informations en rapport avec les activités des utilisateurs sur leur mobile. Cela permet de déterminer si l'utilisateur utilise son téléphone et lui rappeler de prendre les mesures (pression artérielle, température corporelle, etc.) en vue de compléter sa tâche.
- c) *Module BDS* : Ce module permet de stocker les données sur le serveur. Il comprend deux bases de données, la base de données des informations captées et la base de données des patients. La *base de données capteur (BdC)* permet l'enregistrement des données contextuelles et la *base de données Patient (BdP)* permet de stocker les données sur l'état de santé des patients. Une institution de santé publique pourra ensuite y accéder en vue de les analyser.

d) *Composant de contextes* : Le composant de contexte nous permet de capter les événements contextuels suivants : température corporelle, localisation, pression artérielle, rythme cardiaque, réseau, activités.

- La température corporelle, la pression artérielle et le rythme cardiaque permettent de déterminer si l'individu présente une anomalie suivant un certain seuil capté pour ces contextes.
- La localisation nous permet à l'aide du capteur *aGPS* se trouvant sur le téléphone des patients, de déterminer dans quelle région il se trouve. Il nous permet aussi de voir si les autres individus se trouvant dans un rayon proche de lui présentent un danger de contamination en fonction des informations disposées sur le statut de santé des participants qui complètent une tâche.
- Le réseau permet de déterminer si les individus disposent d'une connexion Internet Wifi afin de téléverser les données sur le serveur de données via le *module monitoring*.
- L'activité indique si l'individu utilise son téléphone afin de lui envoyer un rappel pour prendre les mesures via les capteurs externes.

*Composants gestions des tâches* : L'entité Gestion des tâches permet de contrôler et de gérer la distribution des tâches aux participants. Il permet également de garder un historique des tâches complétées. Il existe en général deux méthodes principales permettant aux participants de recevoir des tâches. La première consiste à utiliser la méthode *push* pour envoyer les tâches sur le téléphone des participants via l'application qui est déjà installée sur leur portable. L'autre approche dite *pull* permet aux participants de choisir une tâche et de le télécharger sur leur téléphone afin de l'exécuter.

Dans notre modèle, les participants qui complètent une tâche pour la première fois doivent utiliser la deuxième approche, car ils ne disposent pas encore de l'application *HealthSense* sur leur portable et ils n'ont pas encore consenti à recevoir les tâches de cette manière là. Pour les autres fois, les tâches seront envoyées directement sur leurs téléphones mobiles car les participants auront tous à compléter des tâches identiques pour une même maladie. Cependant, un participant pourra choisir de ne plus recevoir les tâches par la méthode *push* mais plutôt les sélectionner à travers la plateforme *crowdsourcing*. Les données de géolocalisation seront captées automatiquement une fois que les termes de la tâche à compléter sont acceptés. Quant aux données de tension artérielle, de rythme cardiaque et de température corporelle, les participants auront à utiliser les capteurs externes pour capter les données. Ces données seront ensuite transférées et stockées sur la base de données du portable via une connexion *Bluetooth*.

- e) *Composantes sources de données* : contient les différentes sources permettant de capter les données. Dans notre modèle, les capteurs intégrés des mobiles ainsi que des capteurs externes permettent de capter divers types de données biologiques. L'application *HealthSense* utilise ce composant pour récupérer ces données. Une fois ces données sont enregistrées dans la base de données des portables, elle les passe ensuite au module monitoring qui est responsable de faire le contrôle et d'envoyer ces données vers le module *BDS*.

#### **4.2 Comparaison de certains systèmes épidémiologiques avec notre Système Healthsim.**

Plusieurs travaux de recherches et des systèmes ont été proposés et développés dans le but de suivre l'évolution de la propagation de certaines maladies infectieuses à caractères épidémiques. Certains de ces systèmes utilisent des approches mathématiques parfois complexes permettant de prédire l'expansion d'une épidémie

et d'autres se penchent vers l'intégration des applications mobiles et des capteurs médicaux utilisant certains modèles comme le modèle des modèles à compartiment, à base d'agents. Les systèmes utilisant les modèles mathématiques permettent de prédire l'évolution et le comportement de certaine épidémie dans une population, cependant ils ne permettent pas de suivre en temps réel la propagation de la maladie.

Dans notre système nous avons utilisé l'approche de modèle de mobilité utilisé dans les réseaux Manet ainsi que les principes d'application de type crowdsourcing et crowdsensing. L'avantage d'un tel système permettra d'un tel système permet :

- 1) *De suivre en temps réel l'évolution de la maladie au niveau de la population:*  
L'information provenant des capteurs GPS permettra aux institutions de santé publique d'intervenir dans les zones touchées dans un délai record et de prendre les mesures préventives nécessaires.
- 2) *De suivre l'état de santé de Chaque individu:* L'utilisation des capteurs pour la prise des données médicales permet aux institutions de santé d'avoir une idée précise sur l'état de santé de chaque patient. Cela favorise aussi un contrôle plus efficace et permettra aux instances concernées d'intervenir si un patient se trouve dans un état critique.
- 3) *De contrôler et de limiter la propagation d'une épidémie nouvelle :* Pour toute nouvelle épidémie , une fois le symptôme est détecté on pourra intégrer les paramètres dans notre modèle. Les patients à l'aide des capteurs appropriés pourront transmettre les informations de santé aux institutions concernés. Cela favorise largement le contrôle par les institutions de santé lors de l'apparition d'une nouvelle maladie contagieuse à caractère épidémique. Les données recueillies étant des données réels, peuvent être aussi utiliser à des fins statistiques dans le but d'analyser et de mieux comprendre l'évolution des épidémies





## CHAPITRE V

### CONCLUSION

Dans notre travail, nous avons étudié certains aspects liés à la simulation de la propagation de certaines maladies contagieuses d'une zone définie. Par la suite, nous avons proposé un modèle de type *crowdsourcing/crowdsensing* pour la surveillance épidémiologique en temps réel.

Premièrement, nous avons élaboré les problématiques liées à la simulation de la propagation et présenté les différents travaux réalisés dans ce domaine.

Pour notre simulation, nous avons utilisé certains modèles de mobilité exploités dans les réseaux ad-hoc pour représenter le déplacement des humains et pour simuler la propagation des épidémies.

L'approche *SEIR* (Sain, Exposé, Infecté, Guéri) a été prise en considération en vue de représenter les différents stades de contamination lors d'une propagation épidémiologique.

Pour manipuler et traiter les données simulées, nous avons utilisé un outil d'intégration de données (*Pentaho*) qui envoie les informations dans une base de données et qui sont par la suite rendues accessibles, à travers un service Web.

L'utilisation du service Web nous a permis d'interpréter les données simulées et d'analyser les différentes tendances pour chacun des scénarios simulés. Il nous a

également permis de comparer plusieurs scénarios simulés afin d'évaluer les fluctuations au niveau de la propagation d'une épidémie.

Nous avons ensuite présenté notre architecture de crowdsourcing et crowdsensing permettant de suivre la propagation épidémiologique en temps réel. Les différents composants dans notre architecture ainsi que les interactions entre eux ont été expliqués.

Notre but, dans ce travail, n'était pas de proposer une solution qui couvre tous les problèmes liés à la gestion de la propagation épidémiologique, car ce domaine est vaste et comporte beaucoup de défis. Nous n'avons pas pris en considération par exemple :

- *La gestion de sécurité et confidentialité des informations captées* : Cela nécessite d'implémenter des mécanismes de contrôle assez stricts, permettant d'éviter la fuite des informations médicales, ou l'accès par des personnes non autorisées à ces informations.
- *La gestion du volume de données captées*: les informations captées par les clients et envoyés dans la base de données sont généralement de grande taille. Donc, des techniques avancés comme le Data mining et des algorithmes d'analyse de données pourraient être très utiles et efficaces pour une meilleure interprétation des données captées.
- *Une couverture globale de l'épidémie* : Il est important d'avoir des informations sur les individus vivant dans les régions avoisinantes voire même d'autres continents car ces personnes peuvent aussi bien véhiculer la propagation de l'épidémie par leur déplacement.
- *La comparaison entre les phénomènes simulés et réels*: Comme nous n'avons pas eu accès à des données réelles sur les mouvements de la population

concernant les maladies infectieuses. Nous nous sommes contentés de la simulation tout en prenant soin de choisir le modèle le plus rapproché de la réalité. Cet aspect serait très intéressant à étudier dans des travaux futurs.



## ANNEXE

### *Code ns2 permettant simuler le déplacement des noeuds.*

Ce code est identique pour les modèle de mobilité ManhattanGrid, Randomwaypoint et random walk , certains paramètres au niveaux de la configuration des noeuds sont modifiés à savoir le nombre de noeuds utilisé dans la simulation, la topographie ainsi distance de contamination.

#### **ns manathan.tcl**

```
proc create_jean {id } {  
  
    global ns_ test  
  
    #set udp_($id) [new Agent/UDP]  
  
    }  
  
# procedure à appeler après la simulation  
  
proc finish {} {  
  
    #global ns_  
  
    global tracefd_ namtrace ns_  
  
    $ns_ flush-trace  
  
    close $tracefd_  
  
    close $namtrace
```

```
exit 0

    }

# quelques paramètres de ns2 permettant de configurer les noeuds
# (www.isi.edu/nsnam/ns/)

set opt(chan)      Channel/WirelessChannel  ;# channel type

set opt(prop)      Propagation/TwoRayGround ;# radio-propagation model

set opt(netif)     Phy/WirelessPhy         ;# network interface type

set opt(mac)       Mac/802_11              ;# MAC type

set opt(adhocRouting) AODV

set opt(ifq)       Queue/DropTail/PriQueue ;# interface queue type

set opt(ll)        LL                       ;# link layer type

set opt(ant)       Antenna/OmniAntenna     ;# antenna model

set opt(ifqlen)    50                       ;# max packet in ifq

set opt(nn)        100                      ;# number of mobile nodes

set opt(x)         270                      ;# x-dimension of the topography

set opt(y)         170                      ;# y-dimension of the topography

set opt(tr)        temp.rands

set opt(stop)      60.0                    ;# temps de simulation

set opt(seed)      1
```

# permet de techarger le fichier de mouvement créer par bonmotion (contient les informations sur les coordonnées des noeuds)

```
set opt(movements) /home/ubuntu/Downloads/bonnmotion-2.1.3/bin/ManhattanGrid/mana2017/scenariomanhattan2017.ns_movements
```

```
set opt(dCont) 50 ;# Distance de Contamination
```

```
global ListInfecte [list ]
```

# calcul la distance entre les noeuds et ecrire les resultats dans le fichier distance.

```
set nbr [open distance w]
```

```
puts $nbr "\t\t\t Detail of positions"
```

```
puts $nbr "\t\t\t~~~~~"
```

```
puts $nbr "\tTime\tNa\tNb\tNa-X\tNa-Y\tNb-X\tNb-Y\tDstce(d)"
```

```
puts $nbr
```

```
"\t\t\t~~~~~"
```

```
close $nbr
```

```
proc distance { n1 n2 nd1 nd2 ns_obj distCont contador InitialInfecte } {
```

```
    global c n bnd src dst j0 j1 test2
```

```
#set test1 $d
```

```
set a 0
```

```
set nbr [open distance a]
```

```

set x1 [expr int([$n1 set X_])]
set y1 [expr int([$n1 set Y_])]
set x2 [expr int([$n2 set X_])]
set y2 [expr int([$n2 set Y_])]

#calcul de distance entre les noeuds

set d [expr int(sqrt(pow(($x2-$x1),2)+pow(($y2-$y1),2)))]

set jean [expr double ($contador - 1.0000)]
set bernard [expr double ($contador + 1)]

puts $nbr "\t$contador\t$nd1\t$nd2\t$x1\t$y1\t$x2\t$y2\t$d";

if {$d <= $distCont && ( $contador == $contador || $contador == $jean ||
$contador == $bernard ) } {

set L1 [list $contador $nd1 $nd2 $d ]

puts "$L1"

set orival 0;

$ns_obj at $contador "$n1 shape square";

$ns_obj at $contador "$n1 color tan";

```



```
        $ns_obj at $contador "$n2 color brown";  
    }  
close $nbr  
}  
  
# Initialization  
set ns_ [new Simulator]  
  
$ns_ use-newtrace  
  
set now [$ns_ now]  
  
#créer un fichier de visualisation  
set namtrace [open manhattan4.nam w]  
  
$ns_ namtrace-all-wireless $namtrace $opt(x) $opt(y)  
  
set tracefd_ [open $opt(tr) w]  
  
$ns_ trace-all $tracefd_  
  
set chan_ [new $opt(chan)]
```

```
set topo_ [new Topography]

$topo_ load_flatgrid $opt(x) $opt(y)

set god_ [create-god $opt(nn)]

create-god $opt(nn)

# configuration des noeuds mobiles (www.isi.edu/nsnam/ns/)

$ns_ node-config -adhocRouting $opt(adhocRouting) \

                    -macType $opt(mac) \

                    -llType $opt(ll) \

                    -ifqType $opt(ifq) \

                    -ifqLen $opt(ifqlen) \

                    -antType $opt(ant) \

                    -propType $opt(prop) \

                    -phyType $opt(netif) \

                    -topoInstance $topo_ \

                    -agentTrace ON \

                    -routerTrace ON \

                    -macTrace ON \
```

```

-movementTrace ON \

-channel $chan_

# define color index

$ns_ color 0 red

for {set i 0} {$i < $opt(nn)} {incr i} {

set node_($i) [$ns_ node]

$node_($i) random-motion 0 ;# disable random motion

$node_($i) color "red"

    }

#créer une liste qui contient les noeuds infectés

set ListInfecte [list $node_(0) ]

#telecharge le fichier qui contient la position des noeuds

puts "Loading movement file $opt(movements)..."

source $opt(movements)

for {set i 0} {$i < $opt(nn)} {incr i} {

$ns_ initial_node_pos $node_($i) 5;

$ns_ at $opt(stop) "$node_($i) reset";

# initialise la couleur et la forme du noeud infecté

```

```

$ns_ at $now "$node_(0) color red";

$ns_ at $now "$node_(0) shape hexagon";

}

set cont 0

while {$scont <= 60} {

  for {set i 0} {$i < 1} {incr i} {

    for {set j 0} {$j < $opt(nn)} {incr j} {

      if { $i != $j } {

set a [$ns_ at $scont "distance $node_($i) $node_($j) $i $j $ns_ $opt(dCont) $scont
$ListInfecte" ];

      }

    }

  }

  set cont [ expr {$scont + 1} ]

  puts "> $scont"

}

$ns_ at $opt(stop) "finish"

$ns_ at $opt(stop) "$ns_ nam-end-wireless $opt(stop)"

$ns_ at $opt(stop) "puts \"La simulation est termin  \" ; $ns_ halt"

```

```
puts "Starting Simulation..."

puts "les noeuds qui vont etre contamines sont:\n"

puts "T Na Nb dst:"

$ns_run
```

***Code permettant d'interagir avec la base de donnée et de visualiser les informations des noeuds via le service Web. (contamination)***

L'insertion des données dans la base, les processus de manipulation ainsi que les processus de traitement se font à travers l'outil Pentaho via des paramètres comme expliqué dans le chapitre 3.

***Contamination***

***# importation des packages***

**Contamination**

#importation des packages

package arquitetura.funcional.health.base.action;

import java.util.ArrayList;

import java.util.List;

*// etc .....charge les informations sur les noeuds , états de santé, scenario, maladie, coordonnées, maladie avant d'ajouter les données aux noeuds contaminés.*

```
public void cargarNoeud(ActionMapping map, ActionForm form,
HttpServletRequest req, HttpServletResponse res, String metodo, List lst) throws
SerproException
```

```
{
    NgcNoeud ngc = new NgcNoeud();

    if (metodo != null && metodo.equalsIgnoreCase("cad")) {
        req.setAttribute("lstNod", ngc.consultarHabilitados (new String[][]
{"nodDscTitre", "ASC"}}, lst));
    } else{
        req.setAttribute("lstNod", ngc.consultarTodos(new String[][] {
{"nodDscTitre", "ASC"}}));
    }
}
```

#etc ... (public void cargarMaladie, scenario, etasante, coordonnées...)

*// ajoute les differents valeurs a la liste contamination*

```
public List<EntContamination> montarContamination(HttpServletRequest req,
ActionForm form, EntNoeud ent, String sufixo) throws Exception
```

```
{
    List<EntContamination> lst = new ArrayList<EntContamination>();
```

*// recupère les informations sur l'identité du noeud, de la maladie, du temps et de la distance.*

```
String[] conIdtChave = (String[])((DynaValidatorForm)form).get("con_" +
sufixo.toUpperCase() + "conIdtChave");
```

```
String[] conNumTemps = (String[])((DynaValidatorForm)form).get("con_"
+ sufixo.toUpperCase() + "conNumTemps");
```

```
String[] nodIdtChave1 = (String[])((DynaValidatorForm)form).get("con_"
+ sufixo.toUpperCase() + "nodIdtChave1");
```

```
String[] nodIdtChave2 = (String[])((DynaValidatorForm)form).get("con_"
+ sufixo.toUpperCase() + "nodIdtChave2");
```

```
String[] conVlrDistance = (String[])((DynaValidatorForm)form).get("con_"
+ sufixo.toUpperCase() + "conVlrDistance");
```

```
String[] mldIdtChave = (String[])((DynaValidatorForm)form).get("con_" +
sufixo.toUpperCase() + "mldIdtChave");
```

```
String[] mldIdtChaveDsc =
(String[])((DynaValidatorForm)form).get("con_" + sufixo.toUpperCase() +
"mldIdtChaveDsc");
```

*// pour chaque ligne*

```
for (int i = 0; i < conIdtChave.length; i++)
```

```
{
```

```
EntContamination entTmp = new EntContamination(); //
```

*// Copie les details sur la Contamination*

```

if (conIdtChave[i] != null && !conIdtChave[i].equals(""))

    entTmp.setConIdtChave(Long.parseLong(conIdtChave[i]));

    if (conNumTemps != null && conNumTemps.length > 0 &&
conNumTemps[i] != null && !conNumTemps[i].equals(""))

        entTmp.setConNumTemps(Integer.parseInt(conNumTemps[i]));

        if (nodIdtChave1 != null && nodIdtChave1.length > 0 && nodIdtChave1[i]
!= null && !nodIdtChave1[i].equals(""))

            {

                EntNoeud obj = new EntNoeud();

                obj.setNodIdtChave(Long.parseLong(nodIdtChave1[i]));

                entTmp.setEntNoeud1(obj);

            }

            else if(("nodIdtChave" + sufixo).replaceAll("_",
""),equalsIgnoreCase("nodIdtChave1"))

                {

                    entTmp.setEntNoeud1(ent);

                }

```



```

        if (nodIdtChave2 != null && nodIdtChave2.length > 0 && nodIdtChave2[i]
!= null && !nodIdtChave2[i].equals(""))
        {
            EntNoeud obj = new EntNoeud();

            obj.setNodIdtChave(Long.parseLong(nodIdtChave2[i]));

            entTmp.setEntNoeud2(obj);
        }

        else if(("nodIdtChave" + sufixo).replaceAll("_",
"").equalsIgnoreCase("nodIdtChave2"))
        {
            entTmp.setEntNoeud2(ent);
        }

        if (conVlrDistance != null && conVlrDistance.length > 0 &&
conVlrDistance[i] != null && !conVlrDistance[i].equals(""))

entTmp.setConVlrDistance(FormatNumber.parseBigDecimal(conVlrDistance[i]));

        if (mldIdtChave != null && mldIdtChave.length > 0 && mldIdtChave[i] !=
null && !mldIdtChave[i].equals(""))
        {
            EntMaladie entTmpMld = new EntMaladie();

```

```

        entTmpMld.setMldIdtChave(Long.parseLong(mldIdtChave[i]));

        if
        (getResources(req).getMessage("campo.contamination.mldIdtChave.visual").toLowerCase().equals(""))

            alimentarLov(req,          entTmpMld,          mldIdtChaveDsc[i],
            "campo.contamination.mldIdtChave.descricao",
            getResources(req).getMessage("caminho.entidade"));

            entTmp.setEntMaladie(entTmpMld);

        }

        if (mldIdtChave != null && mldIdtChave.length > 0 && mldIdtChave[i] !=
        null && !mldIdtChave[i].equals(""))

        {

            EntMaladie obj = new EntMaladie();

            obj.setMldIdtChave(Long.parseLong(mldIdtChave[i]));

            entTmp.setEntMaladie(obj);

        }

        lst.add(entTmp);

    }

    return lst; //retourne la liste des noeuds contaminés

}

```

```
//etc ... ( public List<EntCoordonees>,<EntEtatsanté>,<EntDeplacement> même
processus
```

```
//actualise et enregistre les informations sur la contamination
```

```
public void AtualizarContamination(ActionForm form, List<EntContamination>
lstPrs, List<EntContamination> lstTmp) throws NumberFormatException,
SerproException
```

```
{
```

```
// Enregistre les modifications sur la contamination
```

```
boolean achou = false;
```

```
for (int i = 0; i < lstPrs.size(); i++)
```

```
{
```

```
EntContamination entPrs = lstPrs.get(i);
```

```
achou = false;
```

```
for (int j = 0; (!achou && (j < lstTmp.size())); j++)
```

```
{
```

```
EntContamination entTmp = lstTmp.get(j);
```

```
if (entPrs.getConIdtChave().equals(entTmp.getConIdtChave())) //
```

```
Changement
```

```
{  
  
    entPrs.setConNumTemps(entTmp.getConNumTemps());  
  
    entPrs.setEntNoeud1(entTmp.getEntNoeud1());  
  
    entPrs.setEntNoeud2(entTmp.getEntNoeud2());  
  
    entPrs.setConVlrDistance(entTmp.getConVlrDistance());  
  
    entPrs.setEntMaladie(entTmp.getEntMaladie());  
  
    lstTmp.remove(entTmp);j--;  
  
    achou = true;  
  
    }}  
  
if (! achou)  
  
    {  
  
        lstPrs.remove(entPrs); i--;  
  
    }}  
  
    lstPrs.addAll(lstTmp); //Inclusion ds champs  
  
}
```

*//etc (actualise et enregistre à nouveau les noueds qui ont changé de statut)*

*// même processus : public void AtualizarContamination, coordonnees, déplacement, etatsanté etc...*

## BIBLIOGRAPHIE

- An, J., Gui, X., Wang, Z., Yang, J., & He, X. (2015). A Crowdsourcing Assignment Model Based on Mobile Crowd Sensing in the Internet of Things. *IEEE Internet of Things Journal*, 2(5), 358–369. <http://doi.org/10.1109/JIOT.2015.2415035>
- Bai, F., & Helmy, A. (2003). Chapter 1 A SURVEY OF MOBILITY MODELS in Wireless Adhoc Networks, University of Southern California, U.S.A, 1–30.
- Bai, F., & Helmy, A. (2004). A Survey of Mobility Models in Wireless Adhoc Networks. *Wireless Ad Hoc and Sensor Networks*, 1–30.
- Bai, F., Sadagopan, N., & Helmy, A. (2003). The IMPORTANT framework for analyzing the impact of mobility on performance of Routing protocols for Adhoc Networks. *Ad Hoc Networks*, 1(4), 383–403. [http://doi.org/10.1016/S1570-8705\(03\)00040-4](http://doi.org/10.1016/S1570-8705(03)00040-4)
- Barthélemy, M., Barrat, A., Pastor-Satorras, R., & Vespignani, A. (2005). Dynamical patterns of epidemic outbreaks in complex heterogeneous networks. *Journal of Theoretical Biology*, 235(2), 275–288. <http://doi.org/10.1016/j.jtbi.2005.01.011>
- Beaglehole, R., Bonita, R., & Kjellström, T. (2000). Basic Epidemiology 34397. <http://doi.org/16,28>
- Behavio. (2016). funf. Retrieved from <http://funf.org>
- Belik, V. V., Geisel, T., & Brockmann, D. (2009). The Impact of Human Mobility on Spatial Disease Dynamics. *2009 International Conference on Computational Science and Engineering*, 4, 932–935. <http://doi.org/10.1109/CSE.2009.432>
- Bettstetter, C. (2001). Mobility modeling in wireless networks: categorization, smooth movement, and border effects. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(3), 55–67. <http://doi.org/10.1145/584051.584056>
- Bettstetter, C., Resta, G., & Santi, P. (2003). The node distribution of the random waypoint mobility model for wireless ad hoc networks. *IEEE Transactions on Mobile Computing*, 2(3), 257–269. <http://doi.org/10.1109/TMC.2003.1233531>
- Bhatia, A. S. (2013). Analysing and Implementing the Mobility over MANETS using Random Way Point Model. *International Journal of Computer Applications*, 68(17), 32–36.

- Bhupendra Suman, S. (2011). Classification and Evaluation of Mobility Metrics for Mobility Model Movement Patterns in Mobile Ad-Hoc Networks. *International Journal on Applications of Graph Theory In Wireless Ad Hoc Networks And Sensor Networks*, 3(3), 25–38. <http://doi.org/10.5121/jgraphoc.2011.3303>
- Bisset, K. R., Feng, X., Marathe, M., & Yardi, S. (2009). Modeling interaction between individuals, social networks and public policy to support public health epidemiology, 2020–2031. Retrieved from <http://dl.acm.org/citation.cfm?id=1995456.1995734>
- Blough, D. M., Resta, G., & Santi, P. (2002). A statistical analysis of the long-run node spatial distribution in mobile ad hoc networks. *Proc. of ACM MSWiM*, 30–37.
- Boccaro, N., & Cheong, K. (1999). Critical behaviour of a probabilistic automata network SIS model for the spread of an infectious disease in a population of moving individuals. *Journal of Physics A: Mathematical and General*, 26, 3707–3717. <http://doi.org/10.1088/0305-4470/26/15/020>
- BonnMotion. (2008). BonnMotion: A mobility scenario generation and analysis tool. Retrieved from <http://sys.cs.uos.de>
- Brabham, D. C. (2008). Crowdsourcing as a Model for Problem Solving: An Introduction and Cases. *Convergence: The International Journal of Research into New Media Technologies*, 14(1), 75–90. <http://doi.org/10.1177/1354856507084420>
- Brabham, D. C., Ribisl, K. M., Kirchner, T. R., & Bernhardt, J. M. (2014). Crowdsourcing applications for public health. *American Journal of Preventive Medicine*, 46(2), 179–187. <http://doi.org/10.1016/j.amepre.2013.10.016>
- Brauer, F. (n.d.). Compartmental models in epidemiology - Wikipedia, the free encyclopedia. Retrieved from [http://en.wikipedia.org/wiki/Compartmental\\_models\\_in\\_epidemiology](http://en.wikipedia.org/wiki/Compartmental_models_in_epidemiology)
- Brauer, F. (2012). Deterministic Compartmental Disease Transmission Models. Retrieved from [http://www.math.ualberta.ca/~irl/summer\\_school/2012/lecture\\_notes/det\\_models.pdf](http://www.math.ualberta.ca/~irl/summer_school/2012/lecture_notes/det_models.pdf)
- Camp, T., Boleng, J., & Davies, V. A. (2002). A Survey of Mobility Models for Ad Hoc Network Research. *Wireless Communications & Mobile Computing (WCMC): Special Issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, 2(5), 483–502. <http://doi.org/10.1177/0897190011431143>

- Carley Fridsma, Douglas, Casma, Elizabeth, Yahja, Alex, Altman, Neal, Chen, Li-Chiou, Kaminsky, Boris, Nave, Demian, K. (2006). Bio War: Scalable Agent-Based Model of Bioattacks. *Systems and Humans*, vol 36(2), pp 252-264. Retrieved from <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1597399&tag=1>
- Chapman, J. D., & Evans, N. D. (2009). The structural identifiability of susceptible-infective-recovered type epidemic models with incomplete immunity and birth targeted vaccination. *Biomedical Signal Processing and Control*, 4(4), 278–284. <http://doi.org/10.1016/j.bspc.2009.02.003>
- Christin, D., Reinhardt, A., Kanhere, S. S., & Hollick, M. (2011). A survey on privacy in mobile participatory sensing applications. *Journal of Systems and Software*, 84(11), 1928–1946. <http://doi.org/10.1016/j.jss.2011.06.073>
- Connell, R., Dawson, P., & Skvortsov, A. (2009). Comparison of an Agent-based Model of Disease Propagation with the Generalised SIR Epidemic Model. *Technical Report Defense Science and Technology Organisation, DSTO-TR-23*, 1–22. Retrieved from <http://handle.dtic.mil/100.2/ADA510899>
- Das, T., Mohan, P., Padmanabhan, V. N., Ramjee, R., & Sharma, A. (2010). PRISM : Platform for Remote Sensing using Smartphones. *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services, MobiSys' 10*, 63–76. <http://doi.org/10.1145/1814433.1814442>
- De Cristofaro, E., & Soriente, C. (2011). Short paper: PEPSI---privacy-enhanced participatory sensing infrastructure. *Proceedings of the Fourth ACM Conference on Wireless Network Security*, 23–28. <http://doi.org/10.1145/1998412.1998418>
- Easley, D., & Kleinberg, J. (2010a). Chapter 21 Epidemics, 663–705.
- Easley, D., & Kleinberg, J. (2010b). Networks , Crowds , and Markets : Reasoning about a Highly Connected World. *Science*, 81, 744. <http://doi.org/10.1017/CBO9780511761942>
- E lance. (2016). E lance. Retrieved from <http://elance-odesk.com/online-work-report-global>
- Estellés-Arolas, E., & González-Ladrón-de-Guevara, F. (2012). Towards an integrated crowdsourcing definition. *Journal of Information Science*, 38(2), 189–200. <http://doi.org/10.1177/0165551512437638>
- Eubank, S. (2005). Network based models of infectious disease spread. *Japanese Journal of Infectious Diseases*, 58(6).

- Eubank, S., Guclu, H., Kumar, V. S., Marathe, M. V., Srinivasan, A., Toroczkai, Z., & Wang, N. (2004). Modelling disease outbreaks in realistic urban social networks. *Nature*, 429(6988), 180–184. <http://doi.org/10.1038/nature02541> [pii]
- Filipe, J. A. N., Otten, W., Gibson, G. J., & Gilligan, C. A. (2004). Inferring the dynamics of a spatial epidemic from time-series data. *Bulletin of Mathematical Biology*, 66(2), 373–391. <http://doi.org/10.1016/j.bulm.2003.09.002>
- Foundation), A. (A. S. (2016). ApacheTomcat.
- Fuchs-kittowski, F., & Faust, D. (2014). Architecture of Mobile Crowdsourcing Systems, 121–136. [http://doi.org/10.1007/978-3-319-10166-8\\_12](http://doi.org/10.1007/978-3-319-10166-8_12)
- Ganti, R. K., Ye, F., & Lei, H. (2011). Mobile crowdsensing: current state and future challenges. *Communications Magazine, IEEE*, 49(11), 32–39. Retrieved from <http://ieeexplore.ieee.org/ielx5/35/6069696/06069707.pdf?tp=&arnumber=6069707&isnumber=6069696>
- Gao, S. (2015). Spatio-Temporal Analytics for Exploring Human Mobility Patterns and Urban Dynamics in the Mobile Age. *Spatial Cognition & Computation*, 15(2), 86–114. <http://doi.org/10.1080/13875868.2014.984300>
- Gonçalves, B., Perra, N., & Vespignani, A. (2013). Social Networks, Contagion Processes and the Spreading of Infectious Diseases. *Handbook of Systems Biology*, 515–527. <http://doi.org/10.1016/B978-0-12-385944-0.00027-7>
- González, M. C., Hidalgo, C. a, & Barabási, A.-L. (2008). Understanding individual human mobility patterns. *Nature*, 453(7196), 779–782. <http://doi.org/10.1038/nature06958>
- Grassly, N. C., & Fraser, C. (2008). Mathematical models of infectious disease transmission. *Nature Reviews. Microbiology*, 6(3), 234–244. <http://doi.org/10.1038/nrmicro1845>
- Groenevelt, R., Altman, E., & Nain, P. (2006). Relaying in mobile ad hoc networks: The Brownian motion mobility model. *Wireless Networks*, 12, 561–571. <http://doi.org/10.1007/s11276-006-6535-0>
- Haas, Z. J., & Pearlman, M. R. (1998). ZRP: The performance of a new routing protocol for the reconfigurable wireless networks. *ICC '98. 1998 IEEE International Conference on Communications. Conference Record. Affiliated with SUPERCOMM'98 (Cat. No.98CH36220)*, 1(C), 562–566. <http://doi.org/10.1109/ICC.1998.682608>



- Hethcote, H. W. (2000). The Mathematics of Infectious Diseases. *SIAM Review*, 42(4), 599–653. <http://doi.org/10.1137/S0036144500371907>
- Hethcote, H. W. (2007). The Mathematics of Infectious Diseases The Mathematics of Infectious Diseases \*. *Society for Industrial and Applied Mathematics*, 42(4), 599–653. Retrieved from <http://www.jstor.org/discover/10.2307/2653135?uid=3739736&uid=2&uid=4&uid=3739256&sid=21104838342357>
- Highchart. (2016). Highchart: Interactive javascript chart for web page. Retrieved from <http://www.highcharts.com>
- Howe, J. (2006). The Rise of Crowdsourcing. *Wired Magazine*, 14(6), 1–5. <http://doi.org/10.1086/599595>
- Howe, J. (2008). Bringing Essential Knowledge & Book Summaries to High Achievers C ROWDSOURCING Why the Power of the Crowd is Driving the Future of Business. *Achievement THE International INSTITUTE*, unedited e, 320. Retrieved from <http://www.amazon.com/Crowdsourcing-Power-Driving-Future-Business/dp/0307396207>
- Hupfer, S. (2012). MoCoMapps: Mobile Collaborative Map-based Applications. *CSCW '12: Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work Companion*, 43–44. <http://doi.org/10.1145/2141512.2141534>
- Jardosh, A., Belding-Royer, E. M., Almeroth, K. C., & Suri, S. (2003). Towards Realistic Mobility Models For Mobile Ad hoc Networks. *The Ninth Annual International Conference on Mobile Computing and Networking (ACM MobiCom 2003)*, 217–229.
- Jardosh, a. P., Belding-Royer, E. M., Almeroth, K. C., & Suri, S. (2005). Real-world environment models for mobile network evaluation. *IEEE Journal on Selected Areas in Communications*, 23(3), 622–632. <http://doi.org/10.1109/JSAC.2004.842561>
- Java. (2016). Java: Software Programming. Retrieved from <https://www.oracle.com/java/index.html>
- Johnson, D. B., & Maltz, D. A. (1996). Dynamic Source Routing in Ad Hoc Wireless Networks. *Sigcomm*.
- Keeling, M. J., & Danon, L. (2009). Mathematical modelling of infectious diseases. *British Medical Bulletin*, 92(1), 33–42. <http://doi.org/10.1093/bmb/ldp038>

- Keeling, M. J., & Eames, K. T. D. (2005). Networks and epidemic models. *Journal of the Royal Society, Interface / the Royal Society*, 2(4), 295–307. <http://doi.org/10.1098/rsif.2005.0051>
- Kermack, W. O., & McKendrick, A. G. (1927). A Contribution to the Mathematical Theory of Epidemics. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*. <http://doi.org/10.1098/rspa.1927.0118>
- Kim, J. (2005). REALISTIC MOBILITY MODELING AND SIMULATION FOR MOBILE WIRELESS NETWORK IN URBAN ENVIRONMENTS by REALISTIC MOBILITY MODELING AND SIMULATION FOR MOBILE WIRELESS NETWORK IN URBAN ENVIRONMENT.
- Kleczkowski, A., & Grenfell, B. T. (1999). Mean-field-type equations for spread of epidemics: The 'small world' model. *Physica A: Statistical Mechanics and Its Applications*, 274(1), 355–360. [http://doi.org/10.1016/S0378-4371\(99\)00393-3](http://doi.org/10.1016/S0378-4371(99)00393-3)
- Korthaus, A., & Dai, W. (2012). Crowdsourcing in heterogeneous networked environments - Opportunities and challenges. *Proceedings of the 2012 15th International Conference on Network-Based Information Systems, NBIS 2012*, 483–488. <http://doi.org/10.1109/NBiS.2012.130>
- Lee, E. K., Chen, C. H., Pietz, F., & Benecke, B. (2010). Disease propagation analysis and mitigation strategies for effective mass dispensing. *AMIA ... Annual Symposium Proceedings / AMIA Symposium. AMIA Symposium, 2010*, 427–431.
- Liao, J., Hu, X., Leeson, M. S., Hines, E. L., Hu, X., Wang, M., & Paolo, E. Di. (2012). A Ripple-Spreading Network Model For The Study Of Infectious Disease Transmission, (Bmei), 1004–1010.
- Madan, A., Cebrian, M., Moturu, S., Farrahi, K., & Pentland, S. (2012). Sensing the "Health State" of a Community. *Pervasive Computing*, 11(4), 36–45. <http://doi.org/10.1109/MPRV.2011.79>
- Margolus, N., & Toffoli, T. (1987). Cellular Automata Machines. *Complex Systems*, 1, 967–993.
- Mavandadi, S., Dimitrov, S., Feng, S., Yu, F., Sikora, U., Yaglidere, O., ... Ozcan, A. (2012). Distributed medical image analysis and diagnosis through crowd-sourced games: A malaria case study. *PLoS ONE*, 7(5), 1–8. <http://doi.org/10.1371/journal.pone.0037245>
- Meyers, L. A. (2007). Contact network epidemiology: Bond percolation applied to infectious disease prediction and control. *Bulletin of the American Mathematical*

*Society*, 44(1), 63–86. <http://doi.org/10.1090/S0273-0979-06-01148-7>

Mikler, A. R., Bravo-Salgado, A., & Corley, C. D. (2009). Global stochastic contact modeling of infectious diseases. *Proceedings - 2009 International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing, IJCBS 2009*, 327–330. <http://doi.org/10.1109/IJCBS.2009.84>

Minkyong, K., Kotz, D., & Songkuk, K. (2006). Extracting a mobility model from real user traces. *Proceedings - IEEE INFOCOM, 0(c)*. <http://doi.org/10.1109/INFOCOM.2006.173>

Munjal, A., Camp, T., & Navidi, W. C. (2011). SMOOTH: a simple way to model human mobility, 1–9.

MySQL. (2016). MySQL: Database. Retrieved from <https://www.mysql.com>

N. Aschenbruck, M. Frank, P. Martini, J. T. (2004). Human mobility in MANET disaster area simulation—a realistic approach. *29th Annual IEEE International Conference on Local Computer Networks (LCN '04)*, 668–675.

Nam. (2002). Nam: Network Animator. Retrieved from <http://www.isi.edu/nsnam/nam/>

Nistal, R., De La Sen, M., Alonso-Quesada, S., & Ibeas, A. (2013). Periodic equilibrium states in a SEIR mathematical model of an infectious non-lethal disease. In *2013 25th Chinese Control and Decision Conference, CCDC 2013* (pp. 2155–2160). <http://doi.org/10.1109/CCDC.2013.6561292>

Ns2. (2005). Ns2 discret event network-simulator. Retrieved from <http://www.isi.edu/nsnam/ns/>

O'Neill, P. D. (2002). A tutorial introduction to Bayesian inference for stochastic epidemic models using Markov chain Monte Carlo methods. *Mathematical Biosciences*, 180, 103–114. [http://doi.org/10.1016/S0025-5564\(02\)00109-8](http://doi.org/10.1016/S0025-5564(02)00109-8)

Odesk. (2016). Odesk. Retrieved from <https://www.upwork.com/about/>

Orival, J.-B. (2017). HealthSim. Retrieved from [https://github.com/vavaldi/Health\\_sim](https://github.com/vavaldi/Health_sim)

pentaho. (2005). Pentaho: Data Integration tools. Retrieved from <http://www.pentaho.com/product/data-integration>

Perez, L., & Dragicevic, S. (2009). An agent-based approach for modeling dynamics

- of contagious disease spread. *International Journal of Health Geographics*, 8, 50. <http://doi.org/10.1186/1476-072X-8-50>
- Ra, M., Liu, B., La Porta, T. F., & Govindan, R. (2012). Medusa: a programming framework for crowd-sensing applications. *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services - MobiSys '12*, (Section 2), 337. <http://doi.org/10.1145/2307636.2307668>
- Ranard, B. L., Ha, Y. P., Meisel, Z. F., Asch, D. A., Hill, S. S., Becker, L. B., ... Merchant, R. M. (2014). Crowdsourcing--harnessing the masses to advance health and medicine, a systematic review. *Journal of General Internal Medicine*, 29(1), 187–203. <http://doi.org/10.1007/s11606-013-2536-8>
- Reddy, S., Estrin, D., & Srivastava, M. (2010). Recruitment framework for participatory sensing data collections. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6030 LNCS, 138–155. [http://doi.org/10.1007/978-3-642-12654-3\\_9](http://doi.org/10.1007/978-3-642-12654-3_9)
- Roy, R. R. (2010). *Handbook of Mobile Ad Hoc Networks for Mobility Models*. <http://doi.org/10.1017/CBO9781107415324.004>
- Royer, E. M. (2001). An Analysis of the Optimum Node Density for Ad hoc Mobile Networks. *IEEE Int. Conf. Commun.*, 857–861.
- Sontag, E. D., & Sontag, E. D. (2006). *Lecture Notes in Mathematical Biology*.
- Stattner, E. (2013). Docteur en Sciences Erick Stattner sociaux : propagation , fouille , collecte de Thèse dirigée par Martine Collard.
- Stattner, E., & Vidot, N. (2011). Social network analysis in epidemiology: Current trends and perspectives. *2011 Fifth International Conference on Research Challenges in Information Science*, Stattner, 1–11. <http://doi.org/10.1109/RCIS.2011.6006866>
- Sun, J., Ibrahim, A. N., Gong, J., Yang, L., Li, Y., & Zhou, J. (2012). Study on spread of vibrio cholera in rivers based on Cellular Automata Model. *International Geoscience and Remote Sensing Symposium (IGARSS)*, 491–494. <http://doi.org/10.1109/IGARSS.2012.6351379>
- Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of “small-world” networks. *Nature*, 393(6684), 440–2. <http://doi.org/10.1038/30918>
- Welbourne, E., Wu, P., Bao, X., & Munguia-Tapia, E. (2014). Crowdsourced Mobile

- Data Collection: Lessons Learned from a New Study Methodology. *HotMobile '14*, 0–5. Retrieved from <http://dl.acm.org/citation.cfm?id=2565608>
- World Health Organization, N. (2008). The Global Burden of Disease: 2004 update. *Update, 2010*, 146. <http://doi.org/10.1038/npp.2011.85>
- Yuen, M. C., King, I., & Leung, K. S. (2011). A survey of crowdsourcing systems. *Proceedings - 2011 IEEE International Conference on Privacy, Security, Risk and Trust and IEEE International Conference on Social Computing, PASSAT/SocialCom 2011*, 766–773. <http://doi.org/10.1109/PASSAT/SocialCom.2011.36>
- Zhang, Y., Low, C. P., & Ng, J. M. (2011). Performance Evaluation of Routing Protocols on the Reference Region Group Mobility Model for MANET. *Online, 2011*(March), 92–105. <http://doi.org/10.4236/wsn.2011.33010>
- Zheng, Q., Hong, X., & Ray, S. (2004). Recent Advances in Mobility Modeling for Mobile Ad Hoc Network Research. *Analysis*.