POLYPUBLIE

Amélioration des performances de localisation et de suivi d'un mobile dans un environnement interne en utilisant le filtrage de Kalman et de ses extensions

Salim Outemzabet

Document en libre accès dans PolyPublie
Open Access document in PolyPublie

Date	2007
Type	Mémoire ou thèse / Dissertation or Thesis Mémoire de maîtrise/Master's thesis
URL de PolyPublie **PolyPublie URL**	https://publications.polymtl.ca/8018/
Conditions d'utilisation **Terms of use**	© Tous droits réservés / All rights reserved
Référence **Citation**	Outemzabet, S. (2007). Amélioration des performances de localisation et de suivi d'un mobile dans un environnement interne en utilisant le filtrage de Kalman et de ses extensions [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie. https://publications.polymtl.ca/8018/

Détails de la thèse
Thesis details

Directeurs de recherche **Advisors**	Chahe Nerguizian
Programme **Program**	Non spécifié

Ce fichier a été téléchargé à partir de PolyPublie, le dépôt institutionnel de Polytechnique Montréal
This file has been downloaded from PolyPublie, the institutional repository of Polytechnique Montréal
https://publications.polymtl.ca

UNIVERSITÉ DE MONTRÉAL

AMÉLIORATION DES PERFORMANCES DE LOCALISATION ET DE SUIVI D'UN MOBILE DANS UN ENVIRONNEMENT INTERNE EN UTILISANT LE FILTRAGE DE KALMAN ET DE SES EXTENSIONS

SALIM OUTEMZABET
DÉPARTEMENT DE GÉNIE ÉLECTRIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE ÉLECTRIQUE)
SEPTEMBRE 2007

© Salim Outemzabet, 2007.

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-35693-7
Our file Notre référence
ISBN: 978-0-494-35693-7

NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:
L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

AMÉLIORATION DES PERFORMANCES DE LOCALISATION ET DE SUIVI D'UN MOBILE DANS UN ENVIRONNEMENT INTERNE EN UTILISANT LE FILTRAGE DE KALMAN ET DE SES EXTENSIONS

présenté par: OUTEMZABET Salim

en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de:

M. CARDINAL Christian, Ph.D., président

M. NERGUIZIAN Chahé, Ph.D., membre et directeur de recherche

M. FRIGON Jean-François, Ph.D., membre

Je dédie ce mémoire :

à mes chers parents, ceux qui se sont privés du plus nécessaire pour m'instruire,

à mon frère Lyes,

à ma sœur, à son mari et à leur petite Chayema,

à toute la famille OUTEMZABET et à tous mes proches,

et surtout...

à ma future épouse, celle qui me porte tendrement dans son cœur.

Voici 5 règles simples pour être heureux :

Libérez votre cœur de la haine,

Libérez votre esprit des inquiétudes,

Vivez simplement,

Donnez plus,

Attendez moins.

REMERCIEMENTS

Tout d'abord, je remercie le tout haut, Allah ou Dieu, pour m'avoir permis d'aller au terme de ce travail.

Au terme de ce présent travail de recherche, je tiens à remercier vivement M. Chahé Nerguizian pour son soutien permanent, son encadrement dynamique, ainsi que ses discussions riches d'idées innovantes et ses remarques judicieuses qui ont contribuées à l'achèvement de ce mémoire. Sa rigueur ainsi que son esprit critique m'ont toujours permis d'avancer dans la bonne direction. Travailler sous sa responsabilité m'a énormément apporté sur le plan scientifique.

Je tient à exprimer ma gratitude à messieurs les membres de jury, Christian Cardinal et Jean-François Frigon, qui ont accepté de juger mon travail de recherche.

J'exprime toute ma reconnaissance au gouvernement d'Algérie, pour m'avoir octroyé une bourse d'études supérieures.

Mes sincères remerciements vont à l'ensemble de mes collègues d'études, qui ont contribués à l'élaboration de ce travail, et avec lesquels j'ai trouvé toute la sympathie et le climat amical. Je pense tout particulièrement à Ali, Amine, Boukari, Florian, Khalil, Lamia, Olfa, Rachid, Stéphane et Zhao.

Je remercie également tous mes amis qui m'ont soutenu ici au canada ou ailleurs, en pensant entre autres à Ahriche, Bahi, Chaoui, Doufana, Larbi, Lyzidi, Sid-Ali et Touhami.

Enfin, je remercie tous ceux qui ont contribués à la réussite de ce mémoire, et que tous ceux que j'ai involontairement oubliés trouvent ici l'expression de ma gratitude.

RÉSUMÉ

Ce travail de recherche s'inscrit dans le contexte d'amélioration des performances des systèmes de localisation et de suivi d'un mobile dans un environnement interne. L'approche d'amélioration proposée est basée sur l'application du filtrage de Kalman et de ses extensions à un système de localisation de fingerprinting basé sur les signaux de puissances reçus au mobile à partir d'un réseau sans fil WLAN. L'application du filtrage de Kalman et de ses extensions a l'avantage d'exploiter toutes les informations relatives à la dynamique du mobile pour réduire les erreurs de localisation et pour ne pas présenter des discontinuités au niveau de la trajectoire de suivi du mobile. Dans ce travail de recherche, trois variantes d'amélioration, relatives au modèle de filtrage caractérisant la dynamique d'un piéton, ont été proposées. La première variante est basée sur un modèle de filtrage linéaire (modélisant le mouvement linéaire en 2D d'un piéton). La deuxième variante est basée sur un modèle de filtrage non-linéaire (obtenu après avoir introduit dans le modèle de filtrage linéaire l'information sur l'angle d'orientation du mobile mesurée par une boussole). Finalement, la troisième variante est basée sur un modèle de filtrage linéarisé (obtenu en remplaçant dans le modèle non-linéaire la variable de l'angle d'orientation par deux variables de vitesses axiales du mobile). Par rapport à chaque modèle de filtrage considéré, les filtres de Kalman, de Kalman étendu et de particules ont été appliqués à la technique de fingerprinting à base du réseau de neurones de type GRNN (General Regression Neural Network). Les résultats expérimentaux obtenus ont permis de valider l'approche d'amélioration proposée, où la précision de localisation, en termes d'erreur moyenne, a été améliorée jusqu'à environ 50%, et la trajectoire de suivi du mobile était bien orientée et plus proche de celle réelle.

ABSTRACT

This research work falls in the context of enhancement of mobiles location tracking systems in indoor environments. The proposed enhancement approach is based on Kalman filtering and its extensions, applied to a fingerprinting location system, based on the received signal strength (RSS) from a wireless local area network (WLAN). The application of Kalman filtering and its extensions has the advantage of using all information about the motion of the mobile to reduce location errors and to avoid trajectory discontinuities. In this research, three variantes of the enhancement approach, related to the filtering model that characterizes the state and the observation of the dynamic of a pedestrian, have been proposed. The first variante is based on a linear filtering model, characterizing the linear 2D motion of a pedestrian. The second is based on a non-linear filtering model, obtained by adding to the linear model the information about the mobile's heading measured by a compass. Finally, the third variante is based on a linearized filtering model, obtained by replacing in the non-linear model the heading variable by the two variables of the axial velocities of the mobile. In each considered variante, Kalman, extended Kalman and particle filters were applied to the GRNN network. Experimental results have permitted to validate the proposed enhancement approach, where the location accuracy, in terms of mean error, has been enhanced about 50%, and the mobile tracking trajectory was well oriented and closer to the real one.

TABLE DES MATIÈRES

DÉDICACE iv

REMERCIEMENTS v

RÉSUMÉ vi

ABSTRACT vii

TABLE DES MATIÈRES viii

LISTE DES FIGURES xii

ACRONYMES ET ABRÉVIATIONS xv

LISTE DES TABLEAUX xvii

LISTE DES ANNEXES xix

CHAPITRE 1 INTRODUCTION GÉNÉRALE 1

1.1 Motivations 1

1.2 Contributions du mémoire 3

1.3 Organisation du mémoire 5

CHAPITRE 2 ÉTAT DE L'ART DES SYSTÈMES DE LOCALISATION EN ENVIRONNEMENTS INTERNES 6

2.1 Introduction 6

2.2 Principe de fonctionnement des systèmes de localisation en environnements internes 7

2.3 Classification des systèmes de localisation en environnements internes 9

2.3.1 Classification suivant la technologie de détection 9

2.3.1.1 Technologie de détection par infrarouge 10
2.3.1.2 Technologie de détection ultrason 10
2.3.1.3 Technologie de détection radiofréquence 11
2.3.2 Classification suivant la technique de localisation 11
2.3.2.1 Technique de localisation par latération 12
2.3.2.2 Technique de localisation par angulation 15
2.3.2.3 Technique de localisation de fingerprinting 16
2.3.3 Classification suivant les propriétés 18
2.4 Exemples de systèmes de localisation en environnements internes . 19
2.4.1 Le système de localisation Active Badge 19
2.4.2 Le système de localisation Active Bat 20
2.4.3 Le système de localisation Cricket 21
2.4.4 Système de localisation RADAR 21
2.4.5 Système de localisation RFID LANDMARC 22
2.5 Conclusion 23

CHAPITRE 3 MODÉLISATION ET EXPÉRIMENTATION D'UN SYSTÈME DE LOCALISATION DANS UN ENVIRONNEMENT INTERNE 25
3.1 Introduction 25
3.2 Description fonctionnelle du système de localisation proposé 26
3.2.1 Technologie de détection (WLAN) 27
3.2.1.1 Standard IEEE 802.11 WLAN 27
3.2.1.2 Caractéristiques d'un environnement interne WLAN 29
3.2.2 Technique de localisation (fingerprinting) 30
3.2.2.1 Algorithme des K-plus proches voisins 32
3.2.2.2 Réseaux de neurones artificiels 33
3.3 Description expérimentale du système de localisation proposé . . . 39

3.3.1 Environnement interne d'expérimentation 40
3.3.2 Matériel utilisé 42
3.3.2.1 Réseaux WLAN 42
3.3.2.2 Usager mobile 42
3.3.2.3 L'outil NetStumbler 43
3.3.2.4 Outil d'implémentation 44
3.3.3 Procédure de collecte des mesures 45
3.3.4 Implémentation des algorithmes de localisation 46
3.4 Résultats et analyse des performances de localisation 47
3.4.1 Fluctuations du signal de puissance RSS 48
3.4.2 Performances de localisation 49
3.4.3 Effet du nombre de points d'accès sur la précision de localisation 51
3.5 Conclusion 52

CHAPITRE 4 AMÉLIORATION DES PERFORMANCES DU SYSTÈME DE LOCALISATION DE FINGERPRINTING EN UTILISANT LE FILTRAGE DE KALMAN ET DE SES EXTENSIONS 53
4.1 Introduction 53
4.2 Filtrage de Kalman et de ses extensions 54
4.2.1 Filtrage de Kalman 54
4.2.1.1 Description du filtrage de Kalman 55
4.2.1.2 Étapes d'estimation du filtre de Kalman 57
4.2.2 Filtrage de Kalman étendu 59
4.2.3 Filtrage de particules 62
4.2.3.1 Description du filtrage de particules 62
4.2.3.2 Étapes d'estimation du filtre de particules 66

4.3 Application du filtrage de Kalman et de ses extensions au cas de la localisation et du suivi 67
4.3.1 Modèle de filtrage linéaire 69
4.3.2 Modèle de filtrage non-linéaire 71
4.3.3 Modèle de filtrage linéarisé 75
4.4 Présentation et analyse des résultats de localisation et de suivi obtenus dans chacun des trois modèles de filtrage proposés 77
4.4.1 Dimensionnement des paramètres de filtrage 77
4.4.2 Résultats et analyse des performances de localisation et de suivi dans le cas du modèle de filtrage linéaire 80
4.4.3 Résultats et analyse des performances de localisation et de suivi dans le cas du modèle de filtrage non-linéaire 83
4.4.4 Résultats et analyse des performances de localisation et de suivi dans le cas du modèle de filtrage linéarisé 86
4.4.5 Analyse des performances de localisation et de suivi selon les paramètres de filtrage 93
4.4.5.1 Influence des paramètres du bruit de mesure sur les performances de localisation et de suivi 93
4.4.5.2 Temps de traitement d'une estimation de localisation 94
4.4.5.3 Effet du nombre de particules sur les performances de localisation et de suivi 95
4.5 Conclusion 96

CHAPITRE 5 CONCLUSION GÉNÉRALE 100
5.1 Améliorations envisageables et travaux futurs 102

BIBLIOGRAPHIE 104

ANNEXES 109

LISTE DES FIGURES

FIG. 2.1	Diagramme fonctionnel d'un système de localisation dans un environnement interne.	8
FIG. 2.2	Techniques de localisation classiques.	12
FIG. 2.3	Approche de localisation par TOA.	13
FIG. 2.4	Approche de localisation par TDOA.	14
FIG. 2.5	Approche de localisation par AOA.	16
FIG. 2.6	Système de localisation RFID LANDMARC.	23
FIG. 3.1	Modèle fonctionnel du système de localisation proposé.	26
FIG. 3.2	Principe de localisation par la technique de fingerprinting.	31
FIG. 3.3	Structure de base d'un neurone artificiel.	34
FIG. 3.4	Architecture multicouche d'un réseau de neurones de type MLP.	35
FIG. 3.5	Architecture du réseau GRNN.	38
FIG. 3.6	Processus expérimental du système de localisation proposé.	40
FIG. 3.7	Plan du $5^{ème}$ étage du pavillon Mckay-Lassonde, situé à l'École Polytechnique de Montréal.	41
FIG. 3.8	Photos de deux couloirs du $5^{ème}$ étage du pavillon Mckay-Lassonde, situé à l'École Polytechnique de Montréal	41
FIG. 3.9	Fenêtre de détection par l'outil NetStumbler, des six points d'accès considérés dans l'expérimentation.	44
FIG. 3.10	Fluctuations du niveau du signal RSS au cours du temps pour une même position de localisation, pour les six points d'accès considérés.	48
FIG. 3.11	Courbes des distributions cumulées de l'erreur de localisation pour les algorithmes KNN, MLP et GRNN.	49

FIG. 3.12 Courbes des erreurs de localisation instantanées pour les algorithmes KNN, MLP et GRNN. 50
FIG. 4.1 Principe de fonctionnement du filtrage de Kalman. 55
FIG. 4.2 Étapes d'estimation récursives du filtre de Kalman. 58
FIG. 4.3 Étapes d'estimation récursives du filtre de Kalman étendu. . 61
FIG. 4.4 Modèles de filtrage proposés pour l'amélioration de la position de localisation du mobile. 68
FIG. 4.5 Photo et fenêtre d'affichage de la boussole F350-COMPASS-RD. 72
FIG. 4.6 Courbes des distributions cumulatives (CDF) de l'erreur de localisation avant et après l'application du filtre de Kalman au réseau GRNN, dans le cas du modèle de filtrage linéaire. 80
FIG. 4.7 Courbes de l'erreur de localisation instantanée avant et après l'application du filtre de Kalman au réseau GRNN, dans le cas du modèle de filtrage linéaire. 81
FIG. 4.8 Trajectoires de suivi du mobile obtenues par les données réelles, avant et après l'application du filtre de Kalman au réseau GRNN, dans le cas du modèle de filtrage linéaire. . . 82
FIG. 4.9 Courbes des distributions cumulatives (CDF) de l'erreur de localisation avant et après l'application des filtres EKF et PF au réseau GRNN, dans le cas du modèle de filtrage non-linéaire. 83
FIG. 4.10 Comparaison des trajectoires de suivi du mobile obtenues avant et après l'application au réseau GRNN, des filtres EKF, PF (modèle de filtrage non-linéaire) et KF (modèle de filtrage linéaire). 86
FIG. 4.11 Courbes des distributions cumulatives de l'erreur de localisation avant et après l'application des filtres KF et PF au réseau GRNN, dans le cas du modèle de filtrage linéarisé. . 87

FIG. 4.12 Courbes des distributions cumulatives de l'erreur de localisation obtenues avant et après l'application des différents filtres au réseau GRNN dans le cas de chacun des trois modèles de filtrage proposés. 89

FIG. 4.13 Comparaison des trajectoires de suivi du mobile obtenues par les filtres KF et PF, appliqués au réseau GRNN dans le cas du modèle de filtrage linéarisé, à celles obtenues par les filtres appliqués au réseau GRNN dans le cas des modèles de filtrage linéaire et non-linéaire. 91

FIG. 4.14 Courbes des vitesses axiales du piéton (suivant x et y) obtenues par filtrage de Kalman appliqué au GRNN dans le cas des modèles linéaire (pas de boussole) et linéarisé (utilisation de la bousole). 92

FIG. 4.15 Influence des paramètres du bruit de mesure sur la précision de localisation du mobile, dans le cas du filtre de Kalman appliqué au réseau GRNN dans le modèle de filtrage linéaire. 94

ACRONYMES ET ABRÉVIATIONS

AGNSS	Assisted Global Navigation Satellite System
ANN	Artificial Neural Network
AOA	Angle Of Arrival
AP	Access Point
CDF	Cumulative Distribution Function
CCK	Complementary Code Keying
CIR	Channel Impulse Response
EKF	Extended Kalman Filter
FPGA	Field Programmable Gate Array
GPS	Global Positioning System
GRNN	Generalized Regression Neural Network
IEEE	Institute of Electrical and Electronics Engineers
IEC	International Electrotechnical Commission
IR	Infrared
ISO	International Organization for Standardization
KF	Kalman Filter
KNN	K-Nearest Neighbors
LLC	Logical Link Control
MAC	Medium Access Control
MLP	Multi-Layer Perceptron
MSE	Mean Square error
NLOS	Non Line-Of-Sight
OFDM	Orthogonal Frequency Division Multiplexing
PC	Personal Computer
PDA	Personal Digital Assistant
PF	Particle Filter

RBF	Radial Basis Function
RF	Radio Frequency
RFID	Radio Frequency IDentification
RSS	Received Signal Strength
SIS	Sequential Importance Sampling
SIR	Sampling Importance Resampling
SNR	Signal to Noise Ratio
SSID	Service Set IDentifier
SVM	Support Vector Machine
TDOA	Time Difference Of Arrival
TOA	Time Of Arrival
USB	Universal Serial Bus
UWB	Ultra-Wide Band
WECA	Wireless Ethernet Compatibility Alliance
Wi-Fi	Wireless-Fidelity
WLAN	Wireless Local Area Network

LISTE DES TABLEAUX

TAB. 3.1 Caractéristiques techniques du réseau sans fil HERMES. . . 42

TAB. 3.2 Paramètres et architectures des algorithmes KNN, MLP et GRNN. 47

TAB. 3.3 Comparaison des performances de localisation entre les algorithmes KNN, MLP et GRNN et la méthode RADAR [1]. . 51

TAB. 3.4 Effet du nombre de points d'accès sur les performances de localisation 51

TAB. 4.1 Dimensionnement des paramètres des trois modèles de filtrage proposés. 79

TAB. 4.2 Performances de localisation avant et après l'application du filtre de Kalman au réseau GRNN, dans le cas du modèle de filtrage linéaire. 82

TAB. 4.3 Performances de localisation avant et après l'application des filtres EKF et PF au GRNN, dans le cas du modèle de filtrage non-linéaire. 84

TAB. 4.4 Comparaison des performances de localisation obtenues dans le cas du modèle de filtrage linéaire à celles obtenues dans le cas du modèle de filtrage non-linéaire. 85

TAB. 4.5 Performances de localisation et de suivi avant et après l'application des filtres KF et PF au réseau GRNN, dans le cas du modèle de filtrage linéarisé. 88

TAB. 4.6 Comparaison des performances de localisation entre les différents filtres appliqués au réseau GRNN dans le cas des trois modèles de filtrage proposés. 90

TAB. 4.7 Temps de traitement d'une estimation de localisation avant et après l'application au réseau GRNN, de chacun des filtres utilisés dans le cas des modèles de filtrage linéaire, non-linéaire et linéarisé. 95

TAB. 4.8 Effet du nombre de particules sur les performances de localisation et de suivi dans le cas du filtre de particules appliqué au réseau GRNN dans le cas du modèle de filtrage linéarisé. 96

LISTE DES ANNEXES

ANNEXE I PROGRAMMES MATLAB 109
I.1 Programme de construction de la base de données des signatures . 109
I.2 Programme de l'algorithme KNN adapté au cas de la localisation . 110
I.3 Programme d'entraînement du réseau MLP adapté au cas de la localisation 112
I.4 Programme d'entraînement du réseau GRNN adapté au cas de la localisation 113
I.5 Programme du filtre de Kalman adapté au cas de la localisation et du suivi 113
I.6 Programme du filtre de Kalman étendu adapté au cas de la localisation et du suivi 115
I.7 Programme du filtre de particules adapté au cas de la localisation et du suivi 116
I.8 Programme fournissant les paramètres de localisation et de filtrage dans le cas des trois modèles de filtrage considérés 118
I.9 Programme permettant d'implémenter le processus de localisation et de suivi 122

ANNEXE II FOIRE AUX QUESTIONS 126

CHAPITRE 1

INTRODUCTION GÉNÉRALE

1.1 Motivations

La localisation ou le positionnement des personnes, d'équipements ou des terminaux mobiles, constitue actuellement un domaine de recherche en plein essor, où plusieurs services ou applications nécessitent des informations de localisation. Parmi ces services on cite par exemples :

- La localisation des soldats, des policiers et des pompiers pour l'accomplissement de leurs missions.
- La localisation dans les hôpitaux des médecins ou des patients et le contrôle des équipements à coûts élevés.
- La localisation et le suivi des robots.
- La visite guidée dans un musée ou dans un campus universitaire.
- La localisation des téléphones sans fil appelant un numéro d'urgence.
- La mise à disposition d'un plan d'étage ou d'indications descriptives de la localisation actuelle d'un utilisateur.

Ainsi, la communauté scientifique et industrielle semble considérer qu'il est tout à fait important de fournir des systèmes de localisation avec de bonnes performances et de développer des techniques pour y parvenir. Si dans un environnement externe la localisation est triviale avec l'utilisation des systèmes de navigation par satellites, comme par exemple le GPS (Global Positioning System) ou encore l'AGNSS (Assisted Global Navigation Satellite System), qui permettent d'atteindre des précisions de l'ordre de quelques mètres, dans un environnement interne (intérieur des bâti-

ments, souterrains, zones urbaines à grande densité, etc.), la localisation constitue une forte limitation de ces systèmes, à cause de la dégradation du signal radio-fréquence dans ces milieux, due au phénomène de multitrajet et à la non-visibilité directe entre l'émetteur et le récepteur (NLOS : Non-Line Of Sight).

Récemment, la majorité des immeubles (hôpitaux, universités, aéroports, centres commerciaux, etc.) ont été équipés par les réseaux sans fils WLAN, ce qui a ouvert la possibilité à développer des systèmes de localisation dédiés aux environnements internes. Le principe de fonctionnement de ces systèmes de localisation est basé en général, sur la technique de fingerprinting [1], qui exploite la dépendance entre les signaux de puissances RSS, provenant des différents points d'accès du réseau WLAN déployé, et les positions de localisation où ses signaux ont été reçus. La précision de localisation de cette technique dépend essentiellement du nombre de points d'accès déployés, du nombre de mesures RSS enregistrées dans la base de données et de l'algorithme d'estimation (ou algorithme de jumelage) utilisé .

Cependant, et même avec cette alternative, les performances de localisation se sont avérées pas assez satisfaisantes à cause des fluctuations du signal de puissance RSS au cours du temps, qui introduisent des erreurs de localisation et des discontinuités dans la trajectoire du mobile. Pour minimiser ces erreurs et avoir une bonne trajectoire, du filtrage doit être appliqué. Le filtrage de Kalman, appliqué jusqu'à la dernière décennie dans le domaine de l'automatique et de la robotique, semble être le mieux adapté pour résoudre le problème posé ci-dessus. Son avantage est qu'il exploite toutes les informations sur la dynamique ou le mouvement du mobile, pour filtrer les différents bruits relatifs aux erreurs de modélisation et de mesure. De ce fait, notre principale motivation derrière cette recherche consiste en cet avantage que présente le filtrage de Kalman et de ses extensions à améliorer une estimation donnée, tout en exploitant le fait que les réseaux WLANs sont déployés dans divers coins, ce qui les rend accessibles à des fins de localisation.

Par conséquent, le but de ce présent travail est d'appliquer le filtrage de Kalman et de ses extensions au système de localisation basé sur la technique de fingerprinting, pour améliorer les performances de localisation et de suivi d'un mobile dans un environnent interne équipé par un réseau WLAN. Pour ce faire, trois phases opératoires sont considérées. En premier lieu, une base de données (contenant les mesures des signatures RSS prises en différentes positions dans l'environnement d'application) est construite. Puis, cette base de données est exploitée par la technique de fingerprinting pour estimer la position de localisation du mobile, en utilisant les algorithmes de jumelage des réseaux de neurones MLP (multi-Layer Perceptron) et GRNN, et de la méthode KNN (K-Nearest Neighbors). Finalement, cette estimation est améliorée en utilisant le filtrage de Kalman et de ses extensions. Dans cette dernière phase, trois variantes d'amélioration, relatives au modèle de filtrage caractérisant la dynamique du mobile, sont proposées. La première variante est basée sur un modèle de filtrage linéaire, qui caractérise le mouvement linéaire en 2D d'un piéton, avec comme observation la position en 2D estimée par le réseau GRNN. La deuxième variante est basée sur un modèle de filtrage non-linéaire, qui est obtenu après avoir introduit dans le modèle linéaire une nouvelle information sur la dynamique du mobile, qui consiste en son angle d'orientation mesuré par une boussole. Et la troisième variante est basée sur un modèle de filtrage linéarisé, qui consiste à remplacer dans le modèle de filtrage non-linéaire, la variable non-linéaire de l'angle d'orientation par les deux variables linéaires des vitesses axiales du mobile.

1.2 Contributions du mémoire

Dans le but d'améliorer les performances de localisation et de suivi d'un mobile dans un environnement interne, notre contribution dans ce travail de recherche consiste en ce qui suit :

- Conception judicieuse et validation expérimentale d'un système de localisation dans un environnement interne, basé sur la technologie sans fil WLAN et la technique de fingerprinting, où les algorithmes de jumelage : MLP, GRNN et KNN, ont été examinés et comparés à l'approche classique RADAR [1].

- Étude théorique et dimensionnement paramétrique des filtres de Kalman (KF), de Kalman étendu (EKF) et de particules (PF).

- Amélioration des performances de localisation d'environ 22%, en termes d'erreur moyenne, en utilisant un modèle de filtrage linéaire, où le filtre KF est appliqué à la technique de fingerprinting à base du réseau GRNN.

- Utilisation d'une boussole digitale pour observer l'orientation du mobile, qui a permis de proposer une deuxième variante d'amélioration en considérant un modèle de filtrage non-linéaire, où les filtres EKF et PF ont été appliqués au réseau GRNN. Dans ce cas, les améliorations apportées sur la précision de localisation, en termes d'erreur moyenne, sont respectivement d'environ 39% dans le cas du filtre PF et 33% dans le cas du filtre EKF.

- Proposition d'une troisième variante d'amélioration en considérant un modèle de filtrage linéarisé, obtenue en remplaçant la variable non-linéaire de l'angle d'orientation par les deux variables des vitesses axiales du mobile. Dans ce cas, l'amélioration apportée sur la précision de localisation, en termes d'erreur moyenne, est d'environ 50% dans le cas de l'application du filtre PF au réseau GRNN.

- Analyse de l'effet des paramètres de filtrage sur les performances de localisation et de suivi.

1.3 Organisation du mémoire

À la suite de ce présent chapitre, le mémoire est structuré comme suit :

- Le Chapitre 2 consiste à décrire l'état de l'art des systèmes de localisation en environnements internes, en donnant leur principe de fonctionnement, leur classification et les solutions disponibles, en mettant en avant les performances et les contraintes qui leurs sont associées.

- Le chapitre 3 consiste à présenter le concept fonctionnel du système de localisation proposé ainsi que sa mise en expérimentation. Ce système est basé sur la technique de localisation de fingerprinting à base de signatures RSS, dont trois algorithmes de jumelages sont examinés : KNN, MLP et GRNN, pour en sortir celui qui a les meilleures performances de localisation pour l'environnement considéré. Les différents résultats expérimentaux ainsi que leurs analyses sont données.

- Le chapitre 4 consiste à appliquer le filtrage de Kalman et de ses extensions au système de localisation proposé dans le chapitre 3, pour améliorer les performances de localisation et de suivi. Dans cette partie, trois variantes d'amélioration relatives au modèle de filtrage caractérisant la dynamique du mobile (piéton), sont proposées, où les filtres de Kalman, de Kalman étendu et de particules sont appliqués. Les différents résultats expérimentaux et leurs analyses sont exposés.

- Finalement, le chapitre 5 présente une conclusion générale avec quelques perspectives de travaux futurs.

CHAPITRE 2

ÉTAT DE L'ART DES SYSTÈMES DE LOCALISATION EN ENVIRONNEMENTS INTERNES

2.1 Introduction

De nos jours, les services que peuvent offrir les systèmes de localisation sont divers et d'un grand intérêt, en citant entre autres, la nécessité de la localisation dans des situations d'urgence pour dépêcher des secours et porter assistance aux personnes en danger, la localisation des soldats, des policiers ou des pompiers pour l'accomplissement de leurs missions, ou encore la localisation dans les hôpitaux, des médecins ou des patients et le contrôle des équipements à coûts élevés.

Par conséquent, ces services nécessitent des données de localisation assez précises pour satisfaire les besoins des utilisateurs. Lorsque la personne ou l'objet à localiser se trouve dans un environnement externe ou à ciel ouvert, le problème ne se pose pas. Les systèmes de localisation par satellites (comme par exemple le GPS) peuvent accomplir cette tâche avec performance, tel qu'il est possible d'obtenir avec ces systèmes des précisions de localisation jusqu'à l'ordre de 1 mètre [4]. Cependant, dans des environnements internes (intérieur des bâtiments, espaces souterrains, zones urbaines à grande densité, etc.), les différents éléments de ces systèmes (satellites et terminaux) ne sont plus en visibilité directe, ce qui fait qu'ils fonctionnent en mode dégradé, ou souvent ne remplissent plus leur tâche de localisation. C'est pourquoi la communauté scientifique et industrielle semble considérer qu'il est tout à fait important de fournir des systèmes de localisation dédiés aux environnements internes, avec de bonnes performances et à développer des techniques pour y parvenir.

Aujourd'hui, les techniques de localisation sont multiples. La bonne maîtrise et connaissance de ces diverses techniques s'avèrent plus que nécessaires afin de judicieusement dimensionner son propre système de localisation. D'un autre côté, le système considéré ne doit pas être surdimensionné, sinon il va entraîner un surcoût soit au niveau de l'infrastructure, soit au niveau du terminal de localisation.

Par conséquent, ce premier chapitre a pour but de donner une description générale ou un état de l'art des systèmes de localisation en environnements internes, en donnant leur principe de fonctionnement, leur classification et les solutions disponibles, en mettant en avant les performances et les contraintes qui leurs sont associées.

2.2 Principe de fonctionnement des systèmes de localisation en environnements internes

D'une manière générale, le concept de localisation est un processus utilisé pour la collecte d'informations sur la position d'une personne, un équipement ou un terminal mobile, par rapport à une référence donnée. En littérature, ce concept peut être mentionné comme : *localisation* (localization ou location) [5], *radiolocalisation* (radiolocation) [6], *position de localisation* (position location) [7], *géolocalisation* (geolocation) [8] ou *positionnement* (positioning) [11]. Le système déployé pour estimer ou déterminer la position d'une entité donnée est appelé *système de localisation* ou *système de positionnement* [11]. Dans ce mémoire, le terme "localisation" sera souvent le plus utilisé.

Un diagramme de blocs illustrant le principe de fonctionnement d'un système de localisation dans un environnement interne a été suggéré par Pahlavan et al [8]. Il est composé d'un bloc de détection, d'un bloc d'estimation (algorithme de localisation) et d'un bloc de décision et de visualisation. La figure 2.1 (développée à partir de

[8]) montre les différents blocs avec leurs paramètres qui les caractérisent.

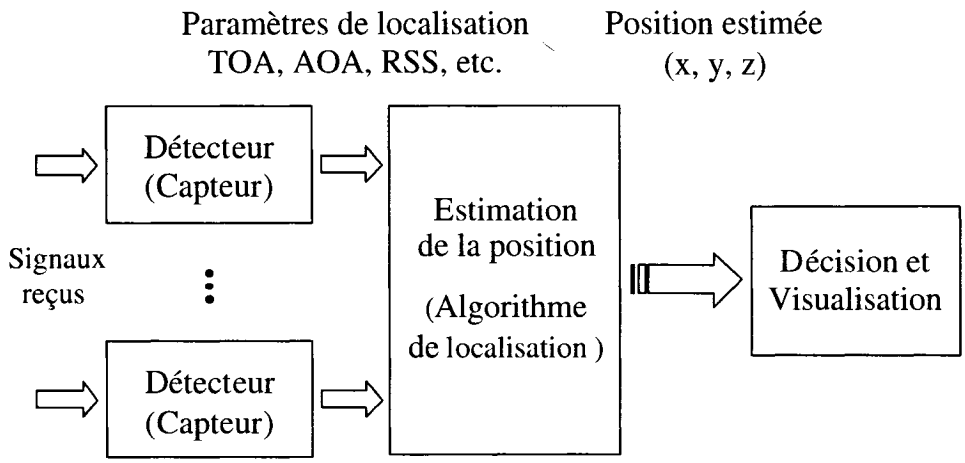


FIG. 2.1 Diagramme fonctionnel d'un système de localisation dans un environnement interne.

Le premier bloc représente les différents détecteurs (capteurs ou récepteurs) qui permettent de détecter les signaux transmis au mobile à partir des stations de base (émetteurs), en se basant sur une technologie de détection, qui peut être de type radiofréquence (RF), ultrason ou infrarouge (IR). Par la suite, le signal détecté sera converti sous forme de paramètres de localisation, qui peuvent être le temps d'arrivée (TOA : Time Of Arrival), la différence de temps d'arrivée (TDOA : Time Difference Of Arrival), l'angle d'arrivée (AOA : Angle Of Arrival) ou le signal de puissance reçu (RSS : Received Signal Strength) [8]. Étant donné une référence physique bien connue, ces paramètres permettent de déduire la distance ou la direction du mobile, ou de construire une base de données comportant les mesures RSS en chaque point de localisation du mobile. Quant au deuxième bloc, il représente l'algorithme de localisation qui permet d'estimer la position du mobile à partir des paramètres de localisation, en utilisant des approches de type déterministe [7, 8, 9] ou de type probabiliste [10]. Finalement, le dernier bloc représente le système de décision et de visualisation, qui décide de la manière de transmettre et de visualiser

l'information de localisation, suivant les besoins des utilisateurs.

2.3 Classification des systèmes de localisation en environnements internes

Les systèmes de localisation en environnements internes peuvent être classifiés suivant le type de la technologie de détection, la technique de localisation ou les propriétés qui les caractérisent [11]. La technologie de détection fait référence à la nature du signal utilisé par les détecteurs (récepteurs) du système. Quant à la technique de localisation, elle décrive les paramètres et la méthode d'estimation de la localisation utilisée. Finalement, les propriétés permettent d'évaluer le système de localisation.

2.3.1 Classification suivant la technologie de détection

Un système de localisation dépend essentiellement des caractéristiques et des limitations des signaux reçus et de la technologie de détection déployée. La diffraction, la réflexion, le temps de propagation et la dispersion sont parmi les facteurs qui affectent les caractéristiques des signaux dans un environnement interne. D'un autre côté, la portée, la largeur de bande, les interférences, la sécurité et le coût sont les limitations de la technologie de détection [12]. Communément, les technologies de détection les plus utilisés dans le cadre de la localisation en environnements internes sont de nature infrarouge, ultrason et radiofréquence. Une description de chacune de ces trois types de technologies est donnée dans ce qui suit.

2.3.1.1 Technologie de détection par infrarouge

Cette technologie est basée sur le signal infrarouge (IR), qui a les mêmes caractéristiques que la lumière visible, dont la vitesse de propagation est approximativement égale à 3×10^8 m/s. Ce type de signal ne peut pas pénétrer un mur ou un matériau, ce qui fait que sa portée est limitée dans les environnements internes. En général, la portée de ce type de signal est d'environ 5 m [6]. Aussi, il y a le problème de l'interférence de la lumière avec ce type de signal qui cause une dégradation de la précision de détection. Parmi les systèmes de localisation basés sur la technologie IR, on distingue le système "Active Badge" [13].

2.3.1.2 Technologie de détection ultrason

La technologie basée sur l'ultrason fonctionne sous une faible bande de fréquence (40 KHz), et avec une vitesse de propagation de celle du son (343 m/s). L'avantage de cette technologie est qu'elle est simple, peu coûteuse et peut atteindre des précisions de localisation de l'ordre de 1 cm [11]. Cependant, l'ultrason ne peut pas pénétrer la majorité des obstacles matériaux, ce qui fait que sa portée est limitée (3 à 10 mètres). Parmi les systèmes de localisation qui utilisent la technologie ultrasonore, on cite par exemples le système "Active Bat" [14] ou encore le système "Cricket" [15]. Ces systèmes permettent une localisation assez précise. Toutefois, ils sont sensibles aux bruits ambiants et nécessitent également la présence d'une visibilité directe entre l'émetteur et le récepteur.

2.3.1.3 Technologie de détection radiofréquence

Cette technologie est plus robuste par rapport aux deux autres technologies, puisqu'elle est basée sur le signal radiofréquence, qui a une grande capacité à pénétrer les matériaux ou les obstacles dans un environnement interne, ce qui fait qu'elle a une grande portée de détection. Aussi, le signal RF a une grande vitesse de propagation, approximativement 3×10^8 m/s. Toutefois, le problème par rapport à cette technologie est que les fréquences allouées sont restreintes à l'usage public. Il y a aussi le problème de multitrajet ("multipath") qui affecte le signal RF. Un des systèmes de localisation basé sur ce type de technologie est le système RADAR [1].

2.3.2 Classification suivant la technique de localisation

En plus de la technologie de détection, les systèmes de localisation sont classifiés suivant la technique de localisation utilisée pour estimer la position du mobile. Généralement, les techniques de localisation qui se présentent sont celles basées sur la distance, l'angle, la signature (fingerprint) ou encore sur une combinaison de ces dernières. La technique de mesure basée sur la distance est souvent appelée *latération*, quant à la technique de localisation basée sur l'angle, elle est souvent appelée *angulation*. Ces deux techniques sont des sous catégories de la *triangulation* [16], qui utilise la géométrie triangulaire pour déterminer la position du mobile. Le diagramme de la figure 2.2 (développé à partir de [11]) montre les trois techniques de localisation classiques (latération, angulation et fingerprinting), qui sont basées sur la technologie de détection radiofréquence. Les techniques de latération et d'angulation ont été étudiées d'une manière extensive dans le cas des systèmes de localisation en environnements externes, là où il y a une visibilité directe entre l'émetteur et le récepteur. En environnements internes, ces deux techniques font face à beaucoup de limitations, telles que la non-visibilité directe entre l'émetteur

et le récepteur, le phénomène du multitrajet et les interférences, ce qui rend leur utilisation non souhaitables dans ces environnements. Quant à la technique de fingerprinting (ou de signatures), celle-ci a reçu ces derniers temps beaucoup d'intérêt dans le domaine de la localisation dans un environnement interne, du fait qu'elle est mieux adaptée à ces environnement et sa mise en oeuvre est simple. La description détaillée de chacune des trois techniques de localisation qu'on vient de citer, est donnée dans ce qui suit.

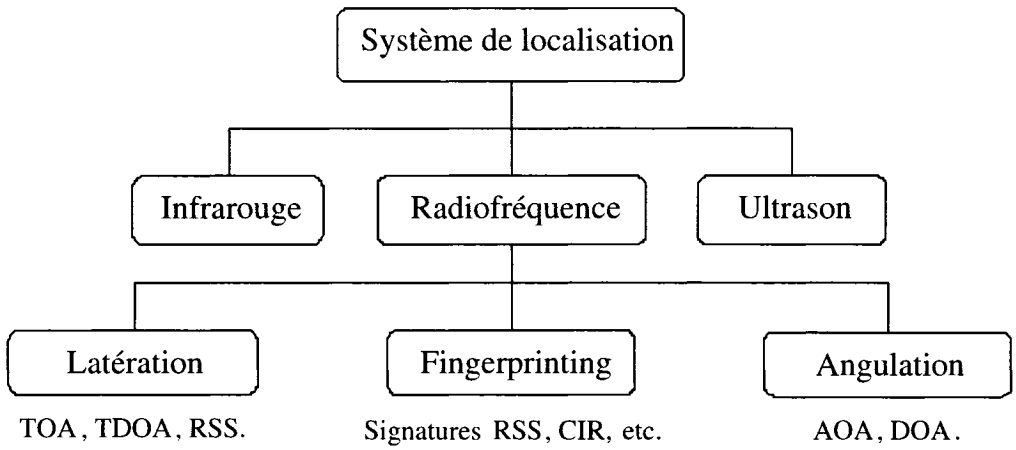


FIG. 2.2 Techniques de localisation classiques.

2.3.2.1 Technique de localisation par latération

La technique de localisation par latération est basée sur la mesure de la distance entre l'émetteur (station de base) et le récepteur (station mobile). Elle permet d'estimer la position du mobile à partir du temps de propagation, qui peut être le temps d'arrivée (TOA) ou la différence de temps d'arrivée (TDOA) ou à partir de l'atténuation du signal de puissance reçu (RSS).

• **Approche TOA :** Cette approche est basée sur la mesure du temps de propagation entre la station de base et le mobile. Ayant obtenu l'information sur le TOA, la

distance entre la station de base et le mobile est déduite facilement en multipliant le temps TOA par la vitesse de la lumière. Chaque distance estimée va former, pour une localisation en 2D, un cercle par rapport à la station de base. La position estimée du mobile est déduite à partir de la région de l'intersection des cercles d'au moins trois stations de base (voir figure 2.3). La position réelle du mobile est le point d'intersection des cercles en pointillés. Malheuresement, en pratique les distances estimées entre le mobile et les stations de base sont plus grandes que les distances réelles (à cause du NLOS qui retarde la propagation du signal), et l'intersection des cercles formés par ces distances, donne plutôt une région d'estimation où se trouve le mobile (région hachurée sur la figure 2.3). Une technique statistique de localisation directe ou itérative [17] est alors utilisée pour situer plus précisément le mobile dans cette région d'incertitude. L'approche TOA dépend principalement de la précision de la synchronisation entre les stations de base et le mobile, et de l'algorithme d'estimation du temps de propagation TOA.

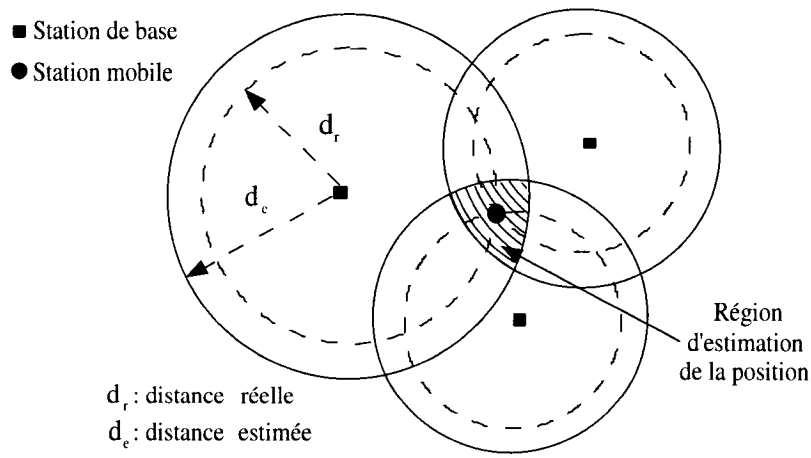


FIG. 2.3 Approche de localisation par TOA.

• **Approche TDOA :** Cette approche a le même principe de fonctionnement que l'approche TOA, qui est basé sur la mesure de distance entre la station de base et le

mobile. Toutefois, elle présente un avantage par rapport à cette dernière en évitant la synchronisation entre le mobile et les stations de base. Dans ce cas, des paires de stations de base synchronisées entre elles sont utilisées, tel que la différence du temps d'arrivée du signal émis par le mobile à chaque paire de stations de base est mesurée. Dans un espace de localisation de 2 dimensions, l'ensemble des positions possible du mobile pour avoir une même différence de temps d'arrivée à une paire de stations de base donnée forme une hyperbole. La position du mobile est estimée facilement en s'appuyant sur de simples relations géométriques, faites avec au moins trois paires de stations de base. L'approche TDOA est illustrée sur la figure 2.4. Les hyperboles en pointillé sont celles qui auraient été obtenues si toutes les différences des temps d'arrivée avaient été mesurées avec exactitude. Leur intersection donne la position exacte du mobile. Toutefois, comme des erreurs de mesure du TDOA sont presque toujours présentes, la localisation du mobile est plutôt une région d'estimation de la position (région hachurée sur la figure 2.4). Les limitations de la technique de localisation basée sur le TDOA sont dues principalement aux caractéristiques de l'environnement interne, à savoir le NLOS et le multitrajet.

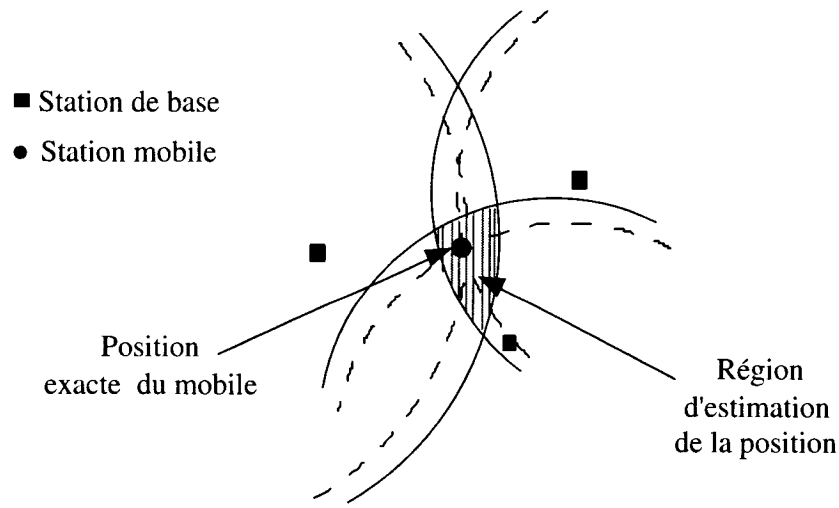


FIG. 2.4 Approche de localisation par TDOA.

• **Approche basée sur l'atténuation du RSS :** Cette approche est aussi une catégorie de la technique de latération, basée sur la mesure de la distance. Mais au lieu de mesurer cette distance à partir du temps (ou la différence du temps) de propagation du signal, elle le fait à partir de l'atténuation du signal de puissance reçu au mobile. En effet, l'atténuation du signal RSS est fonction de la distance séparant l'émetteur (station de base) et le récepteur (station mobile). En connaissant le modèle d'atténuation ("pathloss") du canal de propagation, il est facile de déduire la distance entre le mobile et la station de base. Après avoir mesuré cette distance, la position du mobile sera estimée de la même façon que dans le cas de l'approche TOA. Les limitations de localisation observées avec cette approche sont liées principalement aux caractéristiques de l'environnement interne (NLOS et multitrajet), au modèle de "pathloss" utilisé et à l'algorithme d'estimation de la position du mobile. Pour toutes ces raisons, l'approche basée sur l'atténuation du signal RSS n'est pas souhaitable pour la localisation dans les environnements internes. En pratique, l'utilisation de cette technique nécessite une détermination régulière des paramètres d'atténuation pour tenir compte du changement dynamique de l'environnement.

2.3.2.2 Technique de localisation par angulation

La technique de localisation par angulation, basée sur la mesure de l'angle d'arrivée (AOA) ou la direction d'arrivée (DOA), permet de déterminer la position du mobile par la mesure de l'angle d'incidence du signal reçu au mobile, en utilisant des antennes directionnelles ou encore des réseaux d'antennes, qui sont placées au niveau des stations de base. En utilisant de simples relations géométriques et au moins deux stations de base, la position du mobile peut être calculée à partir de l'intersection des deux lignes radiales des angles d'incidences estimés (voir figure 2.5). Les directions en pointillé sont les directions réelles des trajets directs, et leur intersection donne la position exacte du mobile. Toutefois, comme les antennes

de réception disposent d'une marge d'erreur et d'une résolution angulaire limitée. Donc, chaque station de base localise le mobile plutôt dans un faisceau d'estimation. La localisation du mobile se trouve ainsi sous forme d'une région d'estimation formée par l'intersection de ces deux faisceaux (région hachurée sur la figure 2.5). Comme cette technique prévoit l'utilisation des antennes directionnelles ou encore des réseaux d'antennes pour l'estimation de l'angle d'incidence, son utilisation est moins pratique pour des raisons de coût élevé et aussi le fait qu'elle nécessite une visibilité directe entre les stations de bases et le mobile, ce qui n'est pas le cas dans des environnements internes.

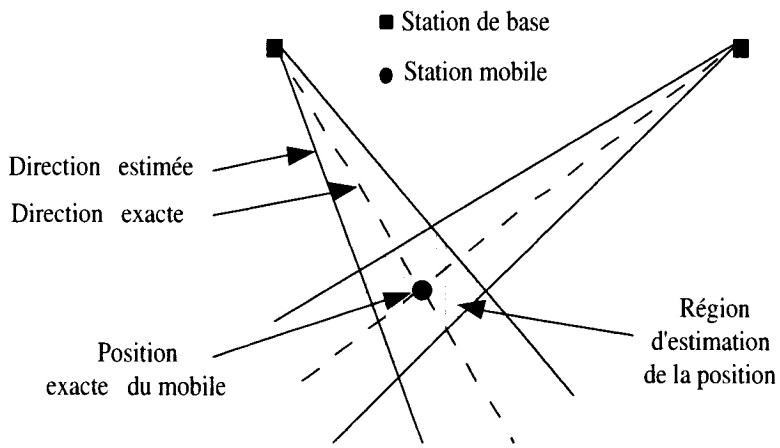


FIG. 2.5 Approche de localisation par AOA.

2.3.2.3 Technique de localisation de fingerprinting

La technique de localisation de fingerprinting tient son nom du terme "fingerprint", qui signifie empreinte digitale ou signature. Cette technique de localisation est simple d'usage par rapport aux deux précédentes techniques, et permet de déterminer la position du mobile à partir de données brutes provenant de la station de base (émetteur). Toutefois, elle requiert une étape de calibration. Le processus

de localisation avec cette technique se fait essentiellement en deux phases : une phase d'entraînement ou d'apprentissage (off-line phase), et une phase d'appel ou d'estimation en temps-réel (on-line phase).

• **Phase d'entraînement :** Cette phase consiste à construire une base de signatures qui, à certaines positions de localisation dans l'environnement considéré, associe un ensemble d'éléments caractérisant cette position, qui doivent permettre de différencier chacune des positions par rapport aux autres positions de l'environnement de localisation. Les éléments pouvant être considérés pour composer les signatures de localisation sont, par exemples, la puissance du signal, la réponse impulsionnelle du canal, les directions d'arrivées ou encore le temps d'arrivée [17]. Selon un algorithme de localisation (aussi appelé algorithme de jumelage), la base de signatures construite est utilisée pour son entraînement et pour lui servir de base de comparaison (pattern-matching).

• **Phase d'appel :** Cette phase consiste à estimer la localisation actuelle du mobile. On recueille au niveau du mobile les paramètres de localisation, qui doivent être identiques à ceux utilisés dans la première phase lors de l'enregistrement de la base de données de signatures. L'algorithme de jumelage devra déterminer la position actuelle du mobile à partir de ces paramètres de localisation, en connaissant les signatures recueillies lors de la première phase avec les positions qui leurs correspondent. En littérature, les algorithmes de localisation peuvent être classifiés sous deux approches : déterministe ou probabiliste, c'est-à-dire suivant le type de relation entre la signature et l'information de localisation. Parmi les algorithmes de localisation de type déterministe, on cite l'algorithme des K-plus proches voisins (KNN : K-Nearest Neighbors) et l'algorithme à base des réseaux de neurones artificiels (ANN : Artificiel Neural Networks). Quant aux algorithmes de localisation de type probabiliste, on cite les machines à support de vecteurs (SVM : Support Vector Machine). En ce qui concerne les avantages de la technique de fingerprin-

ting, on cite le fait que celle-ci ne nécessite pas de visibilité directe entre la station de base et le mobile, comme fût le cas dans les précédentes techniques. Un autre avantage intéressant de cette technique est le fait que le profil de propagation du signal par multitrajet est incorporé comme information dans les signatures. En effet, les signatures sont prises dans l'environnement réel et non à partir de simulations utilisant une modélisation mathématique du canal de propagation. Cette technique donne généralement des erreurs de localisation beaucoup plus faibles que les techniques basées sur la modélisation mathématique du canal de propagation. Par contre, parmi les inconvénients de cette technique on cite le fait que la précision de localisation se dégrade rapidement quand le canal de propagation subit une fluctuation importante, comme par exemple, la variation dynamique du canal dans un centre d'achat. Dans ce cas-ci, la signature d'un point donné change considérablement au cours du temps par rapport au mouvement des personnes et des objets. Le canal de propagation devient ainsi non-statique (dynamique). Plus de détails sur la technique de fingerprinting seront donnés dans le prochain chapitre.

2.3.3 Classification suivant les propriétés

Un système de localisation peut aussi être classifié suivant un ensemble de propriétés qui permettent de l'évaluer. Les propriétés les plus pertinentes pour évaluer un système de localisation sont données comme suit [16] :

• **Performances :** Elles dépendent de la précision de la localisation, du temps de traitement, de la capacité du système à traiter un certain nombre de données de localisation et de la portée du système.

• **Nature de la localisation :** Elle peut être physique, représentée par des coordonnées référentielles (latitude, longitude, altitude), ou symbolique, représentée

par la proximité à une référence bien connue dans l'entourage de localisation.

• **Approche d'implantation :** Elle peut être locale (décentralisée ou encore dite descendante), où le processus de localisation est fait au niveau du mobile (self-positioning), ou externe (centralisée ou encore dite ascendante), où le processus de localisation est fait au niveau de la station de base ou au niveau d'un serveur de localisation centralisé.

• **Coût :** Il dépend de l'infrastructure, de la technologie de détection déployée et de la largeur de bande allouée.

• **Complexité :** Elle dépend de la mise en oeuvre pratique du système de localisation et de la technique de localisation utilisée.

• **Sécurité :** Elle dépend de l'accès à l'information sur la localisation et du type de la technologie de détection déployée.

Plus de détails sur les propriétés, qui permettent d'évaluer les systèmes de localisation, peuvent être trouvés dans [11, 16, 18].

2.4 Exemples de systèmes de localisation en environnements internes

2.4.1 Le système de localisation Active Badge

Le système "Active Badge" [13] est l'un des premiers systèmes de localisation proposé pour des environnements internes. Le processus de localisation de ce système, basé sur la technologie infrarouge, est brièvement décrit comme suit : le mobile à localiser est équipé d'un badge électronique émettant un signal infrarouge toutes les 10 secondes, alors que les récepteurs sont installés au plafond de chaque pièce

de l'environnement d'application. Ces récepteurs sont reliés entre eux pour former un réseau permettant de détecter le badge actif. La position du badge est alors déterminée à partir de la position du récepteur le plus proche. L'inconvénient de ce système est qu'il nécessite l'existence d'une visibilité directe entre le badge et le récepteur, ce qui n'est pas toujours le cas dans un environnement interne. Pour cela, le badge doit être toujours porté sur l'extérieur de l'habillement (attaché sur la chemise ou la blouse de la personne). Un autre inconvénient de ce système est la présence de la lumière du jour qui perturbe la transmission infrarouge entre l'émetteur et le récepteur.

2.4.2 Le système de localisation Active Bat

Le système de localisation "Active Bat" [14] est basé sur une architecture centralisée. Il permet d'offrir un service de localisation meilleur par rapport au système "Active Badge", en combinant à la fois les technologies RF et ultrason. Dans ce système, dès qu'un émetteur mobile (Bat) est détecté, un signal RF est émis par des récepteurs reliés à un PC. Le Bat répond ainsi en émettant une série d'impulsions ultrasonores. Le PC, servant de centre nerveux au système de récepteurs situés au plafond, émet vers chacun des récepteurs un signal de reset via la liaison sérielle. Ce signal est émis en en même temps que les signaux émis par les récepteurs à destination du Bat à localiser. Puis, le PC interroge chacun des récepteurs composant le réseau, en récupérant les intervalles de temps entre le signal de reset et celui de la détection de l'impulsion du signal ultrason. La position du bat est obtenue alors par latération. La précision de ce système de localisation est d'environ 9 cm pour 95% des cas de localisation. Toutefois, la mise en place d'un tel service de localisation nécessite un effort au niveau du déploiement des récepteurs et leur interconnexion avec le système de contrôle centralisé.

2.4.3 Le système de localisation Cricket

Le système de localisation "Cricket" [15] repose sur les mêmes principes de localisation du système "Active Bat". La principale différence est que le système "Cricket" est conçu d'une façon à être décentralisé (traitement effectué par l'équipement mobile) et moins coûteux. Comme pour le système "Active Bat", ce système combine entre les technologies ultrason et RF. Cette combinaison permet d'estimer la distance émetteur/récepteur puis la position occupée par le mobile. Des émetteurs sont placés au plafond de l'immeuble et émettent des signaux RF contenant des informations de localisation. En même temps que ces signaux RF sont émis, une onde ultrasonore est émise à partir de ce même émetteur. Le récepteur reçoit successivement les signaux RF et ultrason. Il effectue une corrélation de ces deux signaux reçus pour extraire la différence de temps d'arrivée (TDOA) entre chacun de ces signaux. Ceci permet d'estimer la distance le séparant de l'émetteur qui a émis ces deux signaux. En réitérant cette même mesure avec plusieurs émetteurs, on détermine précisément la position du mobile (récepteur). Les limitations de ce système sont le risque d'interférences et le fait qu'il nécessite une visibilité directe entre l'émetteur et le récepteur.

2.4.4 Système de localisation RADAR

Le système de localisation RADAR [1] a été proposée par P. Bahl et al. (Microsoft Research) en 2000. Il utilise la technologie IEEE 802.11b WLAN [19], et la technique de localisation de fingerprinting pour localiser un mobile. Le processus de localisation est basé sur une approche centralisée, où trois stations de base (points d'accès), équipées chacune d'une carte réseau sans-fil, reçoivent périodiquement un signal RF du mobile et déterminent la puissance de ce signal. La signature à un point de localisation donné est représentée par un triplet de puissances du signal

émis par le mobile et reçu par les trois stations de base. Les signatures sont prises à 70 positions différentes, de façon à couvrir tout l'espace de localisation. L'ensemble des signatures avec les positions qui leurs correspondent forment ainsi la base de signatures, qui sera utilisée pour la localisation du mobile. L'estimation de la position actuelle du mobile est faite en mesurant (on-line) les différentes puissances reçues par les trois points d'accès, qui vont former la signature de localisation actuelle, puis l'algorithme des K-plus proches voisins (KNN) est utilisé pour rechercher dans la base de données, la signature qui va minimiser la distance Euclidienne par rapport à la signature actuelle. La position actuelle du mobile est celle qui correspond, dans la base de données, à la signature qui assure la distance Euclidienne minimale. La précision de la localisation, rapportée en erreur médiane, est de 2.94 m.

2.4.5 Système de localisation RFID LANDMARC

Le système de localisation LANDMARC [20] est basé sur la technologie RFID (radio frequency identification). Ce système est constitué d'un lecteur RFID qui émet fréquemment un signal RF de faible puissance par l'intermédiaire de son antenne. Il fournit ainsi l'énergie suffisante à la puce électronique d'un tag donné, pour que celui-ci retransmette à travers sa propre antenne un signal modulé représentant l'identification numérique du tag, qui correspond à une séquence de bits stockés dans la puce. Pour localiser le tag mobile, le système LANDMARC propose une technique qui consiste à utiliser un réseau de tags de référence. Cette technique consiste à comparer la puissance reçue à partir du tag à localiser avec celles reçues à partir des tags du réseau de référence (voir figure 2.6). En réalisant cette comparaison, afin de définir les plus proches transpondeurs (tags), on est capable de localiser le tag mobile. Toutefois, les précisons obtenues avec ce système ne sont pas très bonnes, où typiquement elles sont de l'ordre de la demie-longueur du réseau de tags de référence (L/2).

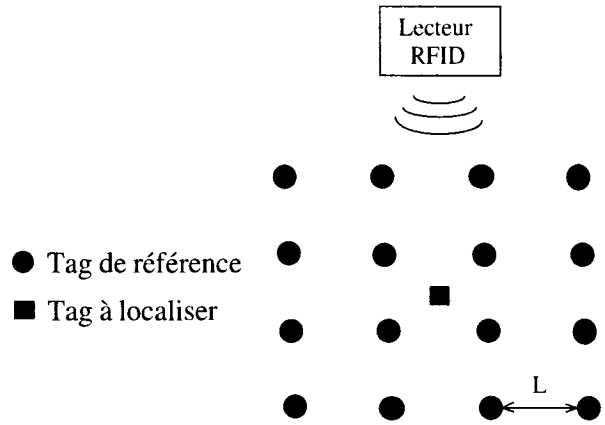


FIG. 2.6 Système de localisation RFID LANDMARC.

Les exemples de systèmes de localisation qu'on vient de citer, constituent un grand pas en avant, en termes de développement des solutions de localisation en environnements internes. Cependant, ils ont quelques inconvénients, tels que la limitation des signaux infrarouge et ultrasons à pouvoir pénétrer des obstacles, et aussi les différents coûts d'infrastructure, des badges électroniques et des récepteurs, qui peuvent être significatifs par rapport à la surface intérieure où le système a été déployé. Il est à noter que l'approche basée sur la mesure de l'angle AOA, n'a pas été utilisée jusqu'à maintenant à cause du problème de multitrajet lié à l'environnement interne et à la résolution angulaire limitée des antennes.

2.5 Conclusion

Dans ce chapitre, une description générale des systèmes de localisation en environnements internes a été présentée, en donnant leur principe de fonctionnement, leur classification et leurs propriétés. Cette description permet d'avoir un bagage de connaissances assez suffisant pour bien concevoir son système de localisation.

Cette étude littéraire a été faite par rapport aux environnements internes, là où il y a beaucoup de contraintes sur la localisation et où des solutions de localisation satisfaisantes tardent à venir, contrairement aux environnements externes, où les solutions de localisation sont très performantes, en citant entre autres le GPS. En se basant sur toutes les informations rapportées ici, les deux prochains chapitres vont se consacrer à la conception et à la mise en expérimentation d'une solution de localisation dédiée à un environnement interne, avec des performances acceptables.

CHAPITRE 3

MODÉLISATION ET EXPÉRIMENTATION D'UN SYSTÈME DE LOCALISATION DANS UN ENVIRONNEMENT INTERNE

3.1 Introduction

Aujourd'hui, beaucoup d'immeubles publics et privés sont équipés de la technologie sans fil IEEE 802.11 (WLAN), une technologie radiofréquence populaire et peu coûteuse, ce qui a ouvert une nouvelle opportunité à développer des systèmes de localisation en environnements internes. Ce nouvel essor a suscité un grand intérêt de recherche pour la technique de fingerprinting, qui est considérée la mieux adaptée à ce type de technologie [1, 9, 17, 21]. Ainsi, un système de localisation exploitant la seule information du niveau de puissance reçu à partir des points d'accès du réseau WLAN, permet de mettre en œuvre aisément et avec un moindre coût un service de localisation.

L'objectif de ce chapitre consiste à modéliser et à expérimenter un système de localisation dans un environnement interne, en exploitant les deux aspects cités ci-dessus, à savoir une infrastructure WLAN existante, et une technique de localisation de fingerprinting simple à implémenter et mieux adaptée aux environnements internes. Ce chapitre est constitué principalement de trois parties. La première partie est consacrée à la description fonctionnelle et à la modélisation du système de localisation proposé, qui est basé sur la technologie IEEE 802.11 et la technique de localisation de fingerprinting. Dans la deuxième partie, le processus expérimental est exposé. Finalement, la dernière partie donne les résultats expérimentaux avec leurs analyses.

3.2 Description fonctionnelle du système de localisation proposé

Cette section donne une description fonctionnelle du système de localisation proposé. La figure 3.1 illustre le modèle fonctionnel de ce système, en caractérisant chacune des composantes présentées dans le chapitre 2 (voir figure 2.1).

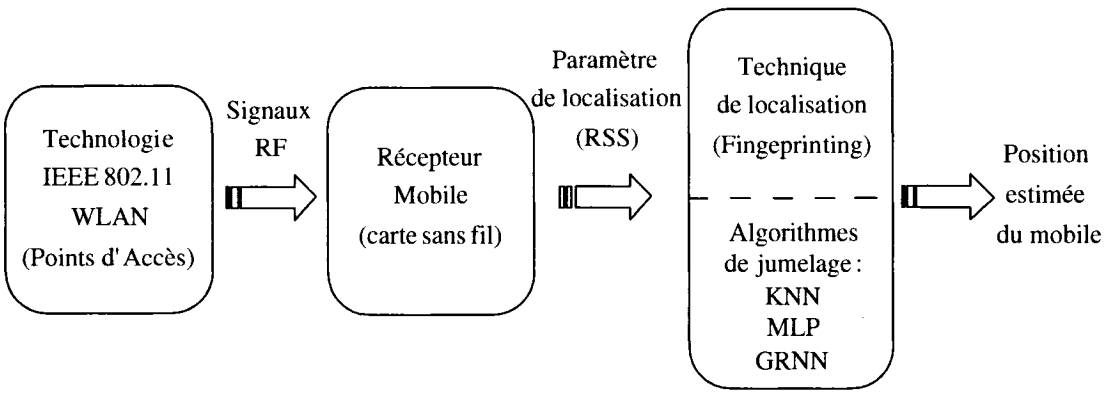


FIG. 3.1 Modèle fonctionnel du système de localisation proposé.

Le choix de la structure du système proposé est dû, comme cité précédemment, au fait que la technologie WLAN est déjà existante et que la technique de localisation de fingerprinting est simple à implémenter et mieux adaptée aux environnements internes. En effet, le canal de propagation dans un environnement interne est caractérisé par un profil de propagation du signal par multitrajet, et par une faible probabilité de recevoir le signal du trajet direct. Dans ces conditions, les techniques de localisation basées sur la modélisation mathématique du canal de propagation ou sur la triangulation, s'avèrent non concluantes à fournir une précision de localisation acceptable. Quant à la technique de fingerprinting, elle a l'avantage d'utiliser le profil de propagation du signal par multitrajet et toutes les atténuations comme des informations utiles pour la localisation du mobile, puisqu'ils sont inclus dans les signatures formées à partir des mesures réelles, effectuées dans l'environnement

d'application. Elle est donc beaucoup plus adaptée aux environnements internes et va permettre de donner une précision de localisation supérieure à celle obtenue avec les techniques basées sur la triangulation et la modélisation mathématique du canal de propagation.

Dans ce qui suit, la description fonctionnelle du système de localisation proposé, est exposée suivant la technologie de détection et la technique de localisation utilisées.

3.2.1 Technologie de détection (WLAN)

Le système de localisation proposé est basé sur la technologie sans fil IEEE 802.11 WLAN, où les signaux transmis sont de nature radiofréquence. Dans ce qui suit, le standard IEEE 802.11 ainsi que les caractéristiques de l'environnement interne dont il est déployé sont présentés.

3.2.1.1 Standard IEEE 802.11 WLAN

La technologie IEEE 802.11 (ISO/IEC 8802-11) est un standard international [19] décrivant les caractéristiques du réseau WLAN. Souvent, par abus de langage (et pour des raisons de marketing) le nom de la norme IEEE 802.11 se confond avec le nom de la certification Wi-Fi (Wireless Fidelity), qui correspond initialement au nom donné à la certification délivrée par la WECA (Wireless Ethernet Compatibility Alliance), l'organisme chargé de maintenir l'interopérabilité entre les matériels répondant à la norme 802.11. Cette norme s'attache à définir les couches basses du modèle OSI, pour une liaison sans fil utilisant des ondes électromagnétiques, c'est-à-dire, la couche physique et la couche liaison de données, constituée elle-même de deux sous-couches : le contrôle de la liaison logique (LLC : "Logical Link Control") et le contrôle d'accès au support (MAC : "Medium Access Control").

En pratique, le standard IEEE 802.11 WLAN permet de relier des ordinateurs portables, des machines de bureau, des assistants personnels (PDA : Personal Digital Assistant), des objets communicants ou même des périphériques, à une liaison haut débit (de 11 Mbit/s en 802.11b à 54 Mbit/s en 802.11a/g) sur un rayon de plusieurs dizaines de mètres dans un environnement interne. Dans un environnement ouvert, la portée peut atteindre plusieurs centaines de mètres. Les différentes déclinaisons de ce standard utilisent les bandes de fréquences libres de 2.4 GHz et 5.8 GHz.

Ainsi, de nombreux produits 802.11 a/b/g sont installés dans des zones à forte concentration d'utilisateurs (hôpitaux, aéroports, centres commerciaux, universités, etc.). Des informations sur la communication sont accessibles et disponibles au niveau des interfaces physiques des cartes de communication. Les informations proviennent des trames émises par les points d'accès (AP) du réseau WLAN. Parmi ces trames, on trouve des trames d'authentification, d'association, etc [22]. Mais, la trame qui sera exploitée dans le cadre de la localisation est la trame de balisage (balise ou "beacon"). En effet, le point d'accès émet périodiquement ces trames afin de signaler sa présence, et pour émettre des informations telles qu'une référence temporelle, le SSID (Service Set Identifier) et d'autres informations caractérisant ce point d'accès. Ces trames sont captées par les clients se trouvant dans la couverture radio du point d'accès émettant. Le client, de son côté, scrute en permanence tous les canaux disponibles afin de capter ces balises qui lui permettent de déterminer le point d'accès avec lequel il est plus avantageux de s'associer. Les balises sont émises périodiquement sur un intervalle de 100 ms, où cette valeur assure de bonnes performances dans bon nombre d'applications [22].

Par conséquent, l'utilisation de ces trames de balisage est un bon moyen pour effectuer la localisation, puisqu'elles sont émises périodiquement et naturellement par le réseau WLAN. Il n'est pas nécessaire de modifier la nature intrinsèque du réseau pour établir les opérations nécessaires à la localisation. Par la suite, la tech-

nique de localisation de fingerprinting est basée sur l'exploitation des informations disponibles dans ces balises.

3.2.1.2 Caractéristiques d'un environnement interne WLAN

Avant d'aborder la technique de localisation utilisée, il est intéressant de connaître les caractéristiques d'un environnement interne et son effet sur le signal émis par le réseau WLAN. Un des éléments qui caractérise l'environnement interne est le phénomène de multitrajet ("multipath"), causé par les différentes diffractions, réflexions et réfractions. Ce dernier affecte beaucoup le signal émis par les points d'accès d'un réseau WLAN, et rend son comportement aléatoire. En effet, l'étude de [23] faite à l'intérieur d'un immeuble équipé par un réseau WLAN, a montré que la distribution du signal de puissance provenant du réseau WLAN est une log-normale pour des données qui ont une moyenne, une médiane et un mode assez proches. De même, il a été montré que la relation entre la portée et l'écart type du signal reçu est linéaire et que lorsque la distance entre l'émetteur et le récepteur augmente l'écart type augmente de même. Parmi les autres éléments qui affectent le signal IEEE 802.11, il y a le corps humain qui bloque le trajet du signal lors de la localisation. En fait, comme le corps humain est constitué de l'eau et que ce dernier a une fréquence de résonnance de 2.4 GHz, il devient un élément significatif dans l'atténuation du signal IEEE 802.11 [5, 24]. D'un autre côté, l'étude faite dans [1] a montré que l'orientation de l'usager affecte le profil du signal du coté où le point d'accès a été bloqué par ce dernier. C'est pour cela qu'il a suggéré d'inclure l'orientation comme une information de localisation. D'autres éléments qui sont incontrôlables, comme la température, le mouvement de l'air et les interférences avec d'autres opérateurs de même fréquence, causent des variations du niveau du signal reçu [25].

3.2.2 Technique de localisation (fingerprinting)

La technique de fingerprinting (ou de signatures) est utilisée pour l'estimation de la position du mobile. Elle est simple à implémenter et mieux adaptée aux environnements internes équipés par les réseaux WLANs. Comme présenté dans le chapitre 2, cette technique est basée sur la construction d'une base de signatures ("fingerprints") qui, à certaines positions dans l'environnement de localisation considéré, associe un ensemble de signatures uniques caractérisant cette position de localisation. Battiti et al [9] ont mentionné que l'information sur la localisation peut être sous forme de coordonnées, qui correspondent aux trois coordonnées cartésiennes (x, y, z) et aux angles d'orientation exprimés en coordonnées sphériques, ou sous forme de variables indicatrices (Nord, Est, Sud, Ouest). Pour ce qui est de l'information de la signature de localisation, tous les travaux de recherches, qui ont été faits dans le cadre des réseaux WLANs, montrent que le niveau du signal de puissance reçu (RSS) est la signature RF la plus simple à extraire et la plus effective jusqu'à présent pour la technique de localisation de fingerprinting. Cette information est obtenue à partir des balises émises périodiquement par les points d'accès du réseau WLAN, et reçues au niveau de la carte sans fil de l'équipement mobile. Aucune modification sur les équipements ne doit être effectuée. L'information RSS a été utilisée pour la première fois par [1].

Le principe de fonctionnement de l'approche de fingerprinting est illustré par le diagramme de la figure 3.2. Dans cette approche, souvent on dénote un vecteur de signatures RSS qui correspond à une position (x, y) donnée. La dimension du vecteur RSS est déterminée par le nombre de points d'accès qui sont à l'écoute dans l'environnement d'application.

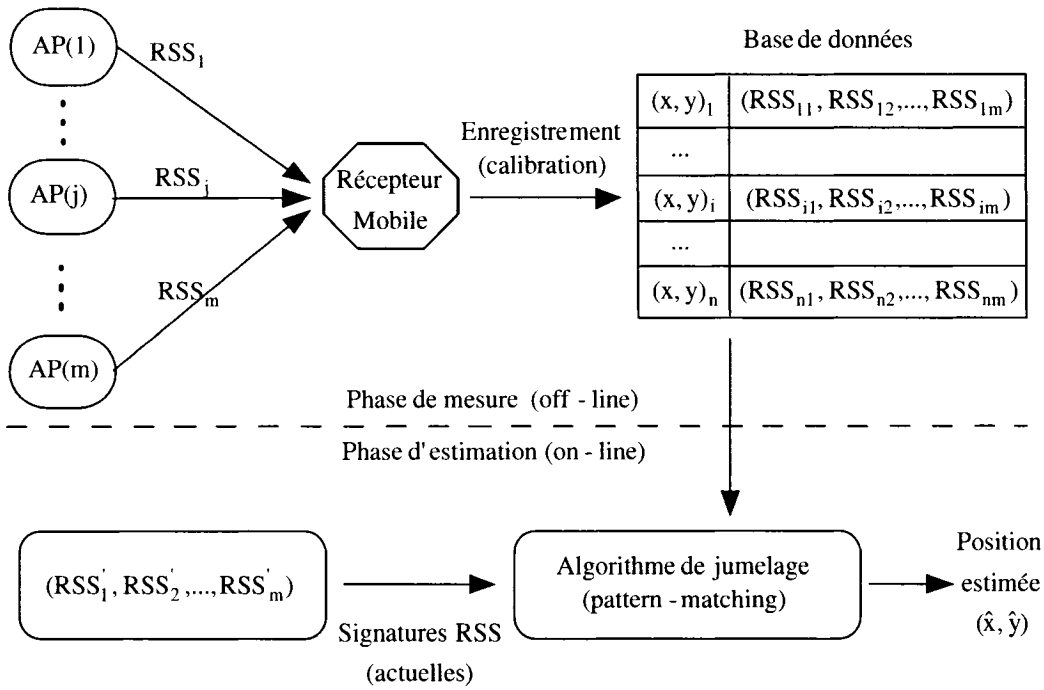


FIG. 3.2 Principe de localisation par la technique de fingerprinting.

Pour créer la base de données de localisation, formée par les couples (vecteurs RSS, position (x, y)), en chaque position de localisation donnée, plusieurs échantillons du vecteur RSS sont recueillis (off-line) durant une période de temps. Par la suite, la moyenne du vecteur RSS est calculée et enregistrée avec la position qui lui correspond dans une base de données. Après avoir enregistré la base de données, formée par les signatures RSS et les positions de localisation qui leurs correspondent, un algorithme de localisation (algorithme de jumelage ou "pattern-matching") va effectuer un apprentissage ou un entraînement sur cette base de données, et exploiter la dépendance entre les signatures RSS et les informations de localisation pour estimer la position actuelle du mobile. Dans notre travail, trois algorithmes de jumelage de types déterministes sont considérés, qui sont l'algorithmedes K-plus proches voisins et deux algorithmes à base de réseaux de neurones artificiels de types MLP et GRNN.

3.2.2.1 Algorithme des K-plus proches voisins

L'algorithme des K-plus proches voisins (KNN : K-Nearest Neighbors) [1] est basé sur une approche déterministe, puisqu'il utilise des grandeurs constantes pour les signatures RSS. L'algorithme utilise les K mesures RSS de la base de données de signatures, qui possèdent le plus de similitudes avec la mesure RSS instantanée (actuelle). Par la suite, le critère de minimisation de la distance Euclidienne est utilisé pour déterminer le degré de similitude entre la mesure RSS instantanée et celles contenues dans la base de données de signatures. La position du mobile sera déduite en fonction de cette similitude et par rapport aux positions qui correspondent aux K plus proches signatures extraites de la base de données.

Soit RSS_i le vecteur de signatures (mesuré off-line) qui correspond à la $i^{ème}$ position du mobile, avec $RSS_i = (RSS_{i1}, ..., RSS_{ij}, .., RSS_{im})$, et soit RSS' le vecteur de signatures (mesuré on-line) qui correspond à la position actuelle du mobile, avec $RSS' = (RSS'_1, ..., RSS'_j, .., RSS'_m)$, où m représente le nombre de points d'accès qui sont à l'écoute par la carte sans fil de l'équipement mobile. La distance Euclidienne entre ces deux vecteurs est donnée par la formule suivante :

$$d_i = \sqrt{\sum_{j=1}^{m}(RSS'_j - RSS_{ij})^2} \tag{3.1}$$

À partir de l'équation (3.1), on forme l'ensemble des K indices i_k (k=1,...,K), dont les distances Euclidienne sont les plus faibles parmi toutes celles possibles dans la base de données, où $i = 1, ..., n$, avec n est le nombre de positions de localisation où les mesures de signatures ont été prises (off-line) pour le remplissage de la base de données. On note cet ensemble E_K, tel que :

$$E_K = \{i_k \ / \ i_k = i \text{ pour } \arg\min_i(d_i), \text{ avec : } k = 1, ..., K \text{ et } K \leq m\} \tag{3.2}$$

La position estimée du mobile est donnée par :

$$(\hat{x}, \hat{y}) = \frac{\sum_{k=1}^{K} \frac{1}{d_{i_k}} \cdot (x_{i_k}, y_{i_k})}{\sum_{k=1}^{K} \frac{1}{d_{i_k}}}, \text{ avec } : i_k \in E_K \tag{3.3}$$

où (x_{i_k}, y_{i_k}) est la position du mobile qui correspond dans la base de données au vecteur signature RSS_{i_k}. Il est à noter que l'algorithme du KNN doit effectuer à chaque fois une recherche dans la base de données, des K plus proches voisins des signatures RSS, avant d'estimer la position du mobile.

3.2.2.2 Réseaux de neurones artificiels

Une autre approche examinée pour estimer la position du mobile est l'approche des réseaux de neurones artificiels (ANN : "Artificial Neural Network") [11, 26, 27, 28]. Cette approche va permettre de modéliser le comportement non-linéaire entre les signatures RSS et la position du mobile, en se basant sur une structure élémentaire qui est le neurone artificiel, dont la conception est inspirée du fonctionnement du neurone biologique. Le modèle d'un réseau de neurones est entièrement défini par trois caractéristiques distinctes : une règle de transmission, une architecture et une règle d'apprentissage [29].

La règle de transmission, qui reflète la structure de base du neurone (voir figure 3.3), détermine le type de fonction qui permet au signal d'entrée de se propager d'un neurone à l'autre. Dans la majorité des modèles, cette règle est composée d'une sommation pondérée, sur laquelle une fonction d'activation (souvent non-linéaire) est appliquée. Il existe de nombreuses formes possibles pour la fonction d'activation. Les fonctions les plus utilisées sont : la sigmoïde, l'identité, le pas unitaire et la fonction linéaire à seuil.

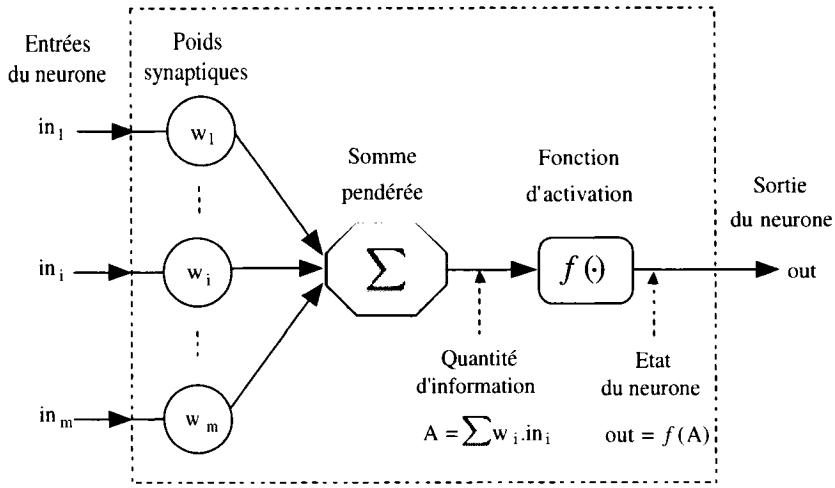


FIG. 3.3 Structure de base d'un neurone artificiel.

Pour ce qui est de l'architecture du réseau de neurones, elle permet de représenter schématiquement les neurones et de définir la façon dont ils sont reliés ou regroupés entre eux. Elle peut avoir une ou plusieurs couches de neurones, et elle peut être à circulation de l'information vers l'avant, soit non-récurrente (réseaux "Feed-forward"), ou récurrente (réseaux "Feed-back").

Une autre propriété, qui est vraisemblablement la plus intéressante des réseaux de neurones, est l'apprentissage. L'apprentissage est une phase de développement d'un réseau de neurones durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré.

Dans notre travail, deux structures de réseaux de neurones sont examinées : le réseau de neurones multicouches (MLP : "Multilayer Perceptron") et le réseau de neurones à régression généralisée (GRNN : "Generalized Regression Neural Network"). Ces deux types de réseaux de neurones utilisent une architecture "feed-forward" et un apprentissage supervisé, où l'on minimise l'erreur de sortie par rapport à une sortie désirée.

• **Réseau de neurones de type MLP :**

Le réseau de neurones de type MLP, ou appelé aussi réseau de neurones à rétro-propagation du gradient de l'erreur ("back-propagation")[28], est basé sur une architecture non-récurrente "feed-forward", et sur un apprentissage supervisé. Dans l'architecture du réseau MLP, les neurones sont arrangés par couche. Il n'y a pas de connexions entre les neurones d'une même couche et les connexions ne se font qu'avec les neurones des couches avales (voir figure 3.4).

Projetons maintenant, l'approche du réseau MLP sur le problème de la localisation par fingerprinting. Soit $RSS = (RSS_1, RSS_2, .., RSS_m)$, le vecteur des signatures RSS, représentant les entrées du réseau MLP, où m correspond au nombre de points d'accès qui sont à l'écoute, et soit (x, y) la position du mobile (en 2D), représentant les sorties du réseau de neurones. L'architecture du réseau MLP est donnée par la figure 3.4.

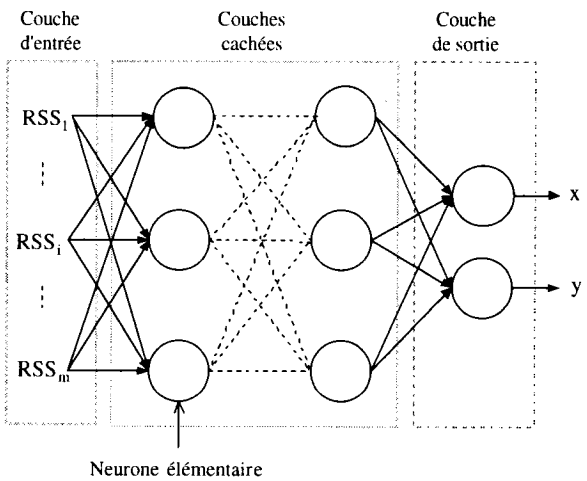


FIG. 3.4 Architecture multicouche d'un réseau de neurones de type MLP.

Dans le cas d'un réseau de neurones de type MLP, et pour des raisons de convergence, souvent on choisit la fonction tangente-sigmoïde (ou tangente-hyperbolique)

comme fonction d'activation pour les couches cachées, et la fonction linéaire pour la couche de sortie. La fonction tangente-sigmoïde est définie par :

$$f(a) = \frac{\exp(a) - \exp(-a)}{\exp(a) + \exp(-a)} \tag{3.4}$$

La règle d'apprentissage du MLP est basée sur l'algorithme de la rétro-propagation du gradient de l'erreur ("back-propagation"), qui minimise l'erreur quadratique moyenne entre les états désirés (x_d, y_d) et les états des sorties de neurones (x, y), pour chaque exemple d'apprentissage, définie comme suit :

$$E = \frac{1}{2}[(x - x_d)^2 + (y - y_d)^2] \tag{3.5}$$

Les étapes d'apprentissage de l'algorithme MLP à rétro-propagation du gradient de l'erreur sont résumées comme suit [28] :

1. Initialiser les poids de connexion w_{ij} de chaque couche du réseau de neurones à des valeurs aléatoires.
2. Initialiser un accumulateur d'erreur totale à zéro, E_T=0.
3. Présenter un vecteur d'entrées RSS et calculer les sorties locales de tous les neurones du réseau MLP.
4. Spécifier la sortie désirée (x_d, y_d) et évaluer les erreurs locales pour toutes les couches.
5. Ajuster les poids de connexion en multipliant l'erreur locale de la couche à la sortie du neurone qui lui correspond.
6. Calculer l'erreur quadratique : $E_T = E_T + \frac{1}{2}[(x - x_d)^2 + (y - y_d)^2]$.
7. Répéter les étapes 3 à 6 pour chaque exemple d'apprentissage.

8. Calculer l'erreur quadratique moyenne : $E_{moy} = E_T/N_p$, où N_p représente le nombre d'exemples d'apprentissage.
9. Si $E_{moy} = E_{moy}$(désirée), aller en 10 sinon en 2.
10. Sauvegarder les poids synaptiques du réseau MLP et mettre fin.

Les étapes de 2 jusqu'à 8 constituent une époque d'apprentissage.

Une fois l'étape d'apprentissage terminée, la structure du réseau MLP, qui contient les poids de connexion ajustés des neurones, est mémorisée pour l'utiliser par la suite lors de l'estimation (on-line) de la position du mobile. Cette étape consiste en une étape de généralisation, où le réseau MLP doit déterminer la position actuelle et inconnue du mobile à partir du vecteur actuel des signatures RSS. Ces signatures peuvent être contenues ou non dans la base de données utilisée dans l'étape d'apprentissage.

• Réseaux de neurones de type GRNN :

Le réseau de neurones à regression généralisée (GRNN : "Generalized Regression Neural Network") a une architecture assez semblable à celle des réseaux de neurones à base radiale (RBF : "Radial Basis Networks") [30]. Comme dans le réseau MLP, le réseau GRNN utilise un apprentissage de type supérivisé, une architecture multicouche et des grandeurs d'entrées déterministes. Toutefois, il possède seulement deux couches de neurones (voire figure 3.5). Les neurones de la couche cachée ont une fonction d'activation à base radiale (forme gaussienne), et ceux de la couche de sortie ont une fonction d'activation linéaire. Le nombre de neurones sur la couche cachée est égale à la dimension du vecteur d'entrée RSS. La fonction d'activation de type RBF est définie comme suit :

$$f(a) = \exp(-\frac{\|a - w\|^2}{2\sigma^2}) \tag{3.6}$$

où $\|\cdot\|$ désigne la norme Euclienne, a réprésente le vecteur entrée du neurone, w représente le vecteur de poids synaptiques de la couche radiale, et σ représente l'écart type ou souvent appelé le "spread", qui définit le seul paramètre du réseau GRNN que l'on fixe a priori.

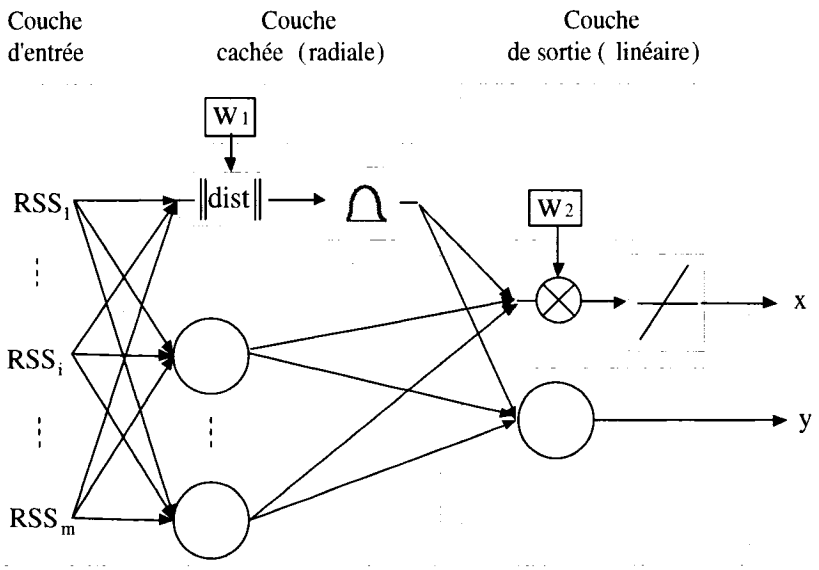


FIG. 3.5 Architecture du réseau GRNN.

Tout comme dans le cas du réseau MLP, le réseau GRNN doit être entraîné avant d'être utilisé. Comme sur la couche cachée il y a autant de neurones que de signatures, le réseau GRNN assigne un neurone à chaque signature. Ce neurone possède une zone d'influence autour de la signature correspondante, dont le rayon est fonction du "spread". Plus le "spread" est faible, plus le neurone mémorise bien la signature qui lui a été assignée. Le réseau GRNN donne alors une excellente précision lorsque la signature à l'entrée est l'une des signatures apprises (bonne capacité de mémorisation). Mais, plus le réseau GRNN mémorise bien, moins il généralise, c'est-à-dire qu'il donne une faible précision quand la signature à l'entrée n'a pas été apprise. Un compromis doit donc être fait (off-line), pour bien ajuster la valeur du paramètre "spread".

Pour faire l'apprentissage du réseau GRNN, plusieurs stratégies se présentent. La plus simple consiste à optimiser simultanément tous les paramètres du réseau en utilisant la rétropropagation du gradient de l'erreur, comme fût le cas avec le réseau MLP. Et comme l'architecture du réseau GRNN est plus simple que celle du MLP, il s'ensuit qu'il y aura moins de traitement, et donc un temps d'apprentissage beaucoup moins long pour le réseau GRNN.

3.3 Description expérimentale du système de localisation proposé

Dans cette section, la mise en expérimentation du système de localisation proposé est exposée. L'environnement de localisation, le matériel et les logiciels utilisés sont présentés. Il est à noter que les étapes de mesure (off-line) et d'enregistrement de la base de données des signatures RSS, pour différentes positions de l'usager mobile, ont été réalisées par un étudiant, dans le cadre de son projet de maîtrise d'ingénierie [31]. Par contre, pour ce qui est de l'étape de mesures pour l'estimation de la position du mobile en temps réel (étape de test), elle a été effectuée dans le cadre de ce présent travail. La figure 3.6 illustre le processus expérimental du système de localisation proposé.

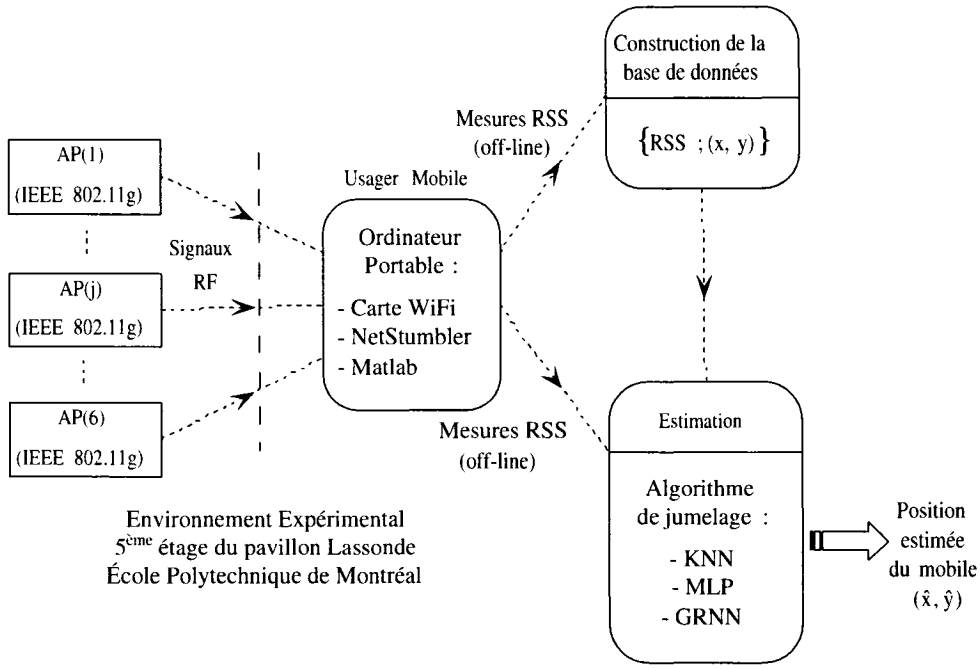


FIG. 3.6 Processus expérimental du système de localisation proposé.

3.3.1 Environnement interne d'expérimentation

L'environnement interne de localisation ou d'expérimentation a été considéré au $5^{ème}$ étage du pavillon Mckay-Lassonde, situé à l'École Polytechnique de Montréal (Québec, Canada). Les mesures expérimentales ont été effectuées dans différents couloirs et salles. Un plan du $5^{ème}$ étage du pavillon Mckay-Lassonde est montré sur la figure 3.7, comportant un espace d'application de forme trapezoidal et ayant pour dimensions 95 m, 70 m et 40 m. La figure 3.8 montre des photos du même étage, illustrant la structure en 3D de ses couloirs.

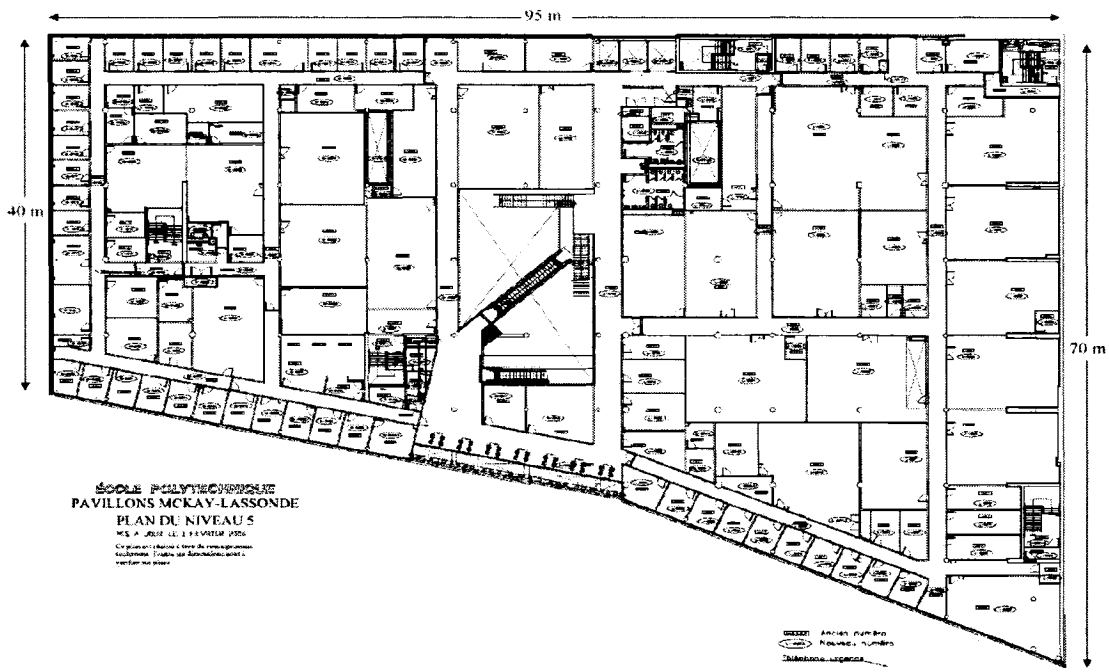


FIG. 3.7 Plan du $5^{ème}$ étage du pavillon Mckay-Lassonde, situé à l'École Polytechnique de Montréal.

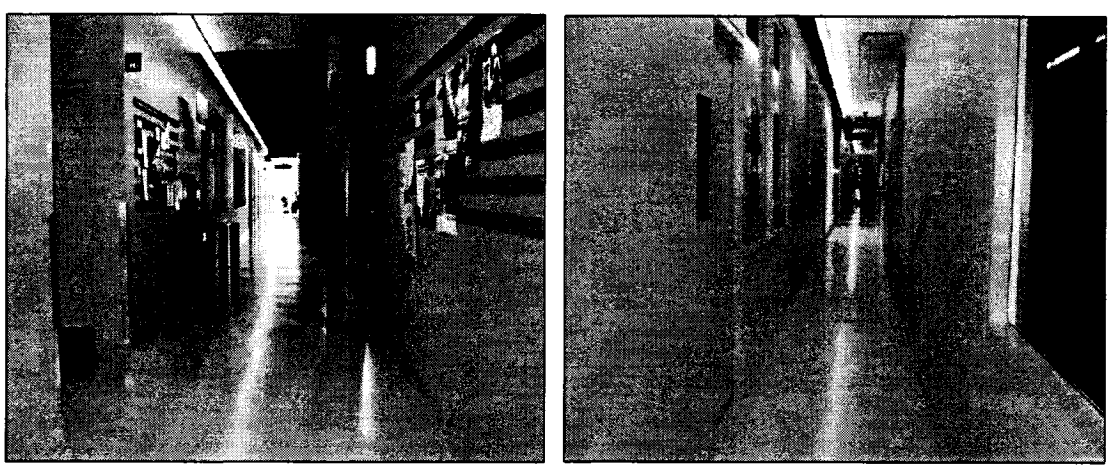


FIG. 3.8 Photos de deux couloirs du $5^{ème}$ étage du pavillon Mckay-Lassonde, situé à l'École Polytechnique de Montréal

.

3.3.2 Matériel utilisé

Dans cette section, le matériel expérimental utilisé pour effectuer les différentes mesures est présenté ainsi que certaines de ses caractéristiques.

3.3.2.1 Réseaux WLAN

L'environnement interne d'expérimentation, considéré au $5^{ème}$ étage du pavillon Mckay-Lassonde, est équipé par un réseau WLAN (HERMES) de norme IEEE 802.11g. La norme 802.11g offre un haut débit (54 Mbps théorique) sur la bande de fréquence de 2.4 GHz. La table 3.1, fournie par le Service Informatique de l'École Polytechnique de Montréal, illustre les différentes caractéristiques du réseau sans fil HERMES. Le nombre total de points d'accès de ce réseau, permettant de couvrir tout le pavillon Mckay-Lassonde, est de 48. Toutefois, comme nous avons considéré qu'un seul étage ($5^{ème}$ étage) pour notre expérimentation, seuls les six points d'accès qui permettent de mieux couvrir cet étage ont été considérés.

Norme	802.11g
Canaux utilisés	1, 6 et 11
Puissance transmise CCK	de 1 mW à 100 mW
Puissance transmise OFDM	de 1 mW à 30 mW
Débits de données (Mbps)	5.5, 6, 9, 11, 12, 18, 24, 36, 40

TAB. 3.1 Caractéristiques techniques du réseau sans fil HERMES.

3.3.2.2 Usager mobile

Dans notre processus expérimental, le mobile consiste en un piéton équipé par un ordinateur portable de type Dell-Centrino, muni d'un système d'exploitation Windows XP. Le portable est équipé par une carte réseau sans fil de type

"Intel®PPO/Wireless3945ABG", ayant une sensibilité de -94 dBm. Cette carte joue le rôle de récepteur, qui permet de détecter les signaux émis par les différents points d'accès du réseau HERMES. Elle permet d'identifier chaque AP en se basant sur son adresse MAC.

3.3.2.3 L'outil NetStumbler

Pour mesurer le niveau du signal RSS, reçu à partir des différents APs du réseau WLAN, le logiciel NetStumbler (ou Network Stumbler) [32] a été utilisé. Cet outil est d'usage gratuit, travaillant sous Windows, et il est mieux adapté à la carte sans fil utilisée. Il permet d'obtenir une multitude d'informations sur les signaux émis par le réseau WLAN, comme par exemple, le niveau du signal RSS, l'intensité du bruit, le SNR (Signal-to-Noise Ratio), le canal utilisé, l'adresse MAC du point d'accès émettant, etc. Il est également possible d'obtenir les graphiques du SNR en fonction du temps. La figure 3.9 montre une fenêtre de détection par l'outil NetStumbler, des six points d'accès du réseau HERMES considérés, avec leurs adresses MAC, les trois canaux assignés au réseau sans fil, et le graphique du SNR en fonction du temps pour l'un des six points d'accès considérés.

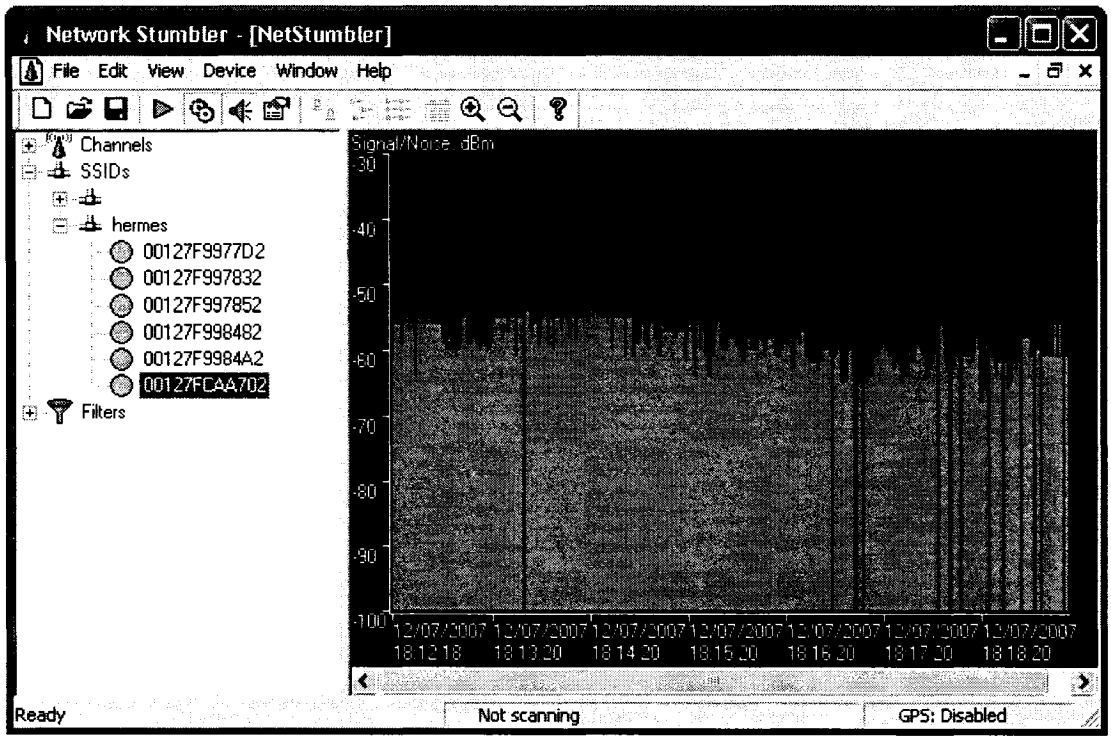


FIG. 3.9 Fenêtre de détection par l'outil NetStumbler, des six points d'accès considérés dans l'expérimentation.

3.3.2.4 Outil d'implémentation

Dans ce travail, l'outil Matlab a été installé sur l'ordinateur portable pour faire les différents traitements de mesures, dont le calcul des moyennes de signatures RSS, l'enregistrement de la base de données contenant les mesures RSS avec les positions de localisation qui leurs correspondent, ainsi que la programmation des trois algorithmes de la technique de fingerprinting, utilisés pour estimer la localisation du mobile.

3.3.3 Procédure de collecte des mesures

La collecte des mesures a été effectuée en plusieurs points des couloirs et des salles du $5^{ème}$ étage du pavillon Mckay-Lassonde. Les étapes de collecte des mesures et d'enregistrement de la base de données de localisation sont données comme suit :

- En chaque point de mesure, un fichier texte est enregistré sous l'outil NetStumbler, contenant les informations sur le niveau du signal de puissance reçu (RSS en dBm), le SNR, le niveau du bruit, le canal utilisé et l'adresse MAC pour chacun des six points d'accès détectés par la carte WiFi.

- Dans le cas de la construction de la base de données (étape d'entraînement) et comme le niveau du signal fluctue au cours du temps, un enregistrement de 60 échantillons a été pris à chaque point de mesure pour chacun des six points d'accès considérés. Un temps d'une demi-seconde séparant deux échantillons de mesures a été fixé. Pour ce qui est des mesures de test (estimation de la position du mobile en temps réel), un seul échantillon RSS a été pris en chaque point de mesure pour que le temps de mesure (qui est de 0.5 seconde) ne dépasse pas le temps d'échantillonnage (temps entre deux position de localisation et qui est pris dans ce présent travail de 2 secondes). Il est à noter que les différentes mesures ont été recueillies en considérant une seule direction (orientation) du fait qu'après plusieurs tests, il a été constaté qu'il n'y a pas d'effet sur le résultat final de localisation.

- Par la suite, le fichier enregistré est traité sur Matlab (voir le programme en Annexe-I.1), pour extraire en chaque point de mesure le niveau de puissance RSS pour chacun des 6 APs, où ils sont différentiés par leurs adresses MAC.

- La valeur RSS qui est prise en compte est la moyenne des 40 échantillons RSS, tel que les dix premières et dernières mesures RSS ne sont pas prises en compte dans le calcul de la moyenne, dans le but de minimiser l'effet de l'usager sur les variations

du signal reçu.

- La valeur "149" est retranchée à la valeur moyenne du niveau du signal RSS, pour retrouver sa valeur réelle en dbm [5].

- Dans le cas où l'un des six points d'accès n'est pas détecté par la carte réseau WiFi, et comme la sensibilité de cette dernière est de -94 dBm, une valeur de -100 dBm est assignée comme valeur RSS au point d'accès non detecté.

À partir de là, la base de données est construite par les différents vecteurs moyens RSS, mesurés en chaque point de position (x, y). Dans notre processus expérimental, le nombre de points de mesures enregistrés dans la base de données, qui va être utilisée lors de l'étape d'entraînement, est de 555 points. Pour ce qui est de l'étape de test (estimation de la position du mobile en temps réel), le nombre de points de mesures enregistrés est de 92 points, recueillis suivant une trajectoire bien définie (de forme rectangulaire). Dans les deux cas, les points de mesures sont séparés par un espacement de 1 mètre.

3.3.4 Implémentation des algorithmes de localisation

Comme cité dans les sections précédentes, trois algorithmes de jumelage de la technique de fingerprinting (KNN, MLP et GRNN) ont été examinés. La table 3.2 définit l'architecture et les paramètres de l'implémentation de chacun d'eux sous Matlab (voir en Annexe I leurs programmes sous Matlab). Il est à noter que les paramètres "K" (pour le KNN), "Spread" (pour le GRNN), et le nombre de neurones pour les trois couches du réseau MLP, ont été choisis comme tels, car après avoir effectué plusieurs tests d'estimation sur les données d'entraînement (off-line), ils ont donnés les meilleurs résultats de localisation.

Algorithme	Architecture et Paramètres
KNN	- Nombre d'entrées RSS (dimension du vecteur RSS) : 6. - Nombre de sorties (position (x, y)) : 2. - Nombre de plus proches voisins : K = 6.
MLP	- Architecture : "feed-forward", définie par la fonction "newff". - Paramètres du réseau neuronal : 6 entrées RSS, 2 couches cachées de 10 et 5 neurones respectivement, et une couche de sortie de 2 neurones qui donne en sortie la position (x, y). - Fonction d'activation : tangente-sigmoïde ("tansig") pour les deux couches cachées, et la fonction linéaire pour la couche de sortie ("purelin"). - Nombre d'itérations pour la phase d'entraînement : 5000.
GRNN	- Architecture définie par la fonction "newgrnn". - Réseau neuronal : 6 entrées RSS, une couche cachée à base radiale (radial basis) de fonction d'activation gaussienne et une couche de sortie de 2 neurones avec des fonctions d'activation linéaires. - Spread : 8.

TAB. 3.2 Paramètres et architectures des algorithmes KNN, MLP et GRNN.

3.4 Résultats et analyse des performances de localisation

Dans cette section, les résultats expérimentaux sont exposés, en donnant les courbes de variation du signal de puissance RSS pour les six points d'accès considérés, ainsi que les performances de localisation en termes d'erreur (distance Euclidienne) moyenne, d'erreur médiane et d'erreur en % CDF ("Cumulatif Density Function"), pour les trois algorithmes considérés (KNN, MLP et GRNN). De même, une analyse des performances suivant le nombre de points d'accès considérés est présentée.

3.4.1 Fluctuations du signal de puissance RSS

La figure 3.10 montre les fluctuations du signal de puissance reçu (RSS) au cours du temps pour les six points d'accès considérés. À partir de cette figure, on constate que pour une même position de localisation, le niveau du signal de puissance RSS fluctue au cours du temps (jusqu'à environ 26 dBm de variation), qui est due aux caractéristiques radio de l'environnement interne de localisation (multitrajet et l'état dynamique du canal de propagation). Ceci justifie le fait de prendre comme valeur RSS, la moyenne des différents échantillons relevés dans une même position, où chaque échantillon est pris à chaque demi-seconde.

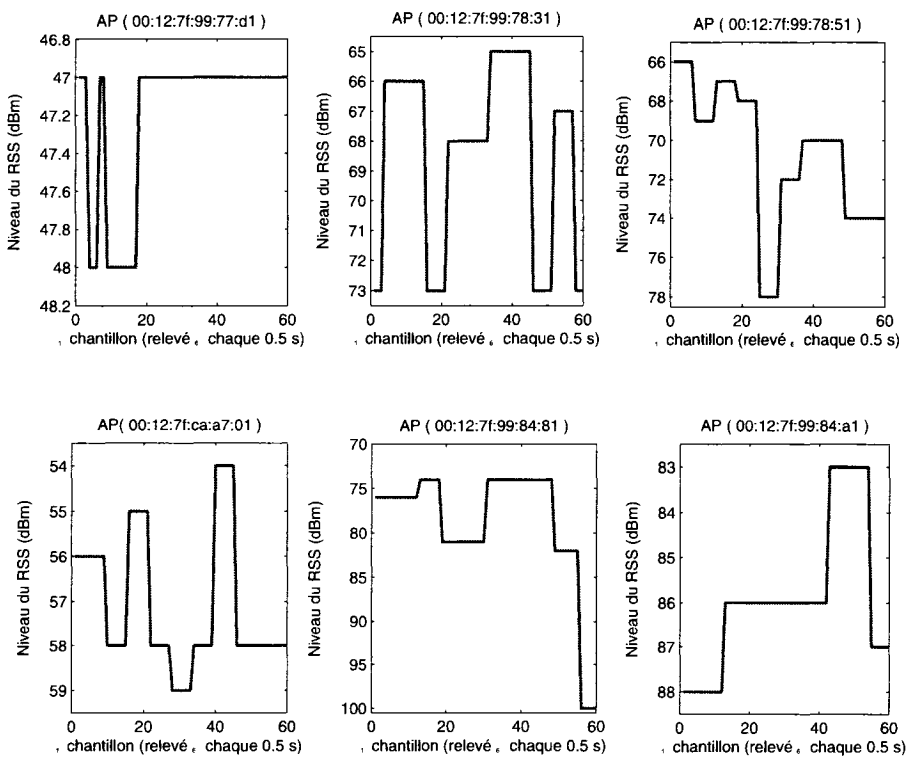


FIG. 3.10 Fluctuations du niveau du signal RSS au cours du temps pour une même position de localisation, pour les six points d'accès considérés.

3.4.2 Performances de localisation

Les figures 3.11 et 3.12 montrent respectivement les distributions cumulées (CDF) des erreurs de localisation du mobile en phase d'appel (on-line), ainsi que les courbes de l'erreur de localisation instantanée, pour les trois algorithmes considérés (KNN, MLP et GRNN). Ici, l'erreur de localisation est définie comme la distance Euclidienne entre les positions (x, y) estimée et réelle :

$$\text{erreur} = \sqrt{(\text{x}_{\text{estimé}} - \text{x}_{\text{réel}})^2 + (\text{y}_{\text{estimé}} - \text{y}_{\text{réel}})^2} \tag{3.7}$$

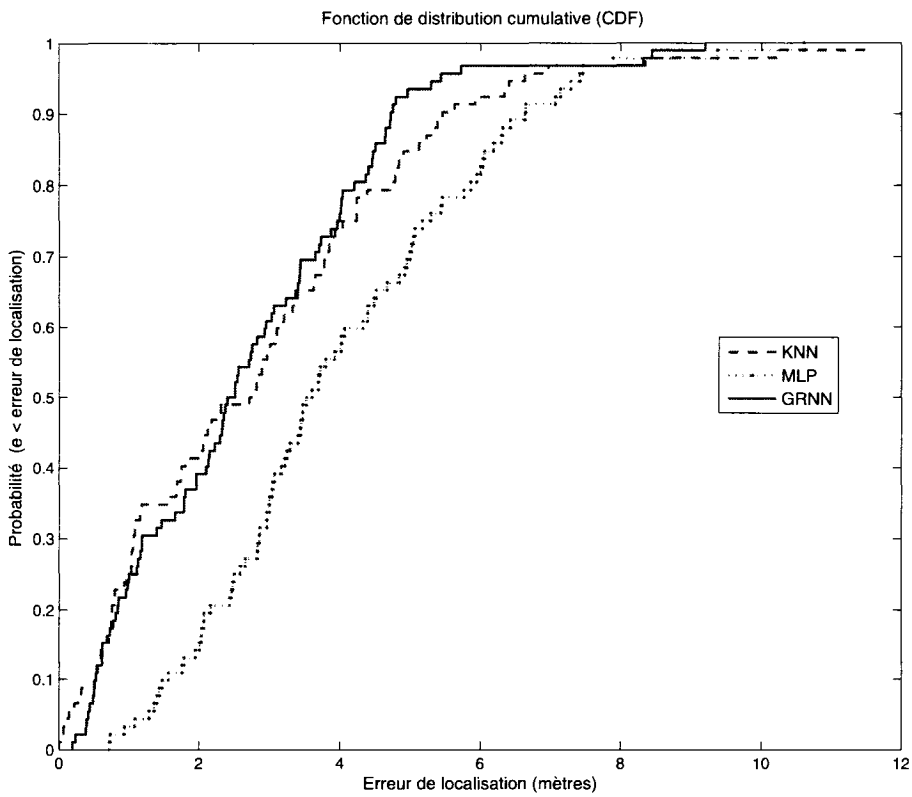


FIG. 3.11 Courbes des distributions cumulées de l'erreur de localisation pour les algorithmes KNN, MLP et GRNN.

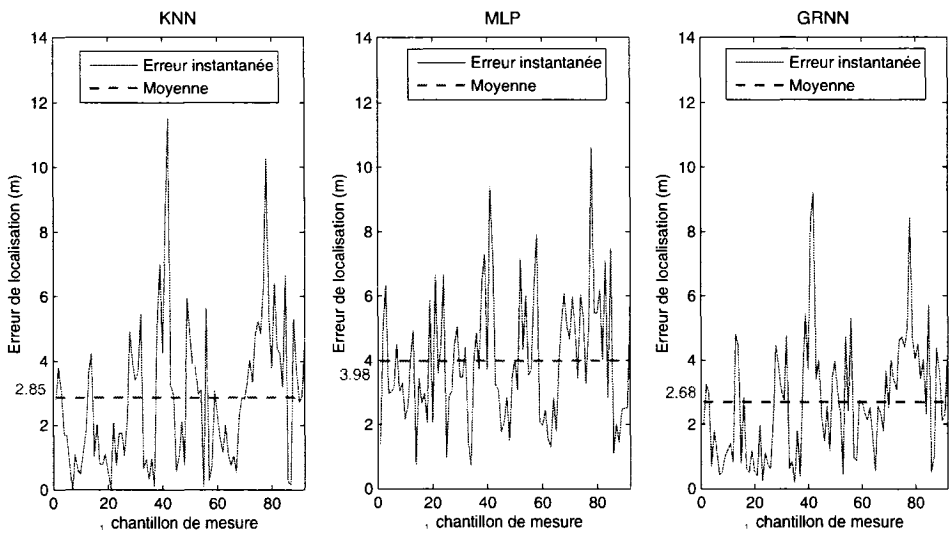


FIG. 3.12 Courbes des erreurs de localisation instantanées pour les algorithmes KNN, MLP et GRNN.

À partir des figures 3.11 et 3.12, on constate que si on fait une analyse sur l'ensemble des positions de localisation estimées, le réseau GRNN donne une meilleure courbe de distribution cumulée de l'erreur de localisation que la méthode KNN et le réseau MLP. De même, les maximas d'erreur de localisation enregistrés par le réseau GRNN sont inférieurs à ceux enrigistrés par la méthode KNN et le réseau MLP.

La table 3.3 donne un résumé des informations contenues dans les figures 3.11 et 3.12, en donnant une comparaison entre les trois algorithmes considérés eux-mêmes, et d'un autre coté entre ces derniers et la méthode RADAR [1], en termes d'erreur (distance Euclidienne) moyenne, d'erreur médiane (erreur obtenue pour 50% des données estimées) et d'erreurs obtenues pour 25%, 75% et 95% des données estimées. Les résultats de la table montrent que le réseau GRNN donne relativement une meilleure précision de localisation par rapport au réseau MLP et à la méthode KNN. D'un autre côté, le réseau MLP et la méthode KNN donnent des erreurs de

localisation inférieures à celle obtenue par la méthode RADAR [1]. Pour ce qui est du réseau MLP, on remarque que ses performances sont relativement dégradées par rapport aux autres algorithmes, qui peuvent être dues à la non-unicité de certaines signatures RSS par rapport aux positions de localisation mesurées.

Algorithme	Erreur de localisation en termes de distance Euclidienne (m)				
	Moyenne	Médiane	CDF à 25%	CDF à 75%	CDF à 95%
KNN	2.85	2.76	1.00	3.81	6.61
MLP	3.98	3.56	2.54	4.99	7.41
GRNN	2.68	2.45	1.05	3.62	5.42
RADAR [1]	3.88	2.94	1.92	4.69	/

TAB. 3.3 Comparaison des performances de localisation entre les algorithmes KNN, MLP et GRNN et la méthode RADAR [1].

3.4.3 Effet du nombre de points d'accès sur la précision de localisation

La table 3.4 montre l'effet du nombre de points d'accès déployés dans l'environnement de localisation sur les performances de localisation en termes d'erreur moyenne.

Algorithme	Erreur de localisation moyenne (m)		
	4 Points d'Accès	5 Points d'Accès	6 Points d'Accès
KNN	4.57	3.50	2.85
MLP	5.92	4.19	3.98
GRNN	4.58	3.04	2.68

TAB. 3.4 Effet du nombre de points d'accès sur les performances de localisation

À partir de la table 3.4, on constate qu'en augmentant le nombre de points d'accès, les performances de localisation s'améliorent. Ce résultat montre l'avantage de la technique utilisée (fingerprinting) par le système de localisation proposé. En effet,

lorsqu'on augmente le nombre des points d'accès, les signatures deviennent plus riches en information et permettent ainsi d'éviter le cas de non-unicité.

3.5 Conclusion

Dans ce chapitre, un système de localisation de fingerprinting a été modélisé est mis en expérimentation dans un environnement interne équipé par un réseau WLAN. Le processus de localisation de ce système a été opéré en deux phases. Une phase de collecte de données (enregistrement d'une base de données de signatures RSS recueillies en différentes positions du mobile) et une phase d'estimation de la position actuelle du mobile. Dans cette dernière phase, trois algorithmes de jumelage (KNN, MLP et GRNN) ont été examinés. Les résultats expérimentaux obtenus ont montré que le réseau GRNN donne relativement les meilleurs performances de localisation par rapport au réseau GRNN et à la méthode KNN. Les précisons de localisation obtenues, en terme d'erreur moyenne, étaient de 2.68 m pour le GRNN, 2.85 m pour le KNN et 3.98 m pour le MLP. D'un autre côté, l'effet d'augmenter le nombre de points déployés sur l'amélioration de la précision de localisation a été analysé.

Toutefois, et comme rapporté dans la sous-section (3.4.1) et illustré par la figure 3.10, le niveau du signal RSS fluctue au cours du temps conduisant à une dégradation des performances de localisation. De ce fait, le chapitre suivant a pour but de proposer une approche, basée sur le filtrage de Kalman et de ses extensions, pour atténuer les erreurs de localisation causées par ces différentes fluctuations du niveau du signal RSS.

CHAPITRE 4

AMÉLIORATION DES PERFORMANCES DU SYSTÈME DE LOCALISATION DE FINGERPRINTING EN UTILISANT LE FILTRAGE DE KALMAN ET DE SES EXTENSIONS

4.1 Introduction

Dans le chapitre précédent, un système de localisation dans un environnement interne, basé sur la technologie WLAN et la technique de fingerprinting, a été proposé. Toutefois, les résultats obtenus ne sont pas assez satisfaisants en termes de précision de localisation et restent ainsi assujettis à des améliorations. En effet, comme illustré dans le chapitre précédent, le niveau du signal RSS fluctue au cours du temps, ce qui introduit des erreurs de localisation. De même, l'approche de fingerprinting, utilisée dans le système de localisation proposé, estime la position de localisation du mobile d'une manière statique, c'est-à-dire elle ne tient pas compte du mouvement de ce dernier, ce qui génère ainsi des discontinuités au niveau de la trajectoire (suivi) du mobile. Pour atténuer l'effet des fluctuations du signal RSS sur la précision de localisation et avoir un bon suivi du mobile, du filtrage doit être appliqué.

Le filtrage de Kalman [33], appliqué plus particulièrement dans le domaine de l'automatique et de la robotique, semble être le mieux adapté aux problèmes de localisation cités ci-dessus. Il propose une approche d'estimation dynamique qui tient compte de la mesure, de l'historique et des caractéristiques du mouvement du mobile. De ce fait, toute information de mesure (observation) liée au comportement du mobile sera utile pour améliorer l'estimation de son état. Toutefois, l'application

du filtrage de Kalman exige des hypothèses fortes (système linéaire et bruits blancs gaussiens). Dans le cas contraire, cette application va s'avérer non concluante. Pour cela, des extensions existent pour les cas non couverts par le filtrage de Kalman.

L'objectif de ce chapitre est d'appliquer le filtrage de Kalman et de ses extensions au système de localisation proposé dans le chapitre précédent, pour améliorer les performances de localisation et de suivi de ce dernier, où trois variantes d'amélioration, relatives au modèle de filtrage caractérisant la dynamique du mobile, sont proposées. Dans ce chapitre, le filtre de Kalman et deux de ses extensions (le filtre de Kalman étendu et le filtre de particules) sont présentés et adaptés au cas de la localisation et du suivi de mobiles. Les résultats expérimentaux obtenus avant et après l'application du filtrage sont exposés et analysés.

4.2 Filtrage de Kalman et de ses extensions

Dans cette section, le principe de fonctionnement du filtrage de Kalman est présenté, où le modèle du système est linéaire et les bruits sont supposés blancs gaussiens. Dans le cas où le système est considéré non-linéaire et/ou les bruits ne sont pas blancs gaussiens, deux extensions du filtrage de Kalman sont présentées : le filtre de Kalman étendu et le filtre de particules.

4.2.1 Filtrage de Kalman

Depuis que R.E. Kalman publia son fameux article en 1960 [33], décrivant une solution récursive pour le problème de filtrage linéaire à temps discret, le filtre de Kalman a eu un grand intérêt dans beaucoup de domaines de recherches et d'applications. Ces dernières sont nombreuses [34], comme par exemple, estimer des conditions initiales inconnues (balistique), prédire des trajectoires de mobiles,

localiser et faire le suivi d'un engin (navigation, radar, etc.), ou également implanter des lois de commande.

4.2.1.1 Description du filtrage de Kalman

Le principe de fonctionnement du filtrage de Kalman est illustré par la figure 4.1 (développée à partir de [35]).

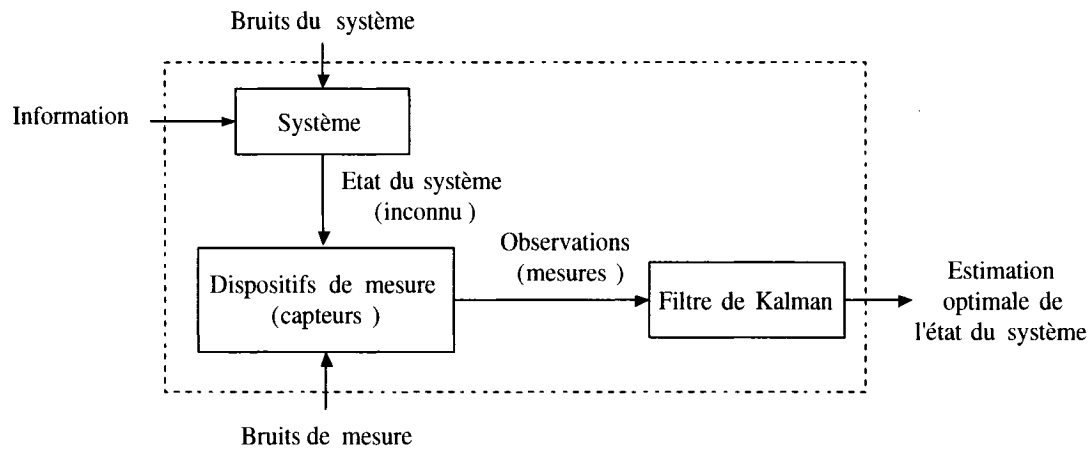


FIG. 4.1 Principe de fonctionnement du filtrage de Kalman.

Ce type de filtrage vise à estimer d'une manière optimale l'état du système, en tenant compte de la connaissance des lois d'évolution du système, des mesures observées, des propriétés statistiques des bruits et de toutes les informations sur les conditions initiales des paramètres du système. Dans un environnement stochastique, lorsque les bruits sont blancs gaussiens et que ses variances sont connues, le filtre de Kalman est optimal, c'est-à-dire que, de tous les filtres, c'est celui dont la variance de l'erreur d'estimation est la plus faible.

Dans le problème du filtrage de Kalman (en temps discret), on considère un vecteur $x_k \in \mathbb{R}^n$ représentant l'état du système non observé, dont les caractéristiques

statistiques sont supposées connues. On détermine une loi d'évolution (modèle) de cet état sous-forme d'une relation de récurrence. Cette loi tient compte de l'état du système à l'instant $(k-1)$, pour déterminer l'état courant du système à l'instant k, tout en tenant compte d'une mesure disponible à ce même instant ($z_k \in \mathbb{R}^m$). Ainsi, le système est modélisé par une équation d'état, déterminant l'état actuel du système en fonction de l'état précédent connu, qui est donnée par :

$$x_k = F.x_{k-1} + w_k \tag{4.1}$$

et d'un autre côté, le système est équipé d'observateurs (capteurs) permettant d'obtenir des informations réelles (mesures) quant à l'état réel du système. Cette relation entre la mesure et l'état du système se modélise par l'équation de mesure (ou d'observation) suivante :

$$z_k = H.x_k + v_k \tag{4.2}$$

où $F \in \mathbb{R}^{n\times n}$ et $H \in \mathbb{R}^{n\times m}$ sont respectivement les matrices d'état et d'observation, de type déterministe et connues a priori, où n est la dimension du vecteur d'état du système x_k, et m est la dimension du vecteur d'observation z_k. En pratique, les matrices F et H devraient varier à chaque étape de temps k, mais dans notre cas, elles sont considérées constantes. w_k et v_k sont les bruits d'état et de mesure, supposés blancs gaussiens centrés et respectivement, de matrices de covariances $Q \in \mathbb{R}^{n\times n}$ et $R \in \mathbb{R}^{m\times m}$, tel que :

$$Q = \mathrm{cov}(w_i) = \mathrm{E}\{w_i.{w_i}^\top\} \tag{4.3}$$

$$R = \mathrm{cov}(v_i) = \mathrm{E}\{v_i.{v_i}^\top\} \tag{4.4}$$

Les bruits d'état et de mesure sont supposés mutuellement indépendants et indépendants de la condition initiale x_0 (état initial du système), où x_0 est supposée gaussienne, de moyenne nulle et de matrice de covariance P_0.

Le développement du filtrage de Kalman se présente comme suit [33] : on définit x_k^- comme l'estimation *a priori* de l'état du système à l'étape k, fournissant des connaissances a priori sur le système, et $\hat{x}_k$ comme l'estimation *a posteriori* de l'état du système à l'étape k, sachant qu'on a une observation z_k. Ainsi, les erreurs d'estimation a priori et a posteriori, et leurs matrices de covariances sont données respectivement comme suit :

$$e_k^- = x_k^- - x_k \tag{4.5}$$

$$e_k = \widehat{x}_k - x_k \tag{4.6}$$

$$P_k^- = \text{cov}(e_k^-) \tag{4.7}$$

$$P_k = \text{cov}(e_k) \tag{4.8}$$

Afin d'obtenir la solution optimale au problème du filtrage de Kalman (au sens de la minimisation de l'erreur quadratique moyenne), l'erreur d'estimation a posteriori e_k doit être une gaussienne de moyenne nulle et de matrice de covariance P_k, et à l'instant k, le système doit être observable. Pour ce faire, le filtrage de Kalman se base sur la propagation de la densité de probabilité conditionnelle p_k de l'état x_k sachant qu'on a les observations $\{z_1, z_2, ..., z_k\}$, avec comme contraintes un modèle du système linéaire et des bruits appliqués supposés blancs gaussiens. Deux paramètres suffisent pour caractériser la densité $p_k(x_k|z_{1,2,...,k})$: sa moyenne $\hat{x}_k$ et sa matrice de covariance P_k. Ces paramètres sont calculés d'une manière récursive suivant les étapes de prédiction et de correction du filtre de Kalman, qui seront présentées dans ce qui suit.

4.2.1.2 Étapes d'estimation du filtre de Kalman

La figure 4.2 (développée à partir de [36]) illustre les opérations récursives du filtrage de Kalman. Ce dernier fournit l'estimation de l'état du système linéaire en se basant

sur deux phases distinctes : la phase de *prédiction* et la phase de *correction.* La phase de prédiction (ou de mise à jour temporelle) utilise l'état estimé à l'instant précédent $(k-1)$ pour produire une estimation de l'état à l'instant courant k. Dans la phase de correction (ou de mise à jour par les observations), les observations à l'instant courant k sont utilisées pour corriger l'état prédit dans le but d'obtenir une estimation plus précise.

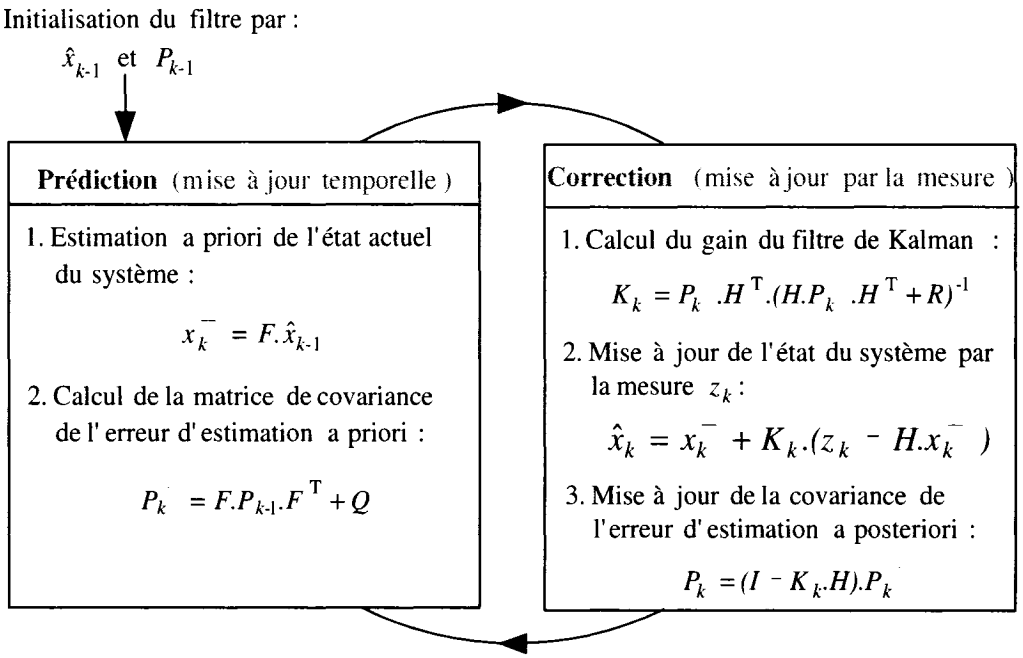


FIG. 4.2 Étapes d'estimation récursives du filtre de Kalman.

Dans la figure 4.2, K_k représente le gain de Kalman minimisant la covariance de l'erreur d'estimation a posteriori (P_k).

Certes, le filtrage de Kalman est un outil très puissant et optimal au sens de la minimisation de l'erreur d'estimation sous les contraintes d'un système linéaire et de bruits gaussien, toutefois, il ne permet pas de couvrir toutes les situations, notamment celles où le système est non-linéaire et/ou les bruits ne sont pas blancs

gaussiens. Plusieurs variantes de ce filtre permettent de répondre aux problèmes d'estimation avec des contraintes non-linéaires et/ou non-gaussiennes.

4.2.2 Filtrage de Kalman étendu

La première solution qui a été proposée pour résoudre le problème du filtrage de Kalman dans le cas d'une dynamique non-linéaire, est le filtre de Kalman étendu (EKF : Extended Kalman Filter) [36, 37, 38]. Cette solution repose sur une linéarisation au premier ordre, par le développement de Taylor, des équations non-linéaires du processus et d'observation en gardant la même structure que le filtre de Kalman classique.

Le processus d'estimation du filtre EKF se présente comme suit : le système est régit par une dynamique non-linéaire avec des bruits blancs gaussiens centrés, mutuellement indépendants et indépendants de la condition initiale de l'état du système. On suppose, comme dans le cas du filtre de Kalman, que $x_k \in \mathbb{R}^n$ est l'état du système à estimer, décrit par l'équation d'état non-linéaire suivante :

$$x_k = f(x_{k-1}) + w_k \tag{4.9}$$

et d'un autre côté, cet état est lié à l'observation (mesure) z_k par l'équation d'observation non-linéaire suivante :

$$z_k = h(x_k) + v_k \tag{4.10}$$

où f et h sont des fonctions non-linéaires modélisant respectivement, la relation entre l'état courant du système à l'instant k et l'état précédent à l'instant $k-1$, et la relation entre l'observation à l'instant courant k et l'état du système à ce même instant. w_k et v_k sont respectivement les bruits d'état et de mesure, définis comme

dans le cas du filtrage de Kalman classique.

Comme les fonctions f et h sont non-linéaires, celles-ci ne peuvent pas être appliquées directement au calcul des covariances de bruits. Pour cela, une matrice des dérivées partielles (la Jacobienne) doit être calculée. À chaque instant k, la Jacobienne est évaluée avec les états estimés courants. Ce processus linéarise essentiellement la fonction non-linéaire autour de l'estimation courante. Sous la condition de dérivabilité des fonctions f et h, on peut alors linéariser ces fonctions autour de l'état estimé a posteriori $\hat{x}_{k-1}$, et autour de l'état estimé a priori x_k^-, où $x_k^- = h(\hat{x}_{k-1})$. Les fonctions f et h sont linéarisées comme suit :

$$f(x_k) \simeq F_k^J.(x_k - \hat{x}_{k-1}) + f(\hat{x}_{k-1}), \text{ avec : } F_k^J = \nabla_{x_{k-1}} f(\hat{x}_{k-1}) \quad (4.11)$$

$$h(x_k) \simeq H_k^J.(x_k - x_k^-) + h(x_k^-), \text{ avec : } H_k^J = \nabla_{x_k} h(x_k^-) \quad (4.12)$$

D'où le modèle du système linéarisé suivant :

$$x_k = F_k^J.(x_k - \hat{x}_{k-1}) + f(\hat{x}_{k-1}) + w_k \quad (4.13)$$

$$z_k = H_k^J.(x_k - x_k^-) + h(x_k^-) + v_k \quad (4.14)$$

où F_k^J et H_k^J sont respectivement, les Jacobiennes (ou les matrices de dérivées partielles) de f et h, définissant les nouvelles matrices d'état et d'observation pour le système linéarisé. Toutefois, ces matrices varient à chaque étape de temps k.

La figure 4.3 (développée à partir de [36]) illustre les étapes d'estimation récursives pour le filtre de Kalman étendu, appliqué au système linéarisé, données par les équations (4.13) et (4.14). De façon similaire au filtre de Kalman classique, le mécanisme d'estimation pour le filtre de Kalman étendu est opéré en deux étapes : une étape de mise à jour temporelle (prédiction), qui donne l'estimation a priori de l'état du système (x_k^-), et une étape de mise à jour par les observations (correction), qui corrige l'estimation a priori de l'état du système en se basant sur la connaissance de l'état observé (z_k), et donnant ainsi l'estimation a posteriori de l'état du système ($\hat{x}_k$). Il est à noter que les matrices Q, R, P_k^- et P_k, et le gain K_k, contenus dans la figure 4.3, sont définis de la même manière que dans le cas du filtre de Kalman, présenté précédemment.

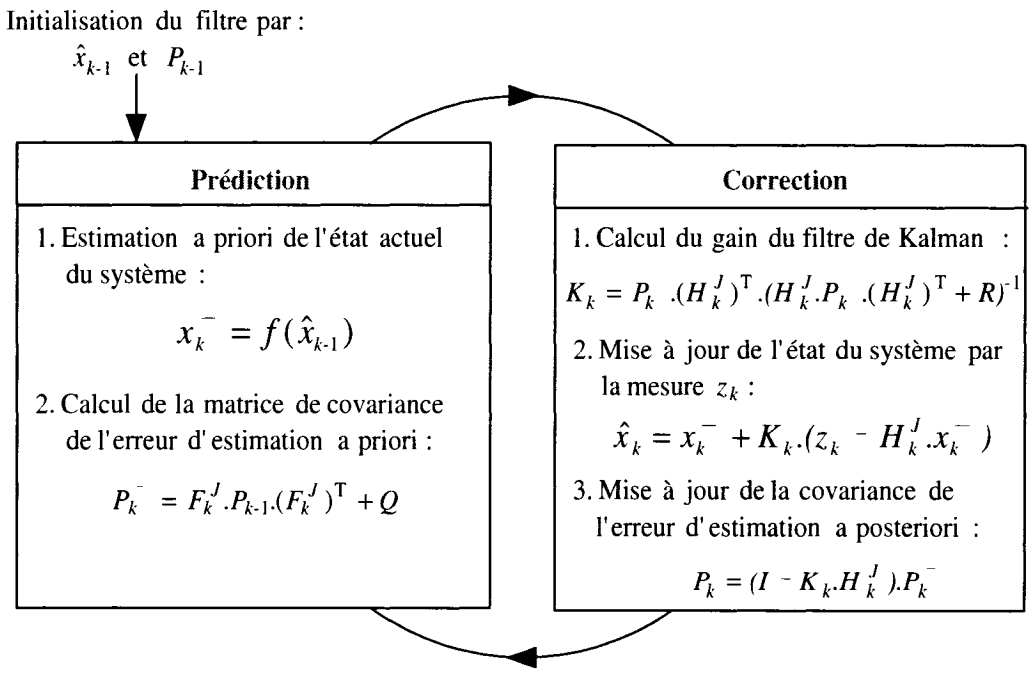


FIG. 4.3 Étapes d'estimation récursives du filtre de Kalman étendu.

4.2.3 Filtrage de particules

Une autre extension du filtrage de Kalman est le filtre de particules, aussi appelé filtre particulaire (PF : "particle filter"), qui permet de s'affranchir des hypothèses nécessaires à l'application du filtre de Kalman (système linéaire et bruits blancs gaussiens). Ce type de filtrage repose sur l'exploitation des approximations données par les méthodes de Monte-Carlo [39, 40, 41]. Ces approximations permettent de modéliser fidèlement les phénomènes alétoires intervenant dans le comportement du système, et présentent un grand intérêt pour la détermination des densités de probabilité conditionnelles, liées à l'état du système.

4.2.3.1 Description du filtrage de particules

On considère un système non-linéaire (en temps discret), modélisant l'équation d'état qui donne l'état courant du système ($x_k \in \mathbb{R}^n$), et l'équation d'observation qui corrige cet état courant à l'aide d'une mesure réelle ($z_k \in \mathbb{R}^m$). Ce système non-linéaire est donné comme suit :

$$x_k = f(x_{k-1}) + w_k \tag{4.15}$$

$$z_k = h(x_k) + v_k \tag{4.16}$$

où w_k et v_k sont les bruits d'état et d'observation. Aucune hypothèse n'est faite a priori sur ces bruits, c'est-à-dire que ces bruits ne sont pas nécessairement blancs gaussiens.

Dans les problèmes d'estimation, on cherche à obtenir la meilleure estimation possible de la densité de probabilité $p(x_k|z_{1,\ldots,k})$. Dans le cadre du filtrage de Kalman, cette densité est toujours supposée gaussienne, et ce ne sont que les paramètres de

cette densité qui sont recherchés (moyenne et matrice de covariance). Dans le cas du filtrage de particules, cette densité n'est pas a priori connue, elle peut prendre n'importe quelle forme. Pour cela, le filtre de particules cherche à obtenir la meilleure estimation possible de cette densité.

Pour ce faire, on suppose que les états $\{x_k, k \in \mathbb{N}\}$ forment une chaîne de Markov, tel que : $x_k|x_{k-1} \sim p(x_k|x_{k-1})$ (probabilité de transition), et ayant comme distribution initiale $p(x_0)$. D'un autre côté, on suppose que les observations $\{z_k, k \in \mathbb{N}\}$ sont indépendantes conditionnellement sous réserve que les états $\{x_k, k \in \mathbb{N}\}$ soient connus. En d'autres termes, chaque observation z_k ne dépend que de l'état x_k, et ayant comme densité de probabilité marginale $p(z_k|x_k)$. Il est à noter que tous les paramètres estimés de $\{x_k, k \in \mathbb{N}\}$ viennent de la distribution a posteriori $p(x_{0,...,k}|z_{1,...,k})$. Afin d'aboutir à une estimation réaliste de cette distribution a posteriori, une séquence d'échantillons (ou particules) aléatoires est générée. Ainsi, on suppose que l'on simule N échantillons (particules) indépendants. Les valeurs espérées vis-à-vis de la distribution de filtrage sont approximées par l'ensemble de ces particules pondérées : $\{(\mathrm{w}_k^{(i)}, x_k^{(i)})$, avec : $i = 1, ..., N\}$. Pour estimer récursivement la distribution a posteriori $p(x_{0,...,k}|z_{1,...,k})$, ainsi que ses différentes propriétés statistiques, le filtrage de particules utilise les approximations données par la méthode de Monte-Carlo séquentielle, aussi appelée méthode d'échantillonnage d'importance séquentiel (SIS : "Sequential Importance Sampling") [39, 40, 41]. Cette méthode se base sur l'approximation de l'espérance relative à la distribution a posteriori, donnée par :

$$\mathbb{E}_{p(x_{0:k}|z_{1:k})}\{g(x_{0:k})\} = \int g(x_{0:k})p(x_{0:k}|z_{1:k})dx_{0:k} \approx \frac{\sum_{i=1}^{N} g(x_{0:k})\mathrm{w}_k^{(i)}}{\sum_{i=1}^{N} \mathrm{w}_k^{(i)}} \qquad (4.17)$$

où g est une fonction d'intérêt intégrable. Comme exemple pour cette fonction d'intérêt, on peut prendre le cas de la moyenne, dans ce cas $g(x_{0:k}) = x_k$, ou alors celui de la covariance, et dans ce cas $g(x_{0:k}) = (x_k - \mathrm{E}\{x_k\})(x_k - \mathrm{E}\{x_k\})^\top$. Ces deux

cas de la fonction d'intérêt g sont remplacés dans l'équation (4.17) pour déterminer l'estimé (moyenne) de l'état du système et la matrice de covariance de l'erreur d'estimation.

Ainsi, l'estimé a posteriori de l'état du système est donné par :

$$\hat{x}_k \approx \sum_{i=1}^{N} \widetilde{\mathrm{w}}_k^{(i)} x_k^{(i)} \tag{4.18}$$

et la matrice de covariance de l'erreur d'estimation a posteriori est donnée par :

$$P_k \approx \sum_{i=1}^{N} \widetilde{\mathrm{w}}_k^{(i)} (x_k^{(i)} - \hat{x}_k)(x_k^{(i)} - \hat{x}_k)^{\top} \tag{4.19}$$

où $\widetilde{\mathrm{w}}_k^{(i)}$ sont les poids normalisés, donnés par :

$$\widetilde{\mathrm{w}}_k^{(i)} = \frac{\mathrm{w}_k^{(i)}}{\sum_{i=1}^{N} \mathrm{w}_k^{(i)}} \tag{4.20}$$

Les pondérations $\mathrm{w}_k^{(i)}$ sont les poids d'importance liés aux particules $x_k^{(i)}$. Elles sont approximés par :

$$\mathrm{w}_k^{(i)} \approx \mathrm{w}_{k-1}^{(i)} \frac{p(z_k|x_k^{(i)}) p(x_k^{(i)}|x_{k-1}^{(i)})}{q(x_k^{(i)}|x_{0:k-1}^{(i)}, z_{1:k})} \tag{4.21}$$

où la fonction q est appellée *distribution d'échantillonnage d'importance.* Son choix conditionne le bon fonctionnement du filtre particulaire [40]. En effet, la méthode d'échantillonnage séquentiel SIS repose sur la génération (ou simulation) de particules à partir de la fonction q, supposée connue a piori, et qui va permettre d'évaluer les poids d'importance $\mathrm{w}_k^{(i)}$. Elle doit être choisie de préférence à minimiser la variance de ces poids et tenir compte de l'observation récente. Généralement, pour des raisons de coût de calcul, on prend comme fonction q, la distribution d'importance

optimale [40], donnée par :

$$q(x_k^{(i)}|x_{0:k-1}^{(i)}, z_{1:k}) = p(x_k^{(i)}|x_{k-1}^{(i)}) \tag{4.22}$$

Il s'en suit que les poids d'importance seront approximés par :

$$\mathrm{w}_k^{(i)} \approx \mathrm{w}_{k-1}^{(i)} p(z_k|x_k^{(i)}) \tag{4.23}$$

Il est à noter que le filtre de particules basé sur la méthode d'échantillonnage SIS est souvent voué à l'échec au bout d'un temps relativement long. En fait, l'augmentation de la dispersion des poids a des effets néfastes sur la qualité de la précision de l'estimation [40], et provoque à long terme la divergence du filtre. En pratique, si la distribution d'importance q n'est pas assez proche de la densité optimale, les poids auront tendance à prendre des valeurs disparates, c'est-à-dire qu'au bout d'un certain temps, seulement quelques particules conservent un poids significatif et toutes les autres ont un poids quasiment nul. Ce phénomène est appelé *dégénérescence des pondérations*, qui provoque la divergence du filtre. Pour éviter une telle situation, une solution consiste à ne sélectionner que les échantillons significatifs de cette estimation de la distribution. Pour cela, une phase de rééchantillonnage est ajoutée à la méthode d'échantillonnage SIS, définie sous le nom de la méthode d'échantillonnage avec rééchantillonnage par importance (SIR : "Sampling Importance Resampling"). Ainsi, le rééchantillonnage va permettre d'éviter le problème de la dégénérescence du filtre de particules, en redistribuant les particules ayant un poids significatif. La performance de l'algorithme peut aussi être affectée par le choix de la méthode de rééchantillonnage appropriée. Le rééchantillonnage systématique proposé par [42] est optimal en termes de variance.

L'algorithme de rééchantillonnage systématique se présente comme suit [42] :

$[\{x_k^{(j)*}, \widetilde{\mathrm{w}}_k^{(j)}, i^{(j)}\}_{j=1}^N] = RESAMPLE(\{x_k^{(i)}, \widetilde{\mathrm{w}}_k^{(i)}\}_{i=1}^N)$

- Initialiser une fonction de distribution cumulée (CDF) : $c_1 = 0$.
- **Pour** $i = 2 : N$,
  - Générer la fonction CDF : $c_i = c_{i-1} + \widetilde{\mathrm{w}}_k^{(i)}$,
- **Fin pour.**
- Tirer aléatoirement un point de départ suivant une loi uniforme : $u_1 \sim \mathbb{U}[0, \frac{1}{N}]$.
- Initialiser la variable i : $i = 1$.
- **Pour** $j = 1 : N$,
  - Calculer : $u_j = u_1 + \frac{1}{N}(j-1)$.
  - **Tant que** $u_j > c_i$,

    $i = i + 1$,
  - **Fin tant que.**
  - Assigner à : $x_k^{(j)*} = x_k^{(i)}$.
  - Assigner à : $\widetilde{\mathrm{w}}_k^{(j)} = \frac{1}{N}$.
  - Assigner à : $i^{(j)} = i$.
- **Fin pour.**
- Fin de l'algorithme.

4.2.3.2 Étapes d'estimation du filtre de particules

Les étapes récapitulatives du déroulement du filtrage de particules de type SIR sont données comme suit :

Initialisation :

Pour $i = 1, ..., N$, tirer aléatoirement les échantillons (particules) $x_0^{(i)}$ suivant la distribution initiale $p(x_0)$.

Prédiction :

Pour $i = 1, ..., N$, estimer a priori les états du système $x_{k|k-1}^{(i)} = f(x_{k-1}^{(i)}) + w_k^{(i)}$, où les $w_k^{(i)}$ sont les échantillons du bruit d'état.

Correction :

Pour $i = 1, ..., N$, évaluer les poids d'importance de chaque particule suivant l'équation (4.23) et normaliser les suivant l'équation (4.20).

Rééchantillonnage :

Faire le rééchantillonnage des particules suivant la procédure décrite précédement : $[\{x_k^{(j)*}, \widetilde{\mathrm{w}}_k^{(j)}, i^{(j)}\}_{j=1}^N] = RESAMPLE(\{x_k^{(i)}, \widetilde{\mathrm{w}}_k^{(i)}\}_{i=1}^N)$.

Les estimés de l'état du système et de sa covariance sont donnés respectivement par les équations (4.18) et (4.19).

4.3 Application du filtrage de Kalman et de ses extensions au cas de la localisation et du suivi

Dans cette section, les filtres de Kalman (KF), de Kalman étendu (EKF) et de particules (PF) sont modélisés et appliqués au cas de la localisation et du suivi dans un environnement interne. Pour cela, il faut définir le modèle de filtrage du système qui caractérise les équations d'état et d'observation décrites dans chacun des trois filtres présentés précédemment. Dans notre travail, trois modèles de filtrage ont été proposés suivant la linéarité ou non du système et suivant le nombre d'observations. La figure 4.4 illustre les trois modèles de filtrage proposés pour l'amélioration de la localisation du mobile.

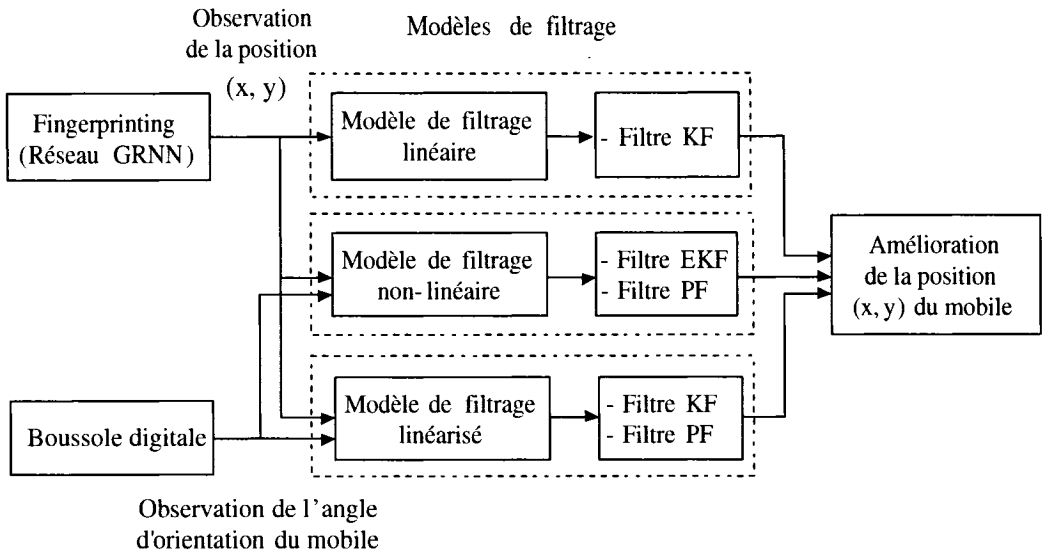


FIG. 4.4 Modèles de filtrage proposés pour l'amélioration de la position de localisation du mobile.

Dans un premier temps, le modèle de filtrage du système est régi par le mouvement linéaire en 2D d'un piéton, dont les observations sont fournies par la technique de fingerprinting à base du réseau GRNN, ce qui donne un modèle de filtrage linéaire. Puis, en introduisant dans ce dernier une nouvelle information sur l'angle d'orientation du mobile, mesuré par une boussole digitale, un modèle de filtrage non-linéaire est obtenu. Le troisième modèle proposé consiste à linéariser le modèle de filtrage non-linéaire en remplaçant la variable de l'angle d'orientation par les deux composantes de la vitesse du mobile suivant les axes x et y, ce qui donne un système de filtrage linéarisé. Il est à noter que le choix du réseau GRNN, comme algorithme de jumelage de la technique de fingerprinting, est dû au fait qu'il donne de meilleurs résultats de localisation que les algorithmes KNN et MLP. Dans ce qui suit, la description de chacun des trois modèles de filtrage proposés est exposée.

4.3.1 Modèle de filtrage linéaire

Dans ce modèle de filtrage, le filtre de Kalman est appliqué à la technique de localisation de fingerprinting à base de l'algorithme GRNN. Pour définir le modèle de filtrage linéaire, il faux adapter les équations d'état (4.1) et d'observation (4.2) au cas de la localisation. L'équation d'état est régie par le mouvement linéaire en 2D d'un piéton, tel que :

$$\begin{cases} \text{position(k)} = \text{position(k-1)} + \text{T.vitesse(k-1)} + \frac{1}{2}\text{T}^2\text{.accélération(k-1)} \\ \text{vitesse(k)} = \text{vitesse(k-1)} + \text{T.accélération(k-1)} \end{cases} \tag{4.24}$$

Dans ce premier modèle de filtrage, et comme on n'a pas d'observations (mesures) sur la vitesse du mobile (piéton), cette dernière est supposée constante, ce qui fait que l'accélération du piéton ne devrait pas apparaître dans le modèle de filtrage (accélération nulle). Or, dans le cas réel, la vitesse du piéton n'est pas constante. Pour refléter le cas réel, l'accélération du mobile (piéton) sera modélisée dans le modèle de filtrage sous forme de bruit d'état. Ainsi, en projetant l'équation précédente suivant les directions x et y, et en considérant que les accélérations suivant x et y modélisent les composantes du bruit d'état, le modèle d'état caractérisant l'équation (4.1) sera défini comme suit :

$$\begin{bmatrix} \mathrm{x}_k \\ \mathrm{y}_k \\ \mathrm{v}_k^x \\ \mathrm{v}_k^y \end{bmatrix} = \begin{bmatrix} \mathrm{x}_{k-1} + \mathrm{T}\mathrm{v}_{k-1}^x + \frac{1}{2}\mathrm{T}^2 w_k^x \\ \mathrm{y}_{k-1} + \mathrm{T}\mathrm{v}_{k-1}^y + \frac{1}{2}\mathrm{T}^2 w_k^y \\ \mathrm{v}_{k-1}^x + \mathrm{T}w_k^x \\ \mathrm{v}_{k-1}^y + \mathrm{T}w_k^y \end{bmatrix} = \begin{bmatrix} 1 & 0 & \mathrm{T} & 0 \\ 0 & 1 & 0 & \mathrm{T} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathrm{x}_{k-1} \\ \mathrm{y}_{k-1} \\ \mathrm{v}_{k-1}^x \\ \mathrm{v}_{k-1}^y \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\mathrm{T}^2 w_k^x \\ \frac{1}{2}\mathrm{T}^2 w_k^y \\ \mathrm{T}w_k^x \\ \mathrm{T}w_k^y \end{bmatrix} \tag{4.25}$$

où x_k, y_k, v_k^x et v_k^y représentent respectivement, les composantes de la position et de la vitesse du mobile suivant les axes x et y à l'étape de temps k. T représente le temps mis entre deux positions successives (le pas d'échantillonnage). w_k^x

et w_k^y définissent les composantes du bruit d'état w_k, qui caractérisent les erreurs de modélisation du système, causées par le fait d'avoir supposé la vitesse du mobile constante (accélération nulle). Ces deux composantes de bruit d'état sont supposées centrées, gaussiennes et mutuellement indépendantes. La matrice de covariance définie par ces deux composantes du bruit d'état est donnée par :

$$Q = E\{w_k.{w_k}^\top\} = E\{\begin{bmatrix} \frac{1}{2}\mathrm{T}^2 w_k^x \\ \frac{1}{2}\mathrm{T}^2 w_k^y \\ \mathrm{T}w_k^x \\ \mathrm{T}w_k^y \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{2}\mathrm{T}^2 w_k^x & \frac{1}{2}\mathrm{T}^2 w_k^y & \mathrm{T}w_k^x & \mathrm{T}w_k^y \end{bmatrix}\} \tag{4.26}$$

Il s'en suit que :

$$Q = \begin{bmatrix} \frac{1}{4}a_x^2\mathrm{T}^4 & 0 & \frac{1}{2}a_x^2\mathrm{T}^3 & 0 \\ 0 & \frac{1}{4}a_y^2\mathrm{T}^4 & 0 & \frac{1}{2}a_y^2\mathrm{T}^3 \\ \frac{1}{2}a_x^2\mathrm{T}^3 & 0 & a_x^2\mathrm{T}^2 & 0 \\ 0 & \frac{1}{2}a_y^2\mathrm{T}^3 & 0 & a_y^2\mathrm{T}^2 \end{bmatrix} \tag{4.27}$$

où : $a_x^2 = E\{(w_k^x)^2\}$, $a_y^2 = E\{(w_k^y)^2\}$ et $E\{w_k^x.w_k^y\} = 0$. Les paramètres a_x et a_y représentent les accélérations du mobile suivant les axes x et y, modélisant les variances du bruit d'état.

Pour ce qui est du modèle d'observation, il est défini par les observations fournies par la technique de fingerprinting à base du réseau GRNN. Il est caractérisé suivant l'équation (4.2) comme suit :

$$\begin{bmatrix} \mathrm{z}_k^x \\ \mathrm{z}_k^y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \mathrm{x}_k \\ \mathrm{y}_k \\ \mathrm{v}_k^x \\ \mathrm{v}_k^y \end{bmatrix} + \begin{bmatrix} v_k^x \\ v_k^y \end{bmatrix} \tag{4.28}$$

où z_k^x et z_k^y représentent respectivement les coordonnées de la position du mobile, estimées (observées) par le réseau GRNN. v_k^x et v_k^y représentent les composantes du bruit d'observation v_k, modélisant les erreurs d'estimation de la position du mobile par le réseau GRNN. Ces deux composantes sont supposées centrées, gaussiennes et mutuellement indépendantes. La matrice de covariance du bruit d'observation définie par ces deux composantes, est donnée par :

$$R = E\{v_k.v_k^\top\} = E\{\begin{bmatrix} v_k^x \\ v_k^y \end{bmatrix} \cdot \begin{bmatrix} v_k^x & v_k^y \end{bmatrix}\} = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix} \tag{4.29}$$

où : $\sigma_x^2 = E\{(v_k^x)^2\}$, $\sigma_y^2 = E\{(v_k^y)^2\}$ et $E\{v_k^x.v_k^y\} = 0$. σ_x^2 et σ_y^2 sont les variances du bruit d'observation, qui modèlisent l'intervalle de confiance à donner aux estimations des positions de localisation.

4.3.2 Modèle de filtrage non-linéaire

Dans ce cas, le modèle de filtrage du système devient non-linéaire suite à l'introduction dans le premier modèle d'une nouvelle information sur la dynamique du mobile, qui consiste en son angle d'orientation, mesuré à l'aide d'une boussole.

La boussole utilisée dans notre travail de recherche est un produit de la compagnie Silicon [43], de type F350-COMPASS-RD. Elle est digitale et peut être reliée directement à un ordinateur portable via un port USB. La figure 4.5 montre une fenêtre d'affichage des grandeurs mesurées par cette boussole, qui sont l'angle de direction par rapport au Nord magnétique, la température et l'inclinaison. Le guide d'emploi de cette boussole ainsi que le logiciel à installer sur le portable sont disponibles dans [43].

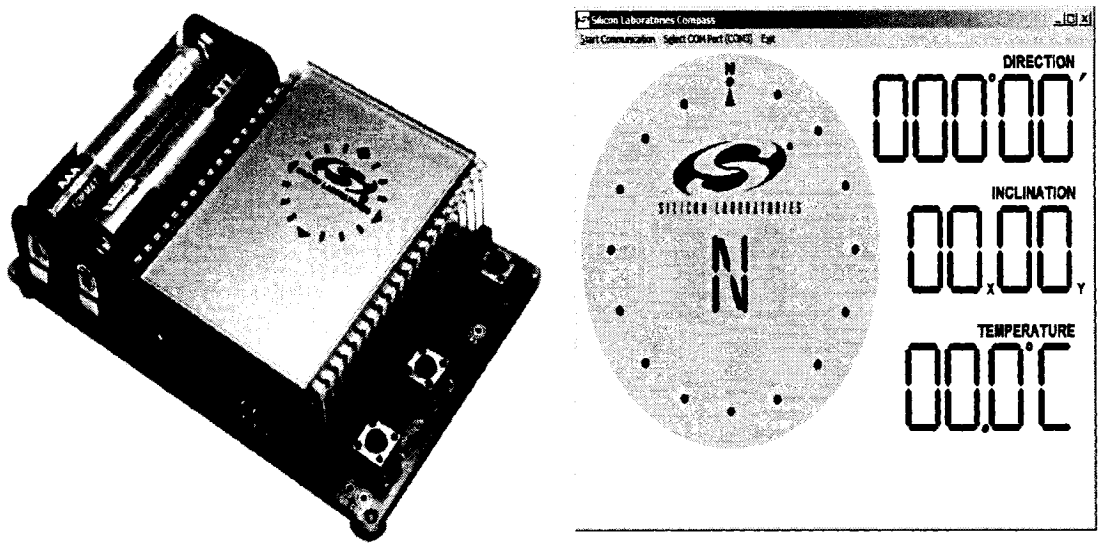


FIG. 4.5 Photo et fenêtre d'affichage de la boussole F350-COMPASS-RD.

Il est à noter que l'angle d'orientation du mobile va introduire une nouvelle variable d'état dans le modèle de filtrage du système. Cette variable va modéliser la projection de la vitesse du piéton sur les axes x et y, c'est-à-dire ($v_0 \cos(\theta)$, $v_0 \sin(\theta)$), où v_0 représente la constante de la vitesse tangentielle du mobile. Il s'en suit que cette variable va introduire une non-linéairité dans le modèle de filtrage du système, ce qui fait que le filtrage de Kalman ne peut plus être appliqué. Pour remédier à cette contrainte, deux extensions du filtrage de Kalman sont appliquées dans le cas de ce modèle de filtrage non-linéaire (le filtre de Kalman étendu (EKF) et le filtre de particules (PF)).

Ainsi, en remplaçant dans l'équation (4.24) la vitesse du mobile par ses deux composantes axiales, modélisées par l'angle d'orientation et en considérant la nouvelle variable θ_k, le modèle non-linéaire, qui caractérise l'équation d'état définie dans le

cas des filtres de Kalman étendu et de particules, est donné comme suit :

$$\begin{bmatrix} \mathrm{x}_k \\ \mathrm{y}_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} \mathrm{x}_{k-1} + \mathrm{T}\mathrm{v}_0 \cos(\theta_{k-1}) \\ \mathrm{y}_{k-1} + \mathrm{T}\mathrm{v}_0 \sin(\theta_{k-1}) \\ \theta_{k-1} \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\mathrm{T}^2 w_k^x \\ \frac{1}{2}\mathrm{T}^2 w_k^y \\ w_k^\theta \end{bmatrix} \tag{4.30}$$

où θ_k représente l'angle d'orientation du mobile à l'étape de temps k. w_k^x et w_k^y représentent les composantes du bruit d'état caractérisant l'effet d'avoir supposé la vitesse du mobile constante (accélération nulle), et w_k^θ représente la composante du bruit d'état caractérisant l'erreur de modélisation de la variable d'état θ_k. Il est à noter que cette dernière sera considérable, du fait qu'il n'est pas possible de modéliser la relation de récurrence régissant la variable d'état θ_k dans le cas du mouvement d'un piéton, où pratiquement ce dernier change d'orientation aléatoirement, c'est-à-dire qu'il n'y a pas une loi de contrôle déterministe qui régit ce changement d'orientation. Pour remédier à ce problème de modélisation, une première solution consiste à ne pas faire de correction sur la variable θ_k, c'est-à-dire que la valeur estimée de cette variable sera la même que son observation. Pour cela, la variance de la composante du bruit d'état w_k^θ est prise relativement élevée [44] pour montrer qu'on n'a pas de confiance sur l'estimation fournie par la relation de récurrence régissant la variable d'état θ_k.

En supposant que les deux composantes du bruit d'état w_k sont centrées, gaussiennes et mutuellement indépendantes, la matrice de covariance du bruit d'état dans le cas du filtrage de Kalman étendu, est donnée par :

$$Q = E\{w_k . w_k^\top\} = \begin{bmatrix} \frac{1}{2}\mathrm{T}^2 w_k^x \\ \frac{1}{2}\mathrm{T}^2 w_k^y \\ w_k^\theta \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{2}\mathrm{T}^2 w_k^x & \frac{1}{2}\mathrm{T}^2 w_k^y & w_k^\theta \end{bmatrix} = \begin{bmatrix} \frac{1}{4}a_x^2\mathrm{T}^4 & 0 & 0 \\ 0 & \frac{1}{4}a_x^2\mathrm{T}^4 & 0 \\ 0 & 0 & \sigma_\theta^2 \end{bmatrix} \tag{4.31}$$

où a_x et a_y sont définies de la même manière que dans le cas du modèle de filtrage linéaire, et σ_θ^2 représente la variance de la composante du bruit d'état w_k^θ.

Pour ce qui est du filtre de particules, la matrice Q n'est pas utilisée dans son algorithme. Ce sont plutôt les composantes du bruit d'état w_k^x qui sont prises en considération. Elles sont générées d'une façon indépendante à chaque étape de temps k, avec comme variances : a_x, a_y et σ_θ, qui sont définies de la même manière que précédemment. Généralement, ces composantes du bruit d'état sont générées de manière gaussienne, pour des raison d'optimalité du filtre et aussi pour ne pas affecter le comportement modélisant la dynamique du mobile. En effet, les particules n'auront pas à favoriser a priori l'exploration d'une direction de localisation d'une itération à l'autre du filtre.

D'un autre côté, le modèle d'observation est défini, en plus des observations fournies par le réseau GRNN, par l'angle d'orientation mesuré par la boussole. Ainsi, le modèle caractérisant l'équation d'observation définie dans le cas des filtres de Kalman étendu et de particules, est donné comme suit :

$$\begin{bmatrix} \mathrm{z}_k^x \\ \mathrm{z}_k^y \\ \mathrm{z}_k^\theta \end{bmatrix} = \begin{bmatrix} \mathrm{x}_k \\ \mathrm{y}_k \\ \theta_k \end{bmatrix} + \begin{bmatrix} v_k^x \\ v_k^y \\ 0 \end{bmatrix} \tag{4.32}$$

où les variables z_k^x, z_k^y, v_k^x et v_k^y sont définies de la même manière que dans le cas du modèle de filtrage linéaire. z_k^θ représente l'observation de l'angle d'orientation du mobile, mesurée à l'aide de la boussole. Comme remarqué, le bruit relatif à la variable z_k^θ est nul. Ceci s'explique par le fait que pour régler le problème de modélisation de la loi régissant l'angle d'orientation du piéton, la solution proposée est que la valeur de l'estimation a priori de la variable d'état θ_k doit être la même que celle fournie par l'observation z_k^θ.

La matrice de covariance R du bruit d'observation pour le filtre EKF est définie de la même manière que dans le cas du modèle linéaire, c'est-à-dire par les variances σ_x^2 et σ_y^2. Pour ce qui est du filtre de particules, la matrice de covariance R est utilisée pour évaluer les poids d'importance w_k définis par l'équation (4.23). En supposant que ces poids sont générés suivant une distribution de mesure gaussienne, les poids d'importance pour chaque particule seront estimés par :

$$\mathrm{w}_k = \mathrm{w}_{k-1} \cdot \frac{1}{2\pi|\det R|} \exp[-\frac{1}{2}(x_k - z_k)^\top R^{-1}(x_k - z_k)] \tag{4.33}$$

Il est à noter que la distribution qui génére les poids w_k est choisie gaussienne de façon à avoir une symétrie sur tout l'espace de localisation, qui est centré autour de la mesure z_k. Donc, toutes les particules auront le même rayon de confiance pour l'estimation de l'état du système.

4.3.3 Modèle de filtrage linéarisé

Dans ce modèle, on propose une deuxième solution au problème de modélisation de l'angle d'orientation θ_k dans le cas du modèle de filtrage non-linéaire. De même, avec ce modèle, il serait possible d'appliquer le filtrage de Kalman. La solution proposée consiste à linéariser le modèle de filtrage non-linéaire défini précédement, en remplaçant la variable d'état de l'angle d'orientation par deux variables représentant la projection de la vitesse du mobile sur les axes x et y, tel que :

$$\begin{cases} \mathrm{v}_k^x = \mathrm{v}_0 \cos(\theta_k) \\ \mathrm{v}_k^y = \mathrm{v}_0 \sin(\theta_k) \end{cases} \tag{4.34}$$

Ainsi, en remplaçant ces deux variables de vitesse dans le modèle défini par l'équation (4.30), le modèle de filtrage linéarisé caractérisant l'équation d'état du système est le même que celui du modèle linéaire (4.25). Dans ce modèle, les filtres de Kal-

man et de particules sont appliqués. Ainsi, la matrice de covariance du bruit d'état est définie, dans le cas du filtre de Kalman, de la même manière que celle du modèle linéaire (voir équation 4.27). Dans le cas du filtre de particules, le bruit d'état du modèle de filtrage linéarisé est généré à chaque étape de temps k de la même manière que dans le cas du modèle de filtrage non-linéaire.

Pour ce qui est du modèle d'observation, il est défini dans ce cas par quatre observations, représentant les coordonnées (x, y) de la position estimée par le réseau GRNN, et les deux vitesses axiales du piéton, calculées en remplaçant la mesure de l'angle d'orientation dans l'équation (4.34). Donc, le modèle caractérisant l'équation d'observation définie dans le cas des filtres de Kalman et de particules, est donné comme suit :

$$\begin{bmatrix} \mathrm{z}_k^x \\ \mathrm{z}_k^y \\ \mathrm{z}_k^{\mathrm{V}^x} \\ \mathrm{z}_k^{\mathrm{V}^y} \end{bmatrix} = \begin{bmatrix} \mathrm{x}_{k-1} \\ \mathrm{y}_{k-1} \\ \mathrm{v}_{k-1}^x \\ \mathrm{v}_{k-1}^y \end{bmatrix} + \begin{bmatrix} v_k^x \\ v_k^y \\ v_k^{\mathrm{V}^x} \\ v_k^{\mathrm{V}^y} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathrm{x}_{k-1} \\ \mathrm{y}_{k-1} \\ \mathrm{v}_{k-1}^x \\ \mathrm{v}_{k-1}^y \end{bmatrix} + \begin{bmatrix} v_k^x \\ v_k^y \\ v_k^{\mathrm{V}^x} \\ v_k^{\mathrm{V}^y} \end{bmatrix} \tag{4.35}$$

où z_k^x et z_k^y représentent les coordonnées (x, y) de la position estimée par le réseau GRNN. $\mathrm{z}_k^{\mathrm{V}^x}$ et $\mathrm{z}_k^{\mathrm{V}^y}$ représentent les observations des vitesses axiales (suivant x et y), calculées à partir de la mesure de l'angle d'observation θ_k, en se basant sur l'équation (4.34). v_k^x et v_k^y représentent les composantes du bruit d'observation, modélisant l'erreur commise par le GRNN sur l'estimation de la position (x,y). $v_k^{\mathrm{V}^x}$ et $v_k^{\mathrm{V}^y}$ représentent les composantes du bruit d'observation, modélisant les erreurs de mesure des vitesses axiales $\mathrm{z}_k^{\mathrm{V}^x}$ et $\mathrm{z}_k^{\mathrm{V}^y}$, retournées par l'angle d'orientation mesuré à l'aide de la boussole. Les composantes du bruit d'observation sont supposées centrées, gaussiennes et mutuellement indépendantes. Ainsi, la matrice de covariance

du bruit d'observation dans le cas du filtre de Kalman, est donnée comme suit :

$$R = E\{v_k.v_k^\top\} = E\{\begin{bmatrix} v_k^x \\ v_k^y \\ v_k^{\mathrm{V}^x} \\ v_k^{\mathrm{V}^y} \end{bmatrix} \cdot \begin{bmatrix} v_k^x & v_k^y & v_k^{\mathrm{V}^x} & v_k^{\mathrm{V}^x} \end{bmatrix}\} = \begin{bmatrix} \sigma_x^2 & 0 & 0 & 0 \\ 0 & \sigma_y^2 & 0 & 0 \\ 0 & 0 & \sigma_{\mathrm{V}^x}^2 & 0 \\ 0 & 0 & 0 & \sigma_{\mathrm{V}^y}^2 \end{bmatrix} \quad (4.36)$$

où σ_x et σ_y sont définies de la même manière que dans le cas du modèle de filtrage linéaire, σ_{V^x} et σ_{V^y} caractérisent la confiance à faire sur les mesures des vitesses axiales $z_k^{\mathrm{V}^x}$ et $z_k^{\mathrm{V}^y}$, calculées à partir de l'angle d'orientation mesuré par la boussole. Pour ce qui est du filtre de particules, cette matrice sera introduite pour générer les poids d'importance, de la même manière que dans le cas du modèle de filtrage non-linéaire (voir l'équation 4.33).

4.4 Présentation et analyse des résultats de localisation et de suivi obtenus dans chacun des trois modèles de filtrage proposés

Dans cette section, les résultats de localisation et de suivi obtenus après l'application du filtre de Kalman et de ses deux extensions (filtres de Kalman étendu et de particules) au réseau GRNN dans chacun des trois modèles de filtrage proposés, sont exposés et analysés. Le dimensionnement des paramètres des trois modèles de filtrage présentés précédemment, est donné dans ce qui suit.

4.4.1 Dimensionnement des paramètres de filtrage

Le test de localisation et de suivi a été conduit sur une trajectoire approximativement rectangulaire, formée de 92 points de localisation, où la distance entre deux points successifs est de 1m, et la vitesse tangentielle du mobile (piéton) est consi-

dérée 0.5 m/s (vitesse de marche régulière d'un piéton), ce qui fait que le temps d'échantillonnage est T=2 s. La condition initiale de l'état du système est supposée nulle.

Dans notre expérimentation, l'axe x du référentiel de mesure est situé à -45° du Nord magnétique (soit au Nord-Ouest), ce qui fait que l'angle d'orientation du mobile est calculé comme suit : θ=-(Angle magnétique mesuré par la boussole - 45). Le signe "-" est inclus dans le calcul à cause du sens de rotation par rapport au Nord magnétique qui est contraire à celui d'un référentiel (repère) cartésien.

Les paramètres a_x et a_y, caractérisant les écarts-types des composantes du bruit d'état pour les trois modèles de filtrage présentés, sont définis par les accélérations du piéton suivant les axes x et y. Comme la vitesse du mobile est de 0.5 m/s, un écart-type de 0.1 m/s est prise sur cette vitesse. De ce fait, les accélérations sont prises égales à : $a_x = a_y$ =0.05 m/s^2 (ce qui correspond au cas pratique où le piéton présente une faible accélération). Pour ce qui est de la variance σ_θ^2 de la composante du bruit d'état relative à l'angle d'orientation, définie dans le cas du modèle de filtrage non-linéaire, sa valeur est prise un peu élevée, pour les raisons citées précédement. La valeur choisie dans notre cas est $\sigma_\theta^2 = 10^4$, ce qui correspond à un écart-type de 100°.

Les paramètres σ_x et σ_y caractérisant le bruit d'observation pour les trois types de filtrage considérés, représentent la confiance à faire sur la position (x, y) estimée par le réseau GRNN. Leurs valeurs sont dimensionnées à partir des calculs d'erreur effectués sur plusieurs échantillons de mesure de la position de localisation, où : σ_x=σ_y=3.5 m. Cela signifie qu'on apporte une confiance assez large pour les positions estimées a priori, qui vont se trouver dans un périmètre de 3.5 m autour de la position retournée par le réseau GRNN. Pour ce qui est des paramètres σ_{v_x} et σ_{v_y}, définissant les écarts types des deux dernières composantes du bruit d'observation

dans le cas du modèle de filtrage linéarisé, ils modèlisent les erreurs de mesure commises par la boussole. Ces deux paramètres représentent la variation de la vitesse du mobile (suivant x et y), obtenue à partir de la mesure de l'angle d'orientation θ. Leurs valeurs sont dimensionnées à partir des calculs d'erreur effectués sur les deux vitesses axiales du mobile en remplaçant chaque mesure de l'angle d'orientation dans l'équation (4.24). Elles sont évaluées à : σ_{v_x}=σ_{v_y}=0.01 m/s.

Pour ce qui est du filtre PF, le nombre de particules est pris égale à 200, de façon à ne pas alourdir le traitement de calcul et avoir en même temps une bonne convergence du filtre.

La table 4.1 résume les différentes valeurs des paramètres de filtrage qu'on vient de dimensionner.

Paramètres	Dimensionnement
Nombre de points de la trajectoire de suivi	92
Distance entre deux points de suivi successifs	1 m
Temps d'échantillonnage	2 s
Constante de vitesse du mobile	0.5 m/s
Paramètres du bruit d'état	$a_x = a_y$ =0.05 m/s^2 $\sigma_\theta^2 = 10^4$
Paramètres du bruit de mesure	σ_x=σ_y=3.5 m σ_{v_x}=σ_{v_y}=0.01 m/s
Nombre de particules	200

TAB. 4.1 Dimensionnement des paramètres des trois modèles de filtrage proposés.

Dans ce qui suit, les résultats de localisation et de suivi pour chacun des trois modèles de filtrage présentés précédement, sont exposés et analysés, tout en donnant l'amélioration apportée par chaque modèle par rapport à l'autre. De même, l'influence des paramètres de filtrage sur la précision de localisation et de suivi est exposée.

4.4.2 Résultats et analyse des performances de localisation et de suivi dans le cas du modèle de filtrage linéaire

Les figures 4.6 et 4.7 montrent respectivement, la distribution cumulative (CDF) de l'erreur de localisation et la courbe de l'erreur de localisation instantanée, avant et aprés l'application du filtre de Kalman à l'algorithme de localisation GRNN. Ici, l'erreur de la position de localisation est définie en termes de distance Euclidienne (voir l'équation 3.7).

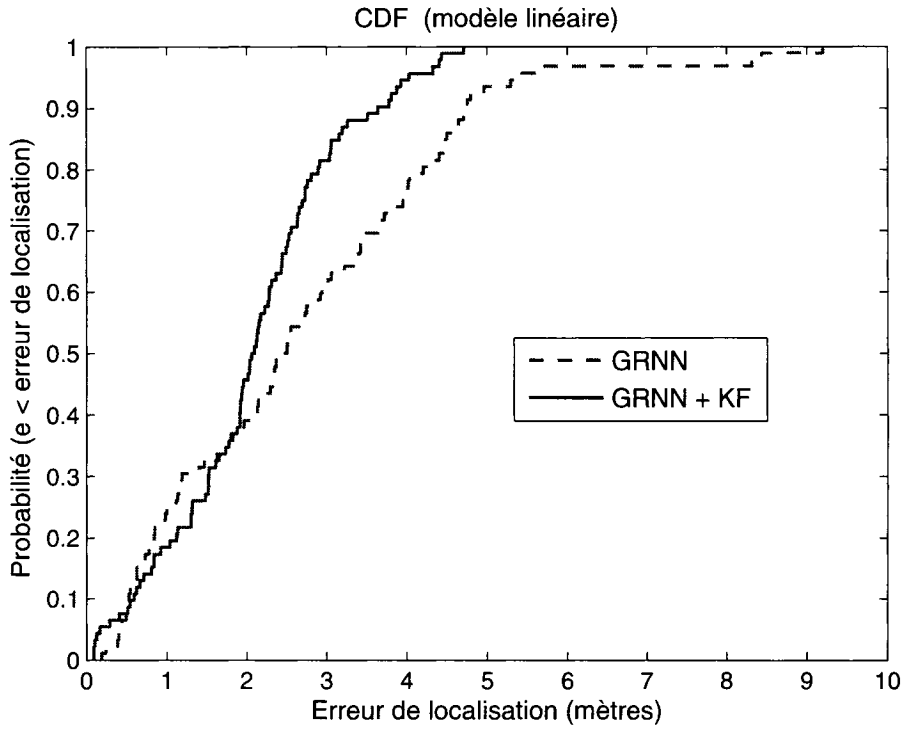


FIG. 4.6 Courbes des distributions cumulatives (CDF) de l'erreur de localisation avant et après l'application du filtre de Kalman au réseau GRNN, dans le cas du modèle de filtrage linéaire.

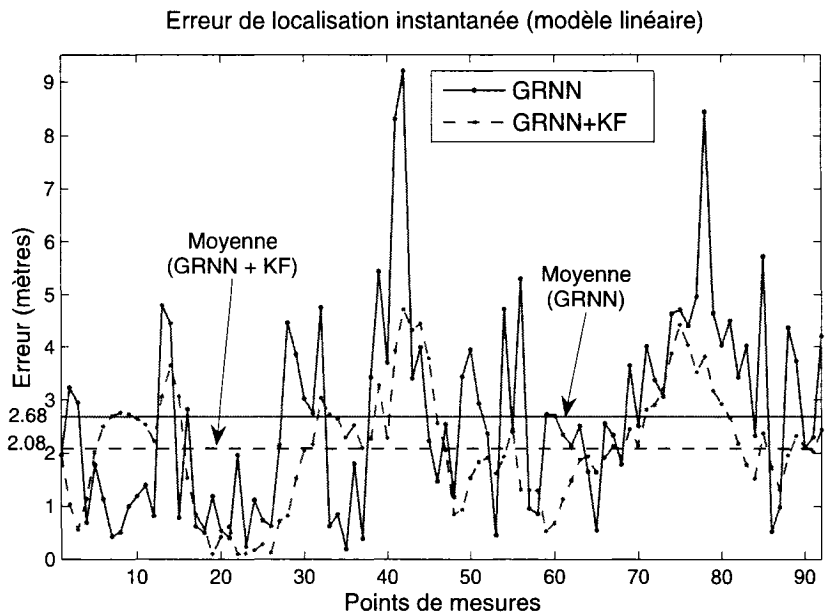


FIG. 4.7 Courbes de l'erreur de localisation instantanée avant et après l'application du filtre de Kalman au réseau GRNN, dans le cas du modèle de filtrage linéaire.

À partir des figures 4.6 et 4.7, on remarque que l'apport de l'application du filtre Kalman est considérable en termes d'amélioration de la précision de localisation et de suivi, où les deux courbes de l'erreur de localisation instantanée et de CDF de l'erreur sont meilleures après l'application du filtre de Kalman au réseau GRNN. D'un autre côté, la figure 4.7 montre que les maximas des erreurs de localisation sont diminués aprés l'application du filtre de Kalman au réseau GRNN, où le maximum de l'erreur de localisation enregistré par le GRNN est de 9.2 m, alors qu'aprés l'application du filtre de Kalman il est réduit à seulement 4.7 m.

La table 4.2 résume les informations contenues dans les figures 4.6 et 4.7, en donnant les performances de localisation avant et après l'application du filtre de Kalman au réseau GRNN, en termes d'erreur moyenne, d'erreurs obtenues pour 50% (médiane), 75% et 95% des données estimées, de maximum et de minimum d'erreur. Les résultats de la table 4.2 illustrent bien l'apport de l'application du filtre de Kalman au réseau GRNN, où la précision de la localisation est améliorée d'environ 22.4%

en termes d'erreur moyenne, et d'environ 15.5% en termes d'erreur médiane. Ceci justifie ainsi l'approche d'amélioration proposée, qui est basée sur l'application du filtrage de Kalman au réseau GRNN.

Algorithme	Erreur de localisation (m)					
	Moyenne	Médiane	75%	95%	Min	Max
GRNN	2.68	2.45	3.62	5.42	0.19	9.20
GRNN + KF	2.08	2.07	2.56	4.02	0.09	4.71
Gain de KF/GRNN	22.39%	15.51%	29.28%	25.83%	-	-

TAB. 4.2 Performances de localisation avant et après l'application du filtre de Kalman au réseau GRNN, dans le cas du modèle de filtrage linéaire.

La figure 4.8 montre les trajectoires de suivi du mobile obtenues par les données réelles, estimées par le réseau GRNN et estimées par l'application du filtre de Kalman au réseau GRNN.

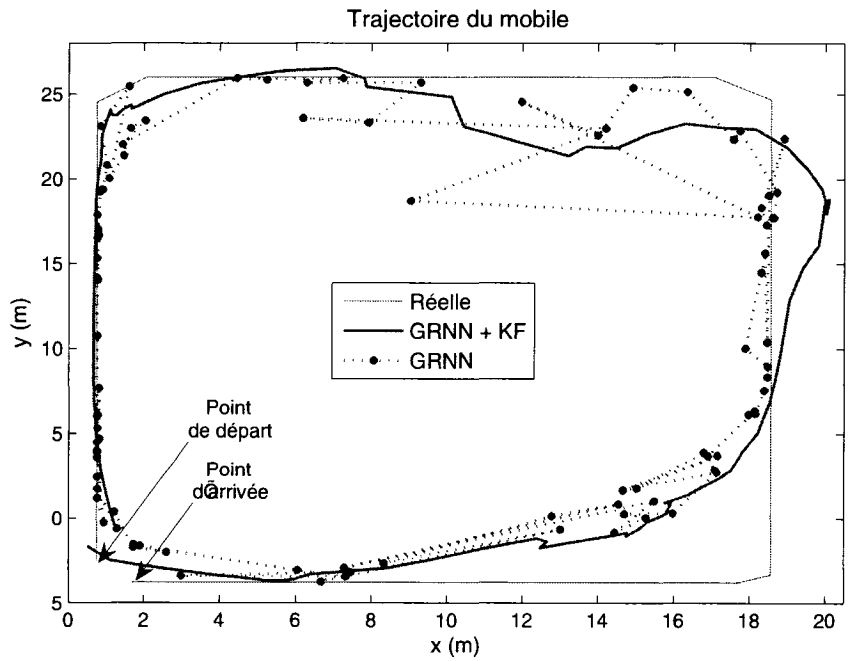


FIG. 4.8 Trajectoires de suivi du mobile obtenues par les données réelles, avant et après l'application du filtre de Kalman au réseau GRNN, dans le cas du modèle de filtrage linéaire.

À partir de la figure 4.8, on constate que l'application du filtre de Kalman au réseau GRNN améliore la trajectoire du mobile et la rend plus proche de celle réelle, alors que celle obtenue par le réseau GRNN seul, présente des discontinuités et des erreurs d'orientation qui sont dues au fait que le réseau GRNN estime la localisation du mobile d'une manière statique, c'est-à-dire qu'il ne tient pas compte de la dynamique du mobile.

4.4.3 Résultats et analyse des performances de localisation et de suivi dans le cas du modèle de filtrage non-linéaire

La figure 4.9 montre les courbes des distributions cumulatives (CDF) de l'erreur de localisation, obtenues avant et après l'application des filtres de Kalman étendu (EKF) et de particules (PF) au réseau GRNN, dans le cas du modèle de filtrage non-linéaire.

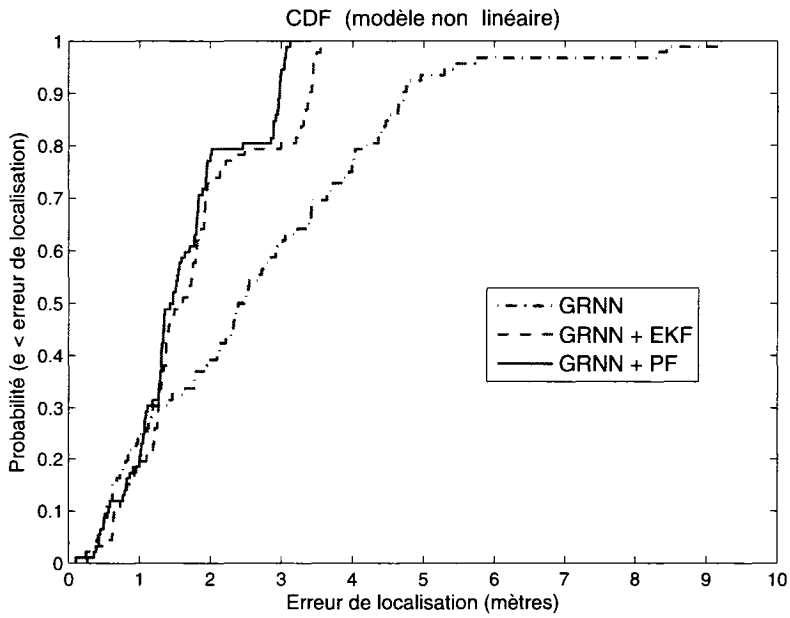


FIG. 4.9 Courbes des distributions cumulatives (CDF) de l'erreur de localisation avant et après l'application des filtres EKF et PF au réseau GRNN, dans le cas du modèle de filtrage non-linéaire.

À partir de la figure 4.9, on constate que les courbes CDF de l'erreur de localisation obtenues par les filtres EKF et PF sont meilleures, en termes de précision de localisation, que celle obtenue par le réseau GRNN seul. D'un autre côté, la courbe CDF de l'erreur de localisation obtenue par le filtre PF est meilleure que celle obtenue par le filtre EKF.

La table 4.3 résume les informations contenues dans la figure 4.9, en donnant les performances de localisation avant et après l'application de chacun des filtres EKF et PF au réseau GRNN, en termes d'erreur moyenne, d'erreur médiane (erreur obtenue pour 50% des données estimées) et d'erreurs obtenues pour 75% et 95% des données estimées. Les résultats de la table montrent bien que l'application des filtres EKF et PF améliore la précision de localisation du mobile. De même, le filtre PF donne relativement une meilleure estimation de localisation par rapport au filtre EKF.

Algorithme	Erreur de localisation (m)			
	Moyenne	Médiane (50%)	75%	95%
GRNN	2.68	2.45	3.62	5.42
GRNN + EKF	1.79	1.59	1.93	3.45
GRNN + PF	1.62	1.45	1.83	3.04

TAB. 4.3 Performances de localisation avant et après l'application des filtres EKF et PF au GRNN, dans le cas du modèle de filtrage non-linéaire.

D'un autre côté, la table 4.4 donne une comparaison entre les performances de localisation, en termes d'erreur moyenne, obtenues par l'application des filtres EKF et PF au réseau GRNN, dans le cas du modèle de filtrage non-linéaire, et celles obtenues par l'application du filtre de Kalman au réseau GRNN, dans le cas du modèle de filtrage linéaire. À partir de cette table, on constate que les filtres EKF et PF donnent de meilleures précisions de localisation, en termes d'erreur moyenne, que le filtre KF appliqué au réseau GRNN dans le cas du modèle de filtrage linéaire,

où les gains en erreur moyenne obtenus après l'application des filtres PF et EKF au réseau GRNN dans le cas du modèle non-linéaire, sont respectivement d'environ 39% et 33%, alors que le gain obtenu par le filtre KF appliqué dans le cas du modèle de filtrage linéaire, est d'environ 22%. Ceci justifie le fait d'ajouter la boussole au système de localisation, qui a permis d'avoir une information supplémentaire sur la dynamique du mobile, qui consiste en son angle d'orientation. Cette information a permis de mieux refléter (observer) le mouvement réel du mobile.

	Algorithme	Moyenne (m)	Gain filtrage/GRNN (%)
	GRNN	2.68	-
Modèle linéaire	GRNN+KF	2.08	22.40
Modèle non-linéaire	GRNN+EKF	1.79	33.21
	GRNN+PF	1.62	39.18

TAB. 4.4 Comparaison des performances de localisation obtenues dans le cas du modèle de filtrage linéaire à celles obtenues dans le cas du modèle de filtrage non-linéaire.

La figure 4.10 illustre mieux l'avantage d'observer l'angle d'orientation du mobile, mesuré par la boussole, où les trajectoires du mobile obtenues après l'application des filtres PF et EKF au réseau GRNN dans le cas du modèle de filtrage non-linéaire, sont plus proche de la trajectoire réelle, et plus orientées que celles obtenues par le réseau GRNN et le filtre KF appliqué à ce dernier dans le cas du modèle de filtrage linéaire. En fait, l'observation de l'angle d'orientation du mobile a permis de mieux orienter la trajectoire estimée du mobile.

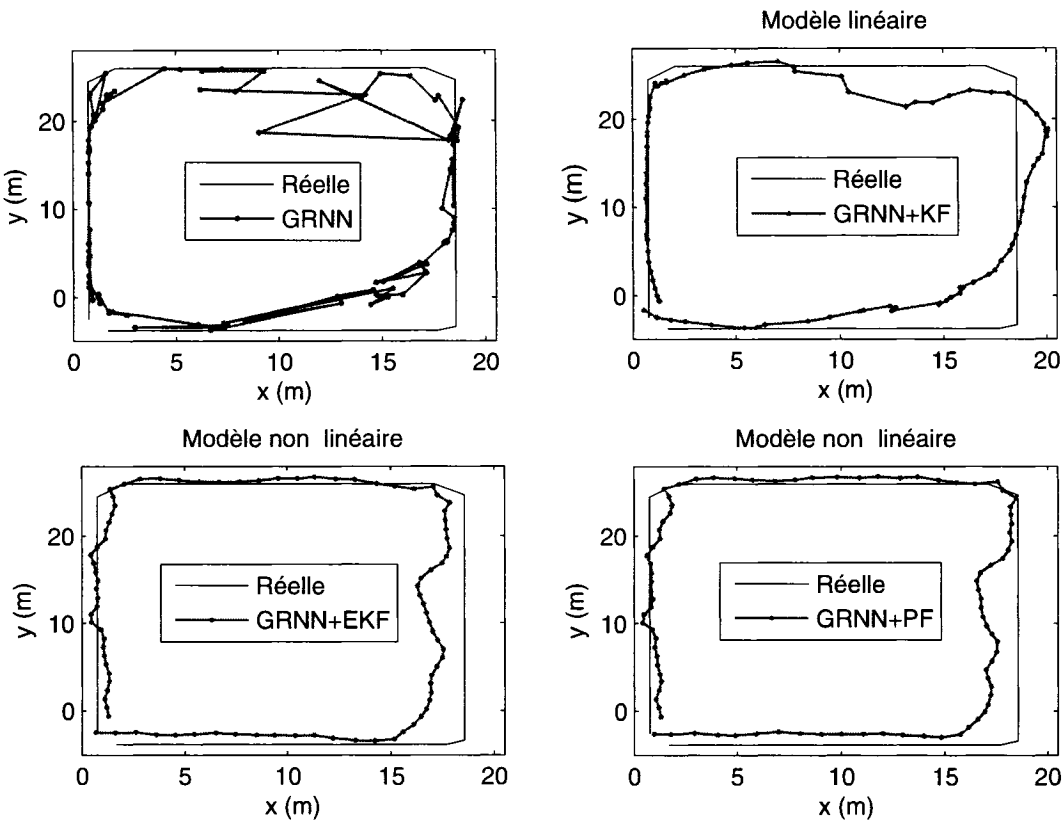


FIG. 4.10 Comparaison des trajectoires de suivi du mobile obtenues avant et après l'application au réseau GRNN, des filtres EKF, PF (modèle de filtrage non-linéaire) et KF (modèle de filtrage linéaire).

4.4.4 Résultats et analyse des performances de localisation et de suivi dans le cas du modèle de filtrage linéarisé

La figure 4.11 montre les courbes des distributions cumulatives (CDF) de l'erreur de localisation avant et après l'application des filtres KF et PF au réseau de neurones GRNN, dans le cas du modèle de filtrage linéarisé. Sur cette figure, on peut remarquer que l'application des filtres KF et PF au réseau GRNN, dans le cas du modèle de filtrage linéarisé, améliore mieux l'erreur de localisation du mobile, où leurs courbes de CDF sont meilleures que la courbe obtenue par le réseau GRNN.

D'un autre côté, la courbe de CDF de l'erreur de localisation obtenue après l'application du filtre PF est nettement meilleure que celle obtenue par l'application du filtre KF. Ceci s'explique par le fait que le filtre PF modélise mieux le comportement aléatoire des bruits par rapport au filtre KF, où les particules générées dans le cas du filtre PF, permettent d'explorer pour une seule estimation de localisation, tout un espace centré autour de l'observation du système.

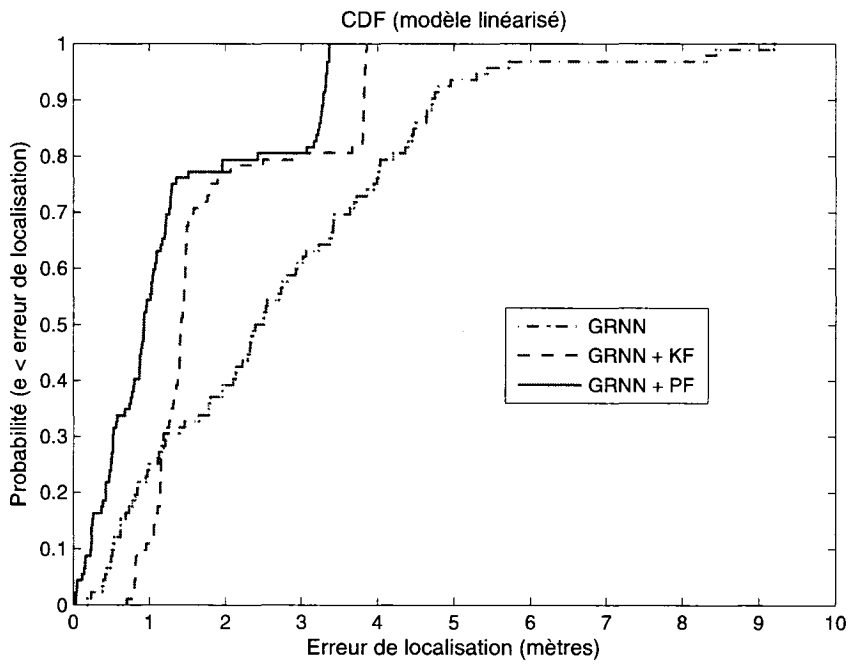


FIG. 4.11 Courbes des distributions cumulatives de l'erreur de localisation avant et après l'application des filtres KF et PF au réseau GRNN, dans le cas du modèle de filtrage linéarisé.

La table 4.5 résume les informations contenues dans la figure 4.11, en donnant les performances de localisation avant et après l'application des filtres KF et PF au réseau GRNN dans le cas du modèle de filtrage linéarisé, en termes d'erreur moyenne, d'erreur médiane (erreur obtenue pour 50% des données estimées) et d'erreurs obtenues pour 75% et 95% des données estimées. Les résultats de la table montrent bien que l'application des filtres KF et PF améliore la précision de la position de

localisation du mobile, surtout dans le cas du filtre PF, où ce dernier donne une précision de localisation nettement meilleure par rapport au réseau GRNN. Ce qui valide ainsi, l'approche d'amélioration proposée, basée sur l'application du filtrage de Kalman et de particules au réseau GRNN dans le cas du modèle de filtrage linéarisé.

Algorithme	Erreur de localisation (m)			
	Moyenne	Médiane (50%)	75%	95%
GRNN	2.68	2.45	3.62	5.42
GRNN + KF	1.83	1.42	1.58	3.84
GRNN + PF	1.33	0.97	1.38	3.21

TAB. 4.5 Performances de localisation et de suivi avant et après l'application des filtres KF et PF au réseau GRNN, dans le cas du modèle de filtrage linéarisé.

D'un autre côté, la figure 4.12 donne une comparaison entre les courbes de CDF de l'erreur de localisation, obtenues par les filtres appliqués au réseau GRNN dans le cas des modèles de filtrage linéaire (M.L), non-linéaire (M.NL) et linéarisé (M.LZ). À partir de cette figure, on constate que globalement, le filtre PF appliqué au réseau GRNN dans le cas du modèle de filtrage linéarisé (M.LZ), donne la meilleure courbe de CDF par rapport aux autres filtres. Ceci, justifie ainsi le fait de linéariser le modèle de filtrage non-linéaire, dans le but de mieux modéliser la relation de récurrence régissant la variable de l'angle d'orientation du mobile, alors que dans le modèle non-linéaire cette modélisation n'était pas judicieuse. D'une autre part, on remarque sur la figure 4.12 que les courbes de CDF obtenues par les filtres appliqués dans le cas des modèles de filtrage non-linéaire et linéarisé, commencent à avoir des allures verticalement croissantes à partir d'un certain pourcentage de données estimées (environ 85% des données, voir la zone encadrée sur la figure 4.12). Ceci peut être expliquer par le fait qu'il n'y a pas assez de données (92 positions seulement), ce qui fait que l'erreur de localisation en termes de % CDF, évolue faiblement dans la zone encadrée sur la figure 4.12. Pour bien illustrer le

problème, on prend la CDF de l'erreur de localisation obtenue par le filtre KF appliqué au réseau GRNN dans le cas du modèle de filtrage linéarisé. L'erreur de localisation pour 85% des données estimées est de 3.81 m, et celle obtenue pour 100% des données de mesures (maximum de l'erreur de localisation) est de 3.86 m. Ceci montre que la variation de l'erreur de localisation, en termes de % CDF, est vraiment infime dans la zone spécifiée sur la figure 4.12.

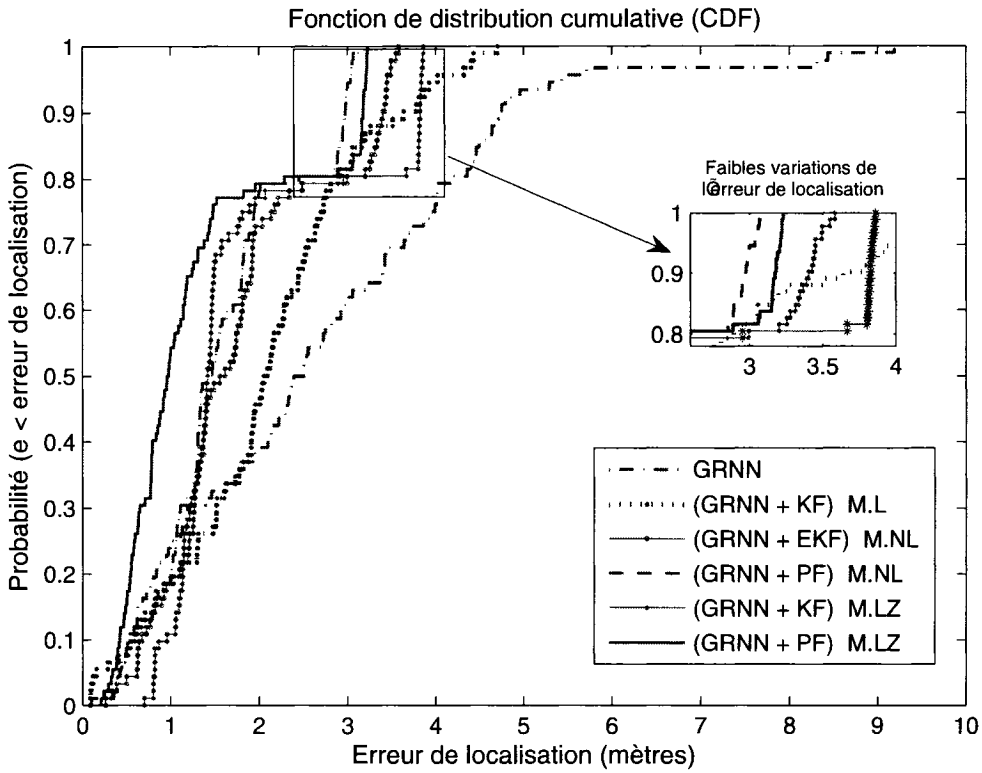


FIG. 4.12 Courbes des distributions cumulatives de l'erreur de localisation obtenues avant et après l'application des différents filtres au réseau GRNN dans le cas de chacun des trois modèles de filtrage proposés.

La table 4.6 donne une comparaison des performances de localisation, en termes d'erreur moyenne, obtenues par l'application des filtres KF et PF au réseau GRNN, dans le cas du modèle de filtrage linéarisé, à celles obtenues dans le cas des modèles de filtrage linéaire et non-linéaire. Sur cette table, les performances du filtre PF,

appliqué dans le cas du modèle de filtrage linéarisé, sont bien meilleures que les autres filtres, tel que l'amélioration obtenue après l'application de ce dernier au réseau GRNN, est d'environ 50% en termes d'erreur moyenne de la position de localisation du mobile. Ce qui montre ainsi la puissance d'estimation du filtre de particules, appliqué dans le cas du modèle de filtrage linéarisé, par rapport aux autres filtres appliqués dans les modèles de filtrage linéaire et non-linéaire.

	Algorithme	Moyenne (m)	Gain filtrage/GRNN (%)
	GRNN	2.68	-
Modèle linéaire	KF	2.08	22.40
Modèle non-linéaire	EKF	1.79	33.21
	PF	1.62	39.18
Modèle linéarisé	KF	1.83	31.75
	PF	1.33	50.37

TAB. 4.6 Comparaison des performances de localisation entre les différents filtres appliqués au réseau GRNN dans le cas des trois modèles de filtrage proposés.

D'un autre côté, la figure 4.13 donne une comparaison des trajectoires de suivi du mobile obtenues par les filtres KF et PF, appliqués au réseau GRNN dans le cas du modèle de filtrage linéarisé, à celles obtenues par les filtres appliqués au réseau GRNN dans le cas des modèles de filtrage linéaire et non-linéaire. On remarque que les trajectoires du mobile obtenues par l'application des filtre KF et PF au réseau GRNN, dans le cas du modèle de filtrage linéarisé, sont relativement meilleures que les trajectoires obtenues dans le cas des modèles de filtrage linéaire et non-linéaire, et elles sont plus proches de la trajectoire réelle. Ce qui justifie ainsi la proposition du modèle de filtrage linéarisé pour l'amélioration des performances de localisation et de suivi du mobile.

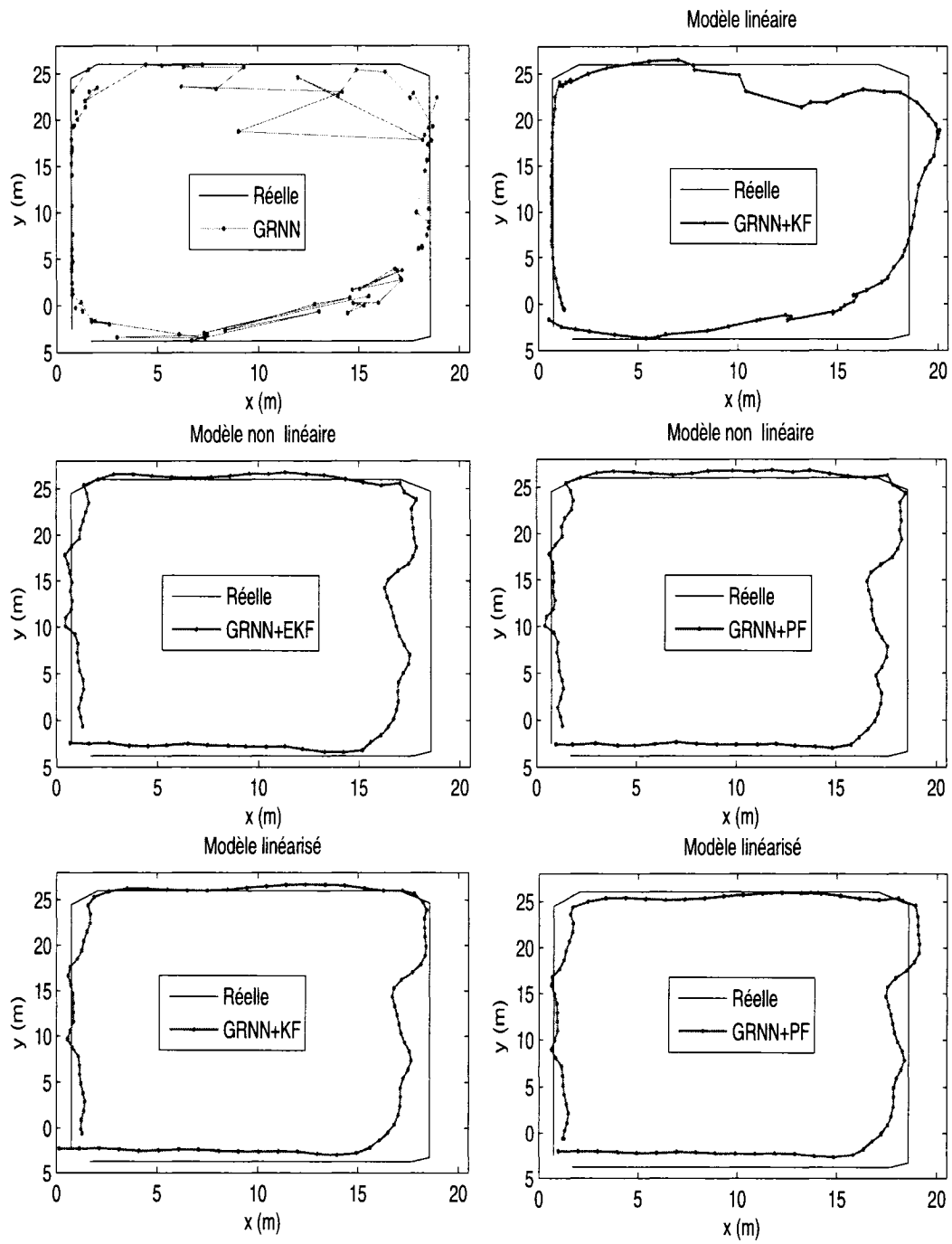


FIG. 4.13 Comparaison des trajectoires de suivi du mobile obtenues par les filtres KF et PF, appliqués au réseau GRNN dans le cas du modèle de filtrage linéarisé, à celles obtenues par les filtres appliqués au réseau GRNN dans le cas des modèles de filtrage linéaire et non-linéaire.

La figure 4.14 montre les variations des vitesses axiales du piéton (suivant x et y) dans le cas du filtre de Kalman appliqué au GRNN dans les modèles de filtrage linéaire et non-linéaire. À partir de la figure, on remarque que les courbes des vitesses axiales dans le cas du modèle linéarisé (où les vitesses sont observées à l'aide de la boussole) sont meilleures que dans le cas du modèle de filtrage linéaire (où il n'y a aucune observation sur les vitesses). Ceci montre l'avantage d'utiliser une boussole pour observer l'orientation des vitesses (suivant les axes x et y).

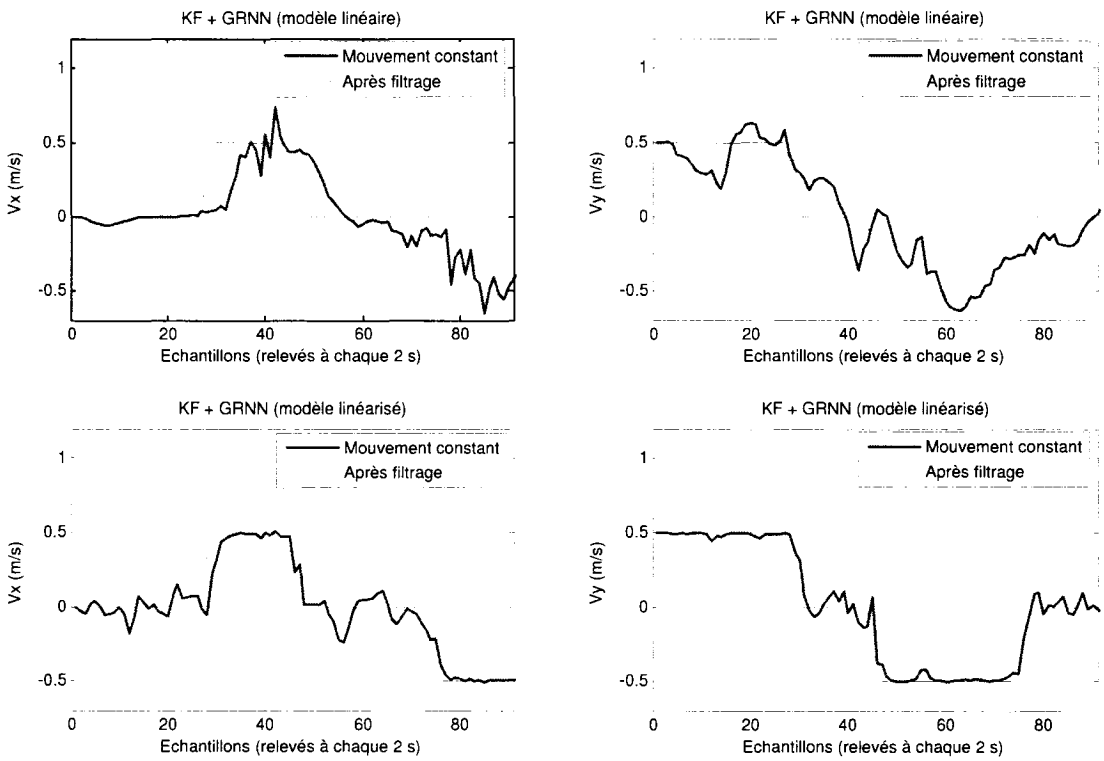


FIG. 4.14 Courbes des vitesses axiales du piéton (suivant x et y) obtenues par filtrage de Kalman appliqué au GRNN dans le cas des modèles linéaire (pas de boussole) et linéarisé (utilisation de la bousole).

4.4.5 Analyse des performances de localisation et de suivi selon les paramètres de filtrage

Dans cette partie, une analyse des performances de localisation et de suivi selon les différents paramètres de filtrage est faite, dont les variances du bruit de mesure, le temps de traitement d'une estimation de localisation donnée et le nombre de particules utilisés par le filtre PF.

4.4.5.1 Influence des paramètres du bruit de mesure sur les performances de localisation et de suivi

La figure 4.15 montre l'influence des paramètres σ_x et σ_y sur les performances de localisation, en termes d'erreur moyenne. Ces paramètres caractérisent les écarts-types du bruit de mesure dans le cas du filtre de Kalman appliqué au réseau GRNN dans le modèle de filtrage linéaire. Sur cette figure, on remarque que la courbe de l'erreur de localisation moyenne converge vers un minimum d'erreur pour les valeurs optimales de $\sigma_x = 1m$ (sur la figure r_x) et de $\sigma_y = 5m$ (sur la figure r_y=5m). Ces valeurs optimales (spécifiques aux données de mesures considérées) représentent les écarts-types réelles de l'erreur commise sur l'observation de la position (x, y) fournie par le réseau GRNN. Avec ces deux valeurs optimales, la précision de localisation et de suivi est relativement améliorée, où elle passe, dans le cas du filtre de Kalman utilisé dans le modèle de filtrage linéaire, de 2.08 mètres à 2 mètres. Toutefois, en pratique, ce genre de dimensionnement ne peut pas être effectuée a priori, du fait qu'on ne dispose pas des données réelles pour effectuer les calculs des écarts-types sur les erreurs d'observation. Comme alternative, les valeurs assignées à ces deux paramètres du bruit de mesure, consistent à donner une marge de confiance aux estimations a posteriori de l'état du système, qui seront centrées autour des données d'observation.

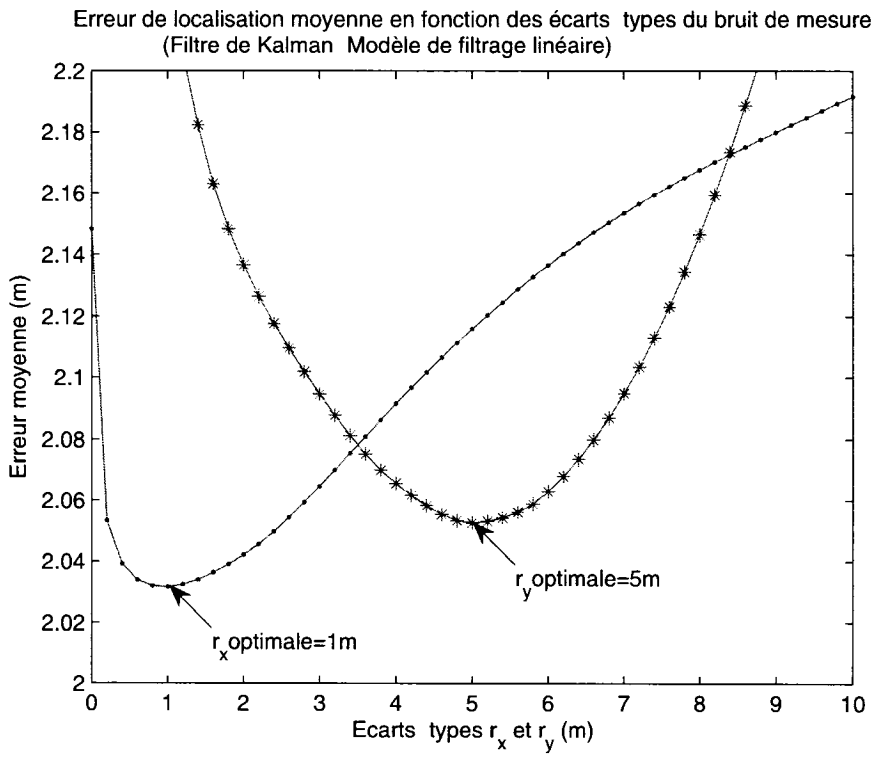


FIG. 4.15 Influence des paramètres du bruit de mesure sur la précision de localisation du mobile, dans le cas du filtre de Kalman appliqué au réseau GRNN dans le modèle de filtrage linéaire.

4.4.5.2 Temps de traitement d'une estimation de localisation

La table 4.7 montre le temps de traitement d'une estimation de localisation donnée, pour chacun des filtres appliqués au réseau GRNN dans le cas des trois modèles de filtrage proposés. Cette table montre que le filtre PF, appliqué dans le cas des modèles de filtrage non-linéaire et linéarisé, consomme beaucoup plus de temps à estimer une localisation donnée par rapport aux autres filtres. Ceci est dû au fait que le filtre PF est basé sur la génération d'un certain nombre de particules (échantillons) pour faire une estimation donnée. Par conséquent, dans les applications où le temps entre deux points de suivi (temps d'échantillonnage) est assez petit, on

prèfère appliquer le filtre de Kalman (KF) dans le cas des systèmes linéaires, et le filtre de Kalman étendu (EKF) dans le cas des systèmes non-linéaires. Donc, l'application du filtre PF va dépendre du temps d'échantillonnage, mis entre deux positions successives de suivi.

	Algorithme	Temps de traitement (ms)
	GRNN	25
Modèle linéaire	GRNN + KF	25 + 16
Modèle non-linéaire	GRNN + EKF	25 + 17
	GRNN + PF	25 + 56
Modèle linéarisé	GRNN + KF	25 + 17
	GRNN + PF	25 + 53

TAB. 4.7 Temps de traitement d'une estimation de localisation avant et après l'application au réseau GRNN, de chacun des filtres utilisés dans le cas des modèles de filtrage linéaire, non-linéaire et linéarisé.

4.4.5.3 Effet du nombre de particules sur les performances de localisation et de suivi

La table 4.8 montre l'influence du nombre de particules sur les performances de localisation et de suivi, en termes d'erreur moyenne et de temps de traitement, dans le cas du filtre PF appliqué au réseau GRNN dans le modèle de filtrage linéarisé. Les résultats de cette table montrent que plus on augmente le nombre de particules, plus l'erreur de localisation (en termes d'erreur moyenne) diminue, et ainsi les performances de localisation et de suivi deviennent de plus en plus meilleures. Ceci s'explique par le fait que si le nombre de particules est grand, les endroits d'estimation explorés par les particules couvrent mieux la position de localisation réelle. Donc, c'est plus probable de cerner cette position de localisation réelle. De même, le filtre PF devient plus stable en termes d'estimation d'une même position de localisation, d'une simulation à l'autre. Cependant, augmenter le nombre de particules

indéfiniment n'améliore pas la localisation. Au contraire, il y aura une dégénérescence du filtre, où les endroits explorés par les particules deviennent de plus en plus éloignés de la position de localisation réelle. D'un autre côté, si le nombre de particules est augmenté indéfiniment, le filtre PF va consommer beaucoup plus de temps pour fournir une estimation de localisation donnée, et devient pratiquement non-réalisable. Pour cela, il faut bien choisir a priori le nombre de particules, en faisant un compromis entre la précision de localisation que peut fournir le filtre PF, et le temps de traitement d'une estimation de localisation donnée.

Nombre de particules	Erreur moyenne (m)	Gain en erreur moyenne de PF/GRNN (%)	Temps de traitement de PF+GRNN (ms)
50	1.71	36.27	17 + 25
100	1.53	43.05	29 + 25
200	1.33	50.37	53 + 25
1000	1.46	45.68	129 + 25
2000	1.48	44.65	682 + 25

TAB. 4.8 Effet du nombre de particules sur les performances de localisation et de suivi dans le cas du filtre de particules appliqué au réseau GRNN dans le cas du modèle de filtrage linéarisé.

4.5 Conclusion

Dans ce chapitre, l'approche du filtrage de Kalman et deux de ses extensions a été appliquée au système de localisation de fingerprinting à base du réseau GRNN, pour améliorer les performances de localisation et de suivi dans un environnement interne. Trois variantes d'amélioration relatives au modèle de filtrage caractérisant la dynamique du mobile, ont été proposées.

Dans un premier temps, seul le filtrage de Kalman a été appliqué au réseau GRNN, où le modèle de filtrage était linéaire. Ce dernier était caractérisé par quatre états

(position et vitesse du mobile suivant les axes x et y) modélisant le mouvement linéaire en 2D d'un piéton, et par deux observations correspondant à la position (x, y) estimée par le réseau GRNN. Les résultats expérimentaux obtenus ont montré l'apport de l'application du filtrage de Kalman au réseau GRNN sur l'amélioration de la précision de la position de localisation, où cette dernière était d'environ 22.4% en termes d'erreur moyenne. De même, les résultats ont montrés que la trajectoire de suivi du mobile, obtenue après l'application du filtre de Kalman au réseau GRNN, était meilleure et plus proche de la trajectoire réelle du mobile, alors que celle obtenue par le réseau GRNN comportait des erreurs d'orientation et des discontinuités.

Par la suite, une boussole digitale a été combinée au système de localisation de fingerprinting, pour apporter au modèle de filtrage une nouvelle observation sur la dynamique du mobile, qui consiste en l'angle d'orientation du mobile. Avec cette nouvelle observation, le modèle de filtrage est devenu non-linéaire. Comme l'application du filtre de Kalman exige la contrainte de linéarité, deux de ses variantes ont été appliquées : le filtre de Kalman étendu (EKF) et le filtre de particules (PF). Les résultats de localisation et de suivi obtenus après l'utilisation de la boussole étaient beaucoup plus meilleurs. Les filtres EKF et PF, appliqués au réseau GRNN dans le cas de ce modèle de filtrage, ont donné de meilleures performances de localisation et de suivi par rapport au filtre de Kalman, appliqué dans le cas du modèle de filtrage linéaire, où les améliorations apportées sur la précision de localisation, en termes d'erreur moyenne, étaient d'environ 39% pour le filtre PF et de 33% pour le filtre EKF, alors que dans le cas du filtre KF, l'amélioration était de 22%. De même, les trajectoires de suivi du mobile, obtenues par les filtres EKF et PF, étaient meilleures que celle du filtre KF, où elles étaient plus orientées et plus proches de la trajectoire de suivi réelle.

Finalement, la dernière variante d'amélioration qui a été proposée, était basée sur

un modèle de filtrage linéarisé. Ce dernier a été obtenu en remplaçant dans le modèle de filtrage non-linéaire, la variable de l'angle d'orientation par les deux vitesses axiales (suivant x et y) du mobile. Dans ce modèle de filtrage, les filtres de Kalman et de particules ont été appliqués au réseau GRNN. Les résultats obtenus dans le cas de ce modèle, lorsque le filtre PF a été appliqué au réseau GRNN, étaient meilleurs que ceux obtenus dans le cas des modèles de filtrage linéaire et non-linéaire, pour ce qui est du filtre PF. Quant à l'application du filtre KF au réseau GRNN dans ce même modèle de filtrage, les résultats de localisation et de suivi étaient relativement les mêmes que ceux du filtre EKF appliqué dans le cas du modèle de filtrage non-linéaire. Dans ce dernier modèle de filtrage, le filtre PF appliqué au réseau GRNN, a permis d'obtenir une amélioration de localisation, en termes d'erreur moyenne, d'environ 50%. De même, la trajectoire de suivi obtenue par ce filtre était plus proche de la trajectoire réelle comparée aux autres trajectoires obtenues dans le cas des modèles de filtrage linéaire et non-linéaire.

Sur un autre plan, une analyse a été faite par rapport à l'influence des paramètres de filtrage sur les performances de localisation et de suivi. Parmi les paramètres de filtrage qui ont été considérés, on distingue les variances définissant le bruit de mesure. Il a été constaté que pour des valeurs optimales de ces dernières (spécifique aux données considérées), la précision de localisation du mobile a été relativement améliorée. Cependant, en pratique on ne dispose a priori d'aucune donnée réelle pour faire ce calcul optimal. Pour cela, il faut dimensionner ces deux paramètres d'une manière générale et non optimale, en donnant par exemple, une marge de confiance sur les estimations observées, comme fût le cas avec les paramètres que nous avons utilisés. Une autre analyse a été faite selon le temps de traitement d'une estimation de localisation donnée. Les résultats ont montré que, certes le filtre PF donnent de meilleurs précisions de localisation que les autres filtres, mais toutefois il est gourmand en matière de temps de traitement. Une dernière analyse a été faite

par rapport à l'influence du nombre de particules sur la précision de localisation et sur le temps de traitement dans le cas du filtre de particules, utilisé dans le modèle de filtrage linéarisé. Les résultats qui ont été obtenus, nous ont permis de conclure qu'il faut faire un compromis par rapport au nombre de particules, de manière à avoir une bonne précision de localisation avec un temps de traitement pas assez grand, du moins inférieur au temps d'échantillonnage, pour que le filtre soit applicable.

En somme, l'approche de filtrage que nous avons proposé pour améliorer les performances de localisation et de suivi dans un environnement interne, a été validée.

CHAPITRE 5

CONCLUSION GÉNÉRALE

Dans ce mémoire, une approche dynamique basée sur le filtrage de Kalman et de ses extensions, a été appliquée à un système de localisation basé sur la technique de fingerprinting, pour améliorer les performances de localisation et de suivi d'un mobile dans un environnement interne équipé par un réseau sans fil WLAN. Ce présent travail s'est opéré essentiellement sur trois phases successives : la phase expérimentale de collecte de mesure, la phase d'estimation de la localisation du mobile par la technique de fingerprinting, et la phase de filtrage ou d'amélioration des performances de localisation et de suivi du mobile.

Dans la première phase, le processus de collecte des mesures a été effectué, où plusieurs mesures ont été recueillies dans l'environnement de localisation, considéré au $5^{ème}$ étage du pavillon Lassonde, sis à l'École polytechnique de Montréal. Ces mesures ont permis de construire la base de données de localisation, contenant les niveaux de puissances RSS (émis par les points d'accès du réseau WLAN), avec les positions de localisation où ils ont été reçus. À partir des résultats de mesures, l'analyse faite sur le comportement du signal RSS a montré que le niveau de ce dernier fluctue au cours du temps pour une même position de localisation. Ceci se justifie par le fait que l'environnement de localisation considéré (environnement interne sans fil) est caractérisé par la propagation du signal par mutlitrajet, et aussi par la non-visibilité directe entre l'émetteur (AP) et le récepteur (mobile).

Dans la deuxième phase, la technique de fingerprinting a été utilisée pour estimer la position de localisation du mobile (piéton dans notre cas). Pour cela, trois algorithmes de jumelage, correspondant à cette technique, ont été examinés (KNN,

MLP et GRNN). Les résultats obtenus ont montré que le réseau GRNN donne de meilleures performances de localisation que la méthode KNN et le réseau MLP. La précision de localisation, en termes d'erreur moyenne, était de 2.68 m pour le GRNN, de 2.85 m pour le KNN et de 3.98 m pour le MLP, soit un gain en erreur moyenne du GRNN, d'environ 6% par rapport au KNN et d'environ 33% par rapport au réseau MLP. De ce fait, le réseau GRNN a été choisi pour être utilisé dans la phase de filtrage comme algorithme de jumelage pour la technique de fingerprinting. D'un autre côté, cette phase nous a permis de constater que le fait d'augmenter le nombre de points d'accès dans l'environnement d'application, va avoir un effet sur l'amélioration de la précision de localisation du mobile.

Finalement, la troisième phase consistait à appliquer du filtrage au système de localisation à base de la technique de fingerprinting, pour améliorer les performances de localisation et de suivi. Cette phase a permis de réduire les erreurs d'estimation faites par la technique de fingerprinting, et aussi de ne pas avoir des discontinuités sur la trajectoire de suivi du mobile. Dans cette phase de filtrage, trois modèles de filtrage ont été considérés (linéaire, non-linéaire et linéarisé). Le modèle de filtrage linéaire était composé de quatre états, modélisant le mouvement linéaire en 2D d'un piéton, et de deux observations, correspondant à la position estimée par le réseau GRNN. Le modèle de filtrage non-linéaire était obtenu après avoir introduit dans le modèle de filtrage linéaire l'observation sur l'angle d'orientation du mobile, qui a été mesurée à l'aide d'une boussole. Finalement, le modèle de filtrage linéarisé a été obtenu en remplaçant dans le deuxième modèle de filtrage, la variable de l'angle d'orientation par les deux variables de vitesses axiales du mobile. Les résultats obtenus dans le cas des trois modèles de filtrage, ont permis de valider l'approche de filtrage qui a été proposée, où les performances de localisation et de suivi obtenues après la phase de filtrage étaient meilleures par rapport à celles obtenues par la technique de fingerprinting à base du réseau GRNN. Les améliorations

apportées sur la précision de localisation du mobile, en termes d'ereur moyenne, étaient respectivement d'environ 50% et 32% après l'application des filtres PF et KF au réseau GRNN dans le cas du modèle de filtrage linéarisé, d'environ 39% et 33% après l'application des filtres PF et EKF au réseau GRNN dans le cas du modèle non-linéaire, et d'environ 22% après l'application des filtres KF au réseau GRNN dans le cas du modèle linéaire. De même, les trajectoires de suivi du mobile obtenues après la phase de filtrage étaient bien orientées et plus proches de la trajectoire réelle, que celle obtenue par le réseau GRNN. D'un autre coté, cette phase de filtrage a montré l'intérêt d'avoir une richesse d'observations sur la dynamique du mobile, où le fait d'avoir dans le modèle de filtrage du système l'angle d'orientation du mobile, les performances de localisation et de suivi ont été nettement améliorées. Toutefois, pour bien appliquer cette phase de filtrage, il faut bien caractériser les relations des variables d'états et d'observation et bien dimensionner a priori certains paramètres de filtrage, dont les matrices de covariance des bruits d'état et de mesures, le choix du temps d'échantillonnage, en concordance avec le temps de traitement de données, ainsi que le nombre de particules dans le cas du filtre de particules.

5.1 Améliorations envisageables et travaux futurs

Comme améliorations envisageables et travaux futurs qui découlent de ce présent travail, il est souhaitable de :

- Observer de nouvelles informations sur le mouvement du mobile, comme par exemples, la variation de la vitesse tangentielle du mobile en utilisant un accéléromètre, ou la rotation du mobile en utilisant un gyroscope. Ceci permettra de mieux modéliser le comportement réel du mobile et ainsi d'améliorer la localisation et le suivi de ce dernier.

- Examiner d'autres algorithmes de localisation, comme par exemple, la méthode SVM (Support Vector Machine). Cette dernière combine entre les techniques statistiques et les techniques d'apprentissage des réseaux de neurones (Perceptron) pour modéliser le comportement non-linéaire entre les signatures RSS et la position du mobile, en se basant sur la redescription de l'espace non-linéaire en un espace linéaire de dimension supérieure.

- Utiliser d'autres technologies de détection pour des fins de localisation dans un environnement, comme par exemple, la technologie ultra-large bande (UWB : "Ultra-Wide Band"). Toutefois, il est peut être plus intéressant d'utiliser dans ce cas, l'approche temporelle (TOA ou TDOA). En effet, comme les systèmes UWB sont basés sur l'émission d'impulsions de largeur très faible, il est possible d'avoir une grande résolution temporelle. Ce qui est intéressant pour la mesure des temps TOA ou TDOA.

- Introduire dans le système de localisation un plan électronique de l'environnent de localisation avec un système à interface graphique. Ce plan va permettre de mieux cerner la trajectoire du mobile et d'éviter de localiser le mobile dans des zones où il n'est pas possible d'y accéder ou de franchir.

- Implanter un système de localisation sur un circuit numérique, tel que le circuit programmable FPGA, où avec ce circuit, on peut implanter le concept de localisation et de le rendre plus accessible à l'usage publique. En plus, ces circuits ont l'avantage d'avoir une vitesse de traitement largement supérieure à celle d'un ordinateur portable ou d'un PDA.

- Appliquer l'approche proposée dans ce mémoire au cas d'un robot mobile, où le modèle de filtrage de ce dernier sera un peu plus compliqué, mais a priori connu et contrôlé, et voir ainsi son effet sur les performances de localisation.

BIBLIOGRAPHIE

[1] P. Bahl, V.H. Padmanabhan, "RADAR : an in-building RF-based user location and tracking system", Proceedings IEEE Infocom 2000, Tel Aviv, Israel, vol. 2, pp. 775-784, Mar. 2000.

[2] Y.Chen, H. Kobayashi, "Signal Based Indoor geolocation", Proc. IEEE International Conference on Communications, April-May 2002, New York.

[3] J.P. Makela K. Pahlavan, X. Li, "Indoor Geolocation Science and Technology", IEEE Communications Magazine, pages 98-104, vol. 40, no. 2, Feb 2002.

[4] Présentation du système GPS, website : http ://www.engref.fr/gpsA.htm.

[5] A. M. Ladd, K. E. Bekris, G. Marceau, A. Rudys, L. E. Kavraki, and D. S. Wallach, "Robotics-based location sensing using wireless ethernet", in Proc. ACM International Conference on Mobile Computing and Networking (MOBICOM'02), 2002, pp. 227-238.

[6] P. Krishnamurthy, "Position location in mobile environments", in Proc. NSF Workshopon Context-Aware Mobile Database Management (CAMM), Providence, RI, Jan. 2002.

[7] T. S. Rappaport, J. H. Reed, and B. D. Woerner, "Position location using wireless communications on highways of the future," IEEE Commun. Mag., vol. 34, no. 10, pp. 33-41, Oct. 1996.

[8] K. Pahlavan, X. Li, and J. P. Makela, "Indoor geolocation science and technology", IEEE Commun. Mag., vol. 40, no. 2, pp. 112-118, Feb. 2002.

[9] R. Battiti, M. Brunato, and A. Villani, "Statistical learning theory for location fingerprinting in wireless LANS," Technical Report, Oct. 2002.

[10] T. Roos, P. Myllymaki, H. Tirri, P. Misikangas, and J. Sievanen, "A probabilistic approach to wlan user location estimation," International Journal of Wireless information Networks, vol. 9, no. 3, pp. 155-164, July 2002.

[11] K. Kaemarungsi, "Design of indoor positioning systems based on location fingerprinting technique", University of Pittsburgh, 2005.

[12] J. A. Tauber, "Location systems for pervasive computing", Area Exam Report, Massachusetts Institute of Technology, Aug. 2002.

[13] R. WANT, A. HOPPER et al. "The Active badge Location System", Olivertti Research Ltd, Cambridge, 1992.

[14] A. Harter, A. Hopper, P. Steggles, A.Ward, and P.Webster, "The anatomy of a context-aware application", in Proc. ACM International Conference on Mobile Computing and Networking (MOBICOM'99), Seattle, WA, Aug. 1999.

[15] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system", in Proc. ACM International Conference on Mobile Computing and Networking (MOBICOM'00), Boston, MA, Aug. 2000, pp. 32-43.

[16] J. Hightower and G. Borriello, "Location systems for ubiquitous computing", IEEEComputer, vol. 34, no. 8, pp. 57-66, Aug. 2001.

[17] C. Nerguizian, C. Despins et S. Affes, "A Framework for Indoor Geolocation using an Intelligent System", INRS-Télécommunications, 3rd WLAN Workshop 2001.

[18] V. Zeimpekis, G. M. Giaglis, and G. Lekakos, "A taxonomy of indoor and outdoor positioning techniques for mobile location services", Journal of ACM Special Interest Group on Electronic Commerce, vol. 3, pp. 19-27, 2002.

[19] IEEE std 802.11-1997 Information Technology-telecommunications And Information exchange Between Systems-Local And Metropolitan Area Networks-specific Requirements-part 11 : Wireless Lan Medium Access Control (MAC) And Physical Layer (PHY) Specifications.

[20] L. M. Ni, Y. Liu, Y. C. Lau and A. P. Patil, "LANDMARC : Indoor Location Sensing Using Active RFID", Wireless Networks, v.10 n.6, p.701-710, November 2004.

[21] G.I. Wassi, D. Grenier, C. Nerguizian, C. Despins, "Indoor Location using Received Signal Strengh of IEEE 802.11b Access Point", in CCECE, Saskatoon (SA), May 2005.

[22] J. Geier, "Understanding 802.11 Frame types", Wi-Fi Panet, August 2002.

[23] J. Small, A. Smailagic, and D. P. Siewiorek, "Determining user location for context aware computing through the use of a wireless lan infrastructure", Online, Dec. 2000.

[24] M. A. Youssef, A. Agrawala, and A. U. Shankar, "WLAN location determination via clustering and probability distributions", in Proc. IEEE International Conference on Pervasive Computing and Communications (PerCom'03), Dallas-Fort Worth, TX, Mar. 23-26, 2003, pp. 23-26.

[25] S. Saha, K. Chaudhuri, D. Sanghi, and P. Bhagwat, "Location determination of a mobile device using ieee 802.11b access point signals", in Proc. IEEE Wireless Communications and Networking Conference (WCNC'03), New Orleans, LA, Mar. 2003, pp. 1987-1992.

[26] C. Nerguizian, L. Hamza, V. Nerguizian and M. Saad, "3D Indoor Geolocation With Received Signal Strength Fingerprinting Technique And Neural Networks", 4th WSEAS International Conference on Electronics, Signal Processing and Control (ESPOCO 2005).

[27] R. Battiti, A. Villani and T. L. Nhat, "Neural Network Models for Intelligent Networks : Deriving the Location from Signal Patterns", Autonomous Intelligent Networks and Systems, UCLA, Los Angelos, May 2002.

[28] Chahé Nerguizian, "Radiolocalisation en milieu minier souterrain", Thèse de doctorat, INR-SEMT, Québec, Août 2003.

[29] Claude Touzet, "Les Réseaux de neurones artificiels : Introduction au connexionnisme, Cours, Exercices et travaux pratiques", France, Juillet 1992.

[30] D. F. Specht, "A general regression neural network", IEEE Trans. Neural Networks, vol. 2, pp. 568-576, November 1991.

[31] R. Hachad, "Caractérisation à bande étroite d'un canal de propagation en environnement interne à 2.4 GHz", Mémoire de Maîtrise en Ingénierie, École Polytechnique de Montréal, 2007.

[32] Site Web : http ://www.netstumbler.com/.

[33] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems", Transaction of the ASME-Journal of Basic Engineering, pp. 35-45, March 1960.

[34] D. Alazard, "Introduction au Filtrage de Kalman", Notes de cours, Exercices corriges, Sessions Matlab, SUPAERO France, Janvier 2005.

[35] P. S. Maybeck, "The Kalman Filter : An Introduction to Concepts", in Autonomous Robot Vehciles, I.J. Cox, G. T.Wilfong (eds), Springer-Verlag, 1990.

[36] G. Welch and G. Bishop, "An Introduction to the Kalman Filter", University of North Carolina, Department of Computer Science, 1995.

[37] M. S. Grewal and A. P. Andrews, "Kalman Filtering - Theory and Practice", Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1993.

[38] Y. Bar-Shalom and X. R. Li. "Estimation and Tracking - Principles, Techniques and Software", Artech House, 1993.

[39] A. Doucet, J.F.G. De Freitas and N.J. Gordon "Sequential Monte Carlo Methods in Practice", Springer Series in Statistics for Engineering and Information Science, New York : Springer-Verlag, 2001.

[40] A Doucet, C Andrieu and S. Godsill, "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering", Statistics and Computing, vol. 10, no. 3, pp. 197-208, 2000.

[41] A. Doucet, S.J. Godsill, and M. West. "Monte Carlo filtering and smoothing with application to time-varying spectral estimation", in Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., Vol. 2 :701-704, 2000.

[42] G. Kitagawa, "Monte Carlo Filter and Smoother for Non-Gaussian Non-Linear State Space Models", Journal of Computational and Graphical Statistics, Vol. 5(No.1), pp.1-25, 1996.

[43] Site Web : http ://www.silabs.com/tgwWebApp/public/web_content/products/Microcontrollers/en/F350-COMPASS-RD.htm.

[44] K. Dutton, S. Thompson, B. Barraclough. "The Art of Control Engineering", ADDISON-WESLEY, ISBN 0-201-17545-2, 1997.

ANNEXE I

PROGRAMMES MATLAB

I.1 Programme de construction de la base de données des signatures

```
%=========================================================================
%Programme permettant de lire et d'enregistrer dans un fichier sous forme
%de base de données les puissances RSS (en dBm), à partir des fichiers de
%mesures sous le format '.tex', fournis par l'outil NetStumbler, où chaque
%fichier correspond à une point de mesure. Le nombre d'échantillons relevés
%en chaque point de mesures est de 60, qui correspond à un temps de 30 s
%(soit 0.5 s par échantillon). Cependant, les 10 premiers et derniers
%échantillons ne sont pas pris en considération.
%=========================================================================
clear all; clc;

%Initialisation des indices des fichiers de mesures sauvegardés :
ind_debut = input('Entrer l'indice du premier fichier de mesures =');
ind_fin = input('Entrer l'indice du dernier fichier de mesures =');
rejet = 10*6; %Nombre d'échantillons de mesures à rejer (=10), où 6
              %correspond au nombre d'APs détéctés.

for ind=ind_debut:1:ind_fin
    %Lecture des données contenues dans chaque fichier de mesures :
    fichier=strcat(['E:\Projet\Mesures\Netstumbler\' num2str(ind) '.txt']);
    text=textread(fichier,'%s', 'delimiter', '');
    raw=char(text);
    [N,M]=size(raw);
    n=0;
    for i=5:1:N
        if raw(i,27:32)=='hermes' %Identification des APs correspondants au
                                  %réseau sans fil 'hermes' déployé.
           n=n+1; %Indice calculant le nombre d'échantilllons de mesures
                  %pour tout les APs du réseau hermes, en un potion donnée.
           id(n,:)=raw(i,40:60);  %Identification de l'adresse MAC de
                                  %chaque AP détecté.
           sig=raw(i,78:86);
           c_num=str2num(sig);
           sig_num(n)=c_num(2);   %Identification de la donnée RSS pour
                                  %chaque échantillon de mesure.
```

```
        end
    end

    %Fichier contenant les adresses MACs des 48 APs du réseau hermes.
    [num,texte]=xlsread('E:\Projet\Mesures\AP_MAC.xls');
    id_ref=char(texte(2:49));
    rss(1:1:48)=-100; %Initialisation des puissances RSS à la valeur
                      %-100dBm, qui correspond à la valeur RSS assignée au
                      %AP non détecté par la carte sans fil du portable, en
                      %un point de mesure donné.
    for j=1:48        %Itérations suivant le nombre total des APs (=48).
        m=0;
        for k=rejet+1:n-rejet %Itérations suivant le nombre d'échantillons
                              %de mesures considérés pour l'enrigistrement.
            if id_ref(j,:)==id(k,:) & sig_num(k)>0
               m=m+1; %Nombre d'échantillons de mesures pour chaque AP, ce
                      %nombre oscille aux alentours de 60.
               sig_vect(j,m)=sig_num(k)-149; %Correction des puissances RSS
                                             %à leurs vraies valeurs.
               rss(j)=mean(sig_vect(j,1:m)); %Calcul des moyennes RSS.
            end
        end
    end
    %Enregistrement des signatures RSS dans un fichier comme base de
    %données de localisation :
    xlswrite('E:\Projet\Mesures\Database.xls',rss)
end
%=========================================================================
```

I.2 Programme de l'algorithme KNN adapté au cas de la localisation

```
function xy=KNN(RSS_tr,xy_tr,RSS_ts,K)
%=========================================================================
%Fonction définissant la méthode des K-plus proches voisins (KNN : K-Nearest
%Neighbors) adaptée au cas de la localisation.
%=========================================================================

%Entrées :
%--------
%RSS_tr : données des signatures d'entraînement.
%xy_tr  : données des positions d'entraînement.
%RSS_ts : données des signatures RSS actuelles.
%K      : nombre de voisins les plus proches.

%Sortie :
%-------
```

```
%xy : position (x, y) estimée par le KNN.

n=size(RSS_ts,2);
for i=1:n
    %Déterminer les K signatures les plus proches de la signature.
    %actuelle du mobile à localiser.
    K_Voisins=K_Neighbors(RSS_tr,RSS_ts(:,i),K); %Indices des K signatures
              %les plus proches ainsi que les distances Euclidienne qui
              %les séparent de la signature actuelle du mobile.
    num=[0;0];
    den=0;
    eps=0.0001;
    %Utiliser les positions correspondantes aux K signatures les plus
    %pour estimer la position actuelle du mobile.
    for j=1:K
       h=K_Voisins(1,j); %Indice de la j-ème signature la plus proche.
       d=K_Voisins(2,j); %Distance Euclidienne qui sépare la signature
                         %j et celle actuelle du mobile.
       num=num+(1/(d+eps))*xy_tr(:,h);
       den=den+(1/(d+eps));
    end
    position(:,i)=num/den; %Position (x, y) estimée du mobile.
end xy=position;

%=========================================================================
function K_Voisins=K_Neighbors(RSS_tr,RSS_ts,K)
%Fonction qui permet d'identifie les K-plus proches signatures par rapport
% à celle actuelle du mobile.
%=========================================================================

%Entrées :
%--------
%RSS_tr : données des signatures d'entraînement.
%RSS_ts : données des signatures RSS actuelles.
%K      : nombre de voisins les plus proches.

%Sortie :
%-------
%K_Voisins : les K signatures les plus proches à celle actuelle du mobile.

n=size(RSS_tr,2); if K>n
   dip('Erreur : K doit être inférieur ou égale à n')
end A=ones(1,n); A=RSS_ts*A; dist=sum((RSS_tr-A).^2);
dist=sqrt(dist); %Distance Euclidienne d'ordre 2 entre la signature
                 %actuelle du mobile et celles des données d'entraînement.
max_d=max(dist);

%Déterminer les indices des K plus proches signatures de celle actuelle :
for i=1:K
```

```
    min_d= min(dist);
    indice=find(dist==min_d);
    indice=min(indice);
    z(1,i)=indice; %Indice de la i-ème signature la plus proche.
    z(2,i)=dist(indice); %Distance euclidienne qui sépare la signature
                         %i et celle actuelle du mobile.
    dist(indice)=max_d+1;
end K_Voisins=z;
%=========================================================================
```

I.3 Programme d'entraînement du réseau MLP adapté au cas de la localisation

```
%=========================================================================
%Algorithme d'entraînement off-line du réseau de neurones multicouche (MLP:
%Multi-layer Perceptron), adapté au cas de la localisation.
%=========================================================================
clc; close all;   clear all;

N_AP = 6; %Nombre de points d'accès
input_tr=xlsread('E:\Projet\Mesures\Database.xls'); %Base de données
                                                    %d'entraînement
[n,m]=size(input_tr);
RSS_tr=input_tr(1:n,4:N_AP+3)'; %Donées des signatures d'entraînement.
xy_tr=input_tr(1:n,1:2)'; %Donées des positions correspondantes aux
                          %signatures d'entraînement.

[rss, minmax_rss]=mapminmax(RSS_tr); %Normalisation des signatures.
[xy, minmax_xy]=mapminmax(xy_tr); %Normalisation des positions.

%Définition de l'architecture du réseau MLP :
net = newff(minmax(rss),[10 5 2],{'tansig','tansig', 'purline'},'trainrp');
net.trainParam.show=100; net.trainParam.epochs =5000;
net.trainParam.goal=1e-4;

%Entraînement du réseau MLP :
net_mlp=train(net,rss,xy);

%Sauvegarde des paramètres d'entraînement du réseau MLP :
save MLP_Training net_mlp minmax_rss minmax_xy N_AP

%Calcul de l'erreur d'estimation commise sur les données d'entraînement :
ann=sim(net_mlp,rss);
xy=mapminmax('reverse',ann,minmax_xy); %(x,y) estimées par le réseau MLP.
err_xy=sqrt((xy_tr(1,:)-xy(1,:)).^2+(xy_tr(2,:)-xy(2,:)).^2);
```

```
mean_xy=mean(err_xy) %erreur moyenne.
%=========================================================================
```

I.4 Programme d'entraînement du réseau GRNN adapté au cas de la localisation

```
%=========================================================================
%Algorithme d'entraînement off-line du réseau de neurones à régression
%généralisée (GRNN : General Regression Neural Network), adapté au cas de
%la localisation.
%=========================================================================
clc; close all;   clear all;

N_AP = 6; %Nombre de points d'accés.
Spread = 8; %Valeur du paramètre spread.
input_tr=xlsread('E:\Projet\Mesures\Database.xls'); %Base de données
                                                    %d'entraînement
[n,m]=size(input_tr);
RSS_tr=input_tr(1:n,4:N_AP+3)'; %Donées des signatures d'entraînement.
xy_tr=input_tr(1:n,1:2)'; %Donées des positions correspondantes aux
                          %signatures d'entraînement.

%Définition de l'architecture du réseau GRNN :
net_grnn = newgrnn(RSS_tr,xy_tr,Spread);

%Sauvegarde des paramètres d'entraînement du réseau GRNN :
save GRNN_Training net_grnn N_AP Spread

%Calcul de l'erreur d'estimation commise sur les données d'entraînement :
xy=sim(net_grnn,RSS_tr); %Position (x,y) estimée par le réseau GRNN.
err_xy=sqrt((xy_tr(1,:)-xy(1,:)).^2+(xy_tr(2,:)-xy(2,:)).^2);
mean_xy=mean(err_xy) %erreur moyenne.
%=========================================================================
```

I.5 Programme du filtre de Kalman adapté au cas de la localisation et du suivi

```
%=========================================================================
function [x_kf,t_kf] = Kalman_Filter(F,Q,z,H,R,os)
%Fonction définissant l'algorithme du filtre de Kalman (KF : Kalman
```

```
%Filter), adapté au cas de la localisation, et permettant de retourner
%l'état estimé et son temps de traitement.
%=========================================================================

%Entrées :
%--------
%z : vecteur d'observation contenant la position estimée et/ou l'angle
    %d'orientation et/ou les vitesses axiales.
%Q : matrice de covariance du bruit d'état.
%R : matrice de covariance du bruit d'observation.
%F : matrice d'état.
%H : matrice d'observation.
%os : dimension du vecteur d'observation sélectionnant en même temps le
      %type modèle de filtrage à utiliser : os=2 ~ modèle linéaire, os=3 ~
      %modèle non-linéaire et os=4 ~ modèle linéarisé.

%Sorties :
%--------
%x_kf : état du système estimé a postriori.
%t_kf : temps mis par le filtre KF pour traiter une estimation donnée.

%Initialisation :
ss = size(Q,1);
N  = size(z,2);
x_kf=zeros(ss,N); x_kf(1:2,1) = z(1:2,1);
P = zeros(ss,ss,N);

tic; %Déclenchement du compteur de temps de traitement.
for t=2:N
    clc;
    fprintf('model = %i : KF : t = %i / %i  \r',os,t,N);
    fprintf('\n')

    %Phase de prédiction :
    xpred = F*x_kf(:,t-1); %Estimation a priori de l'état du système.
    Ppred = F*P(:,:,t-1)*F' + Q; %Matrice de covariance de l'erreur
                                 %d'estimation a priori.

    %Phase de correction :
    K = Ppred*H'*inv(H*Ppred*H' + R); %Gain de Kalman;
    x_kf(:,t) = xpred + K*(z(:,t) - H*xpred); %Estimation a posteriori de
                                              %l'état du système.
    P(:,:,t) = (eye(ss) - K*H)*Ppred; %Matrice de covariance de l'erreur
                                      %d'estimation a posteriori.
end
t_kf = toc/N; %Temps de traitement moyen.
%=========================================================================
```

I.6 Programme du filtre de Kalman étendu adapté au cas de la localisation et du suivi

```
function [x_ekf,t_ekf] =Extended_Filter(Q,z,os,R,T,V,F,H)
%=========================================================================
%Fonction définissant l'algorithme du filtre Kalman étendu (EKF : Extended
%Kalman Filter), adapté au cas de la localisation et permettant de
%retourner l'état estimé et son temps de traitement.
%=========================================================================

%Entrées :
%--------
%z : Vecteur d'observation contenant la position estimée et/ou l'angle
     %d'orientation et/ou les vitesses axiales du mobile.
%Q : Matrice de covariance du bruit d'état
%R : Matrice de covariance du bruit d'observation.
%V : Vitesse tangentielle du mobile supposée constante.
%T : Temps ou pas d'échantillonnage.
%os: dimension du vecteur d'observation sélectionnant en même temps le
      %type modèle de filtrage à utiliser : os=2 ~ modèle linéaire, os=3 ~
      %modèle non-linéaire et os=4 ~ modèle linéarisé.
%F : matrice d'état.
%H : matrice d'observation.

%Sorties :
%--------
%x_ekf : état du système estimé a postriori.
%t_ekf : temps consommé par le filtre EKF pour une estimation donnée.

%Initialisation :
%---------------
ss = size(Q,1);
N  = size(z,2);
x_ekf = zeros(ss,N); x_ekf(1:3,1) = z(1:3,1);
P  = zeros(ss,ss,N); %Matrice de covariance d'erreur a piori.

tic; %Déclenchement du compteur de temps de traitement.
for t=2:N
    clc;
    fprintf('model = %i : EKF : t = %i / %i  \r',os,t,N);
    fprintf('\n')
    x_ekf(3,t) = z(3,t);

    %Phase de prédiction :
    x_Pred = feval('ffun',x_ekf(:,t-1),T,V,os,F); %Estimation a priori de
                                                  %l'état du système.
    Jx = [ 1 0 -T*V*sind(x_Pred(3));0 1 T*V*cosd(x_Pred(3));0 0 1];%Matrice
```

```
                                                    %Jacobienne de ffun.
    PPred = Q + Jx*P(:,:,t-1)*Jx'; %Matrice de covariance de l'erreur
                                   %d'estimation a priori.

    %Phase de correction :
    zPred = feval('hfun',x_Pred,os,H); %Observation prédite.
    Jz = eye(os,os); %Matrice Jacobienne de hfun.
    K = PPred*Jz'*inv(R + Jz*PPred*Jz'); % Gain de Kalman.
    x_ekf(:,t) = x_Pred + K *(z(:,t)-zPred); %Estimation a posteriori de
                                             %l'état du système.
    P(:,:,t) = PPred - K*Jz*PPred; %Matrice de covariance de l'erreur
                                   %d'estimation a posteriori.
end
t_ekf = toc/N; %Temps de traitement moyen.
%=========================================================================
```

I.7 Programme du filtre de particules adapté au cas de la localisation et du suivi

```
%=========================================================================
function [x_pf,t_pf] = Particle_Filter(F,ax,ay,q,H,z,os,R,N_P,T,V,no_runs)
%Fonction définissant l'algorithme de filtrage de particules (PF : Particle
%Filter) de type SIR (Sampling Importance Resampling), adapté au cas de la
%localisation, et permettant de retourner l'état estimé et son temps de
%traitement.
%=========================================================================

%Entrées :
%--------
%z : vecteur d'observation.
%F : matrice d'état.
%H : matrice d'observation.
%q : matrice définissant le bruit d'état du filtre de particules.
%os : dimension du vecteur d'observation sélectionnant en même temps le
      %type modèle de filtrage à utiliser : os=2 ~ modèle linéaire, os=3 ~
      %modèle non-linéaire et os=4 ~ modèle linéarisé.
%R : matrice de covariance du bruit d'observation.
%ax et ay : accélérations axiales du mobile suivant x et y définissant le
           %bruit d'état ou de processus.
%N_P : nombre de particules.
%T : temps d'échantillonnage.
%V : vitesse tangentielle du mobile supposée constante.
%no_runs : nombre de simulations faites pour avoir une estimation donnée.

%Sorties :
```

```
%x_pf : état du système estimé a posteriori.
%t_pf : temps mis par le filtre PF pour traiter une estimation donnée.

%Initialisation :
N=size(z,2); %Nombre de données de test.
ss=size(q,1); %Dimension de l'espace d'état.
xPred_pf=zeros(ss,N,N_P); % Vecteur d'état prédit a priori.
zPred_pf=zeros(os,N,N_P); % Vecteur d'état prédit a priori.
w = zeros(N,N_P);           % Poids d'importance.
x_particle = zeros(ss,N,N_P); % Etat estimé pour chaque run (simulation).
x_particle(1,1,:)=z(1,1); x_particle(2,1,:)=z(2,1);

 for j=1:no_runs
    tic; %Déclenchement du compteur de temps de traitement.
    for t=2:N
        if os==3
           x_particle(3,t,:)=z(3,t);
        end
        clc;
        fprintf('model=% i :run=%i/%i: PF :t=%i/%i\r',os,j,no_runs,t,N);
        fprintf('\n')

        %Phase de prédiction :
        for i=1:N_P
            %Bruit d'état :
            if os==3
            noise_q=q*[ax*randn;ax*randn;10^2*randn];
            else
            noise_q=q*[ax*randn;ax*randn];
            end
            xPred_pf(1:ss,t,i)=feval('ffun',x_particle(1:ss,t-1,i),T,...
                               V,os,F)+ noise_q;
        end

        %Phase d'évaluation des poids d'importance :
        for i=1:N_P
            zPred_pf(:,t,i) = feval('hfun',xPred_pf(:,t,i),os,H);
            err_z=z(:,t,1)-zPred_pf(:,t,i);
            w(t,i)=w(t-1,i)*(abs(2*pi*det(R)))^(-os/2)*exp(-0.5*(err_z)'...
                   *pinv(R)*(err_z))+ 1e-99;
        end

        %Normalisation des poids d'importance :
        w(t,:)=w(t,:)./sum(w(t,:));

        %Phase de rééchantillonnage :
        [w(t,:),new_index]=resample(w(t,:),N_P); %Rééchantillonnage par la
                                  %fonction resample de type systématique.
        x_particle(:,t,:)=xPred_pf(:,t,new_index); %Assigner les nouveaux
```

```
                                                    %indices aux estimées.
    end
x_run(:,:,j)=mean(x_particle,3); %Moyennes des estimées par rapport au
                                 %nombre de particules.
temps(j) = toc; end
x_pf=mean(x_run(:,:,j),3); %Moyennes des estimées par rapport au nombre
                           %de runs.
t_pf = mean(temps)/N; %Temps de traitement par estimation.

%=========================================================================
function [w_new,new_index]=resample(w,N_P)
%Fonction de rééchantillonnage systématique proposée par G. Kitagawa.
%=========================================================================

%Entrée :
%-------
%w : poids d'importance normalisés.
%N_P : nombre de particules.

%Sortie :
%-------
%w_new : poids d'importance apès rééchantillonnage.
%new_index : nouveaux indices après rééchantillonnage.

cdf = cumsum(w); %Fonction cumulée des poids d'importance normalisé.
u1 = (1/N_P)*rand; %Variable alétoire de départ suivant une loi uniforme.
i = 1; for j = 1:N_P
    u = u1 + (j-1)/N_P;
    while (u > cdf(i))
        i = i + 1;
    end
    w_new(j)=1/N_P;
    new_index(j) = i;
end return
%=========================================================================
```

I.8 Programme fournissant les paramètres de localisation et de filtrage dans le cas des trois modèles de filtrage considérés

```
%=========================================================================
function [z,x,Q,q,R,F,H,t_obs] = System(os,V,fing_select,N_AP,T,ax,ay)
%Fonction qui fournit les différents paramètres de filtrage pour les trois
%modèles du sytème considérés : linéaire, non-linéaire et linéarisé.
%=========================================================================
```

```
%Entrées :
%--------
%os : dimension du vecteur d'observation sélectionnant en même temps le
      %type modèle de filtrage à utiliser : os=2 ~ modèle linéaire, os=3 ~
      %modèle non-linéaire et os=4 ~ modèle linéarisé.
%V : vitesse tangentielle du mobile supposée constante.
%fing_select : indice de selection de l'algorithme de fingerprinting
               %utilisé : fing_select=1 ~ KNN, fing_select=2 ~ MLP et
               %fing_select=3 ~ GRNN.
%N_AP : nombre de points d'accès (AP) utilisés
%T : temps ou pas d'échantillonnage.
%ax et ay : accélérations axiales du mobile suivant x et y définissant le
            %bruit d'état ou de processus.

%Sorties :
%--------
%z : vecteur d'observation contenant la position estimée et/ou l'angle
    %d'orientation et/ou les vitesses axiales.
%x : vecteur position réel des données de test.
%Q : matrice de covariance du bruit d'état.
%q : matrice contenant les paramètres définissant le bruit d'état.
%R : matrice de covariance du bruit d'observation.
%F : matrice d'état.
%H : matrice d'observation.
%t_obs : temps consommé par l'algorithme de fingerprinting pour traiter
         %une estimation donnée.

tic; %Déclenchement du compteur de temps de traitement.

%Base de données d'entraînement :
input_tr=xlsread('E:\Projet\Mesures\Database.xls');
[n,m]=size(input_tr);
RSS_tr=input_tr(1:n,4:N_AP+3)';
xy_tr=input_tr(1:n,1:2)';

%Base de données de test :
input_ts=xlsread('E:\Projet\Mesures\Tracking.xls');
[N,M]=size(input_ts);
RSS_ts=input_ts(1:N,5:N_AP+4)';
x=input_ts(1:N,1:2)';

%Estimation de la position de localisation du mobile :
if fing_select==1
   %Estimation de la position du mobile par l'algorithme KNN :
   K = N_AP;
   xy_ts=KNN(RSS_tr,xy_tr,RSS_ts,K);

elseif fing_select==2
   %Estimation de la position par l'algorithme MLP :
```

```
   load MLP_Training;
   rss_ts= mapminmax('apply',RSS_ts,minmax_rss);
   ann=sim(net,rss_ts);
   xy_ts=mapminmax('reverse',ann,minmax_xy);

elseif fing_select==3
   %Estimation de la position du mobile par l'algorithme GRNN :
   load GRNN_Training;
   xy_ts=sim(net_grnn,RSS_ts);
end

%Définition des paramètres de filtrage pour les trois modèles considérés :
if os==2
   %Paramètres du modèle de filtrage linéaire :
   F = [1 0 T 0; 0 1 0 T; 0 0 1 0; 0 0 0 1];
   H = [1 0 0 0; 0 1 0 0 ];
   R = [3.5^2 0;0 3.5^2 ];
   Q = [(ax^2)*T^4/4 0 (ax^2)*T^3/2 0;0 (ay^2)*T^4/4 0 (ay^2)*T^3/2;...
       (ax^2)*T^3/2 0 (ax^2)*T^2 0;0 (ay^2)*T^3/2 0 (ay^2)*T^2];
   q=  [T^2/2 0;0 T^2/2;T 0;0 T];
   z=xy_ts(1:2,:);

elseif os==3
   %Paramètres du modèle de filtrage non-linéaire :
   F=[];H=[];
   R = [3.5^2 0 0;0 3.5^2 0;0 0 0];
   Q =[(ax^2)*T^4/4 0 0;0 (ay^2)*T^4/4 0;0 0 10^4];
   q= [T^2/2 0 0;0 T^2/2 0;0 0 10^2];
   z(1:2,:)=xy_ts(1:2,:);
   z(3,2:N)=input_ts(1:N-1,4)'; %Les observations de l'angle d'orientation.

elseif os==4
   %Paramètres du modèle de filtrage linéarisé :
   F = [1 0 T 0; 0 1 0 T; 0 0 1 0; 0 0 0 10];
   H = [1 0 0 0; 0 1 0 0 ;0 0 1 0 ;0 0 0 1];
   Q = [(ax^2)*T^4/4 0 (ax^2)*T^3/2 0;0 (ay^2)*T^4/4 0 (ay^2)*T^3/2;...
       (ax^2)*T^3/2 0 (ax^2)*T^2 0;0 (ay^2)*T^3/2 0 (ay^2)*T^2];
   q= [T^2/2 0;0 T^2/2;T 0;0 T];
   R = [3.5^2 0 0 0;0 3.5^2 0 0;0 0 0.01^2 0; 0 0 0 0.01^2];
   z(1:2,:)=xy_ts(1:2,:);

   %Mesures des vitesses suivant les axes x et y respectivement :
   z(3,2:N)=V.*cosd(input_ts(1:N-1,4)');
   z(4,2:N)=V.*sind(input_ts(1:N-1,4)');
end
t_obs = toc/N; %Temps de traitement moyen.

%=========================================================================
function xPred = ffun(x,T,V,os,F)
```

```
% Fonction d'état qui permet d'estimer a priori (prédire) l'état du sysème.
%=========================================================================

%Entrées :
%--------
%x : vecteur d'état du système.
%T : temps d'échantillonnage.
%V : vitesse tangentielle du mobile supposée constante
%os : dimension du vecteur d'observation sélectionnant en même temps le
      %type modèle de filtrage à utiliser : os=2 ~ modèle linéaire, os=3 ~
      %modèle non-linéaire et os=4 ~ modèle linéarisé.
%F : matrice d'état.

%Sortie :
%-------
%xPred : vecteur d'état estimé a priori.

if nargin < 5
    error('Pas assez d'arguments d'entrée');
end

if os==3
   xPred =[ x(1) + T*V*cosd(x(3))
            x(2) + T*V*sind(x(3))
            x(3) +       0       ];
else
    xPred =F*x;
end

%=========================================================================
function z = hfun(x,os,H)
% Fonction permettant de donner l'observation prédite à partir de la
%fonction d'observation du système.
%=========================================================================

%Entrées :
%--------
%x : vecteur d'état.
%os : dimension du vecteur d'observation sélectionnant en même temps le
      %type modèle de filtrage à utiliser : os=2 ~ modèle linéaire, os=3 ~
      %modèle non-linéaire et os=4 ~ modèle linéarisé.
%H : matrice d'observation.

%Sortie :
%-------
%z : observation estimée à l'étape de temps k.

if nargin < 3
    error('Pas assez d'arguments d'entrée');
```

```
end

if os==2
   z=H*x;
else
   z=x;
end
%=========================================================================
```

I.9 Programme permettant d'implémenter le processus de localisation et de suivi

```
%=========================================================================
%Ce programme permet d'implémenter le système de localisation basé sur
%l'application des filtres KF, EKF et PF à la technique de fingerprinting
%dont trois algorithmes de jumelage sont examinés : KNN, MLP et GRNN.
%=========================================================================
clear all;  close all; clc;

%Définition des paramètres de localisation et de filtrage :
%-----------------------------------------------------------%
fing_select=input('Sélectionner l'algorithme de fingerprint fing_select=');
           %fing_select=1 ~KNN, fing_select=2 ~MLP et fing_select=3 ~GRNN.
os=input('Sélectionner le modèle de filtrage, os='); %os=2 ~modèle linéaire,
           %os=3 ~modèle non-linéaire et os=4 ~modèle linéarisé.
filt_select=input('Sélectionner le filtre appliqué, filt_select =');
           %filt_select=1 ~KF, filt_select=2 ~EKF et filt_select=3 ~PF.
T=2; %Temps ou pas d'échantillonnage.
V=0.5; %Vitesse tangentielle du mobile supposée constante.
N_AP=6; %Nombre de points d'accès.
ax=0.05; ay=0.05; %Valeurs des accélérations axiales du mobile suivant x
                  %et y définissant le bruit d'état ou de processus.
N_P = 200; %Nombre de particules.
no_runs = 1; % Nombre de simulations par itération.

%Localisation et définition des modèles de filtrage :
%-----------------------------------------------------%
[z,x,Q,q,R,F,H,t_obs] = System(os,V,fing_select,N_AP,T,ax,ay);
%z : vecteur observation contenant la position estimée par l'algorithme de
    %fingerprinting et/ou l'angle d'orientation et/ou les vitesses axiales.
%x : vecteur position réel des données de test.
%Q : matrice de covariance du bruit d'état.
%q : matrice contenant les paramètres définissant le bruit d'état.
%R : matrice de covariance du bruit d'observation.
%F : matrice d'état.
```

```
%H : matrice d'observation.
%t_obs : temps consommé par l'algorithme de fingerprinting pour traiter
         %une estimation donnée.
x_fing=z(1:2,:); %Position estimée par l'algorithme de fingerprinting.

%Application du filtrage :
%-----------------------%
if filt_select==1
  %Filtre de Kalman (KF) :
  if  os~=3
      [x_kf,t_kf] = Kalman_Filter(F,Q,z,H,R,os);
      x_filt=x_kf;
      t_filt=t_kf;
  end
elseif filt_select==2
  % Filtre de Kalman étendu (EKF) :
  if  os==3
      [x_ekf,t_ekf] =Extended_Filter(Q,z,os,R,T,V,F,H);
      x_filt=x_ekf;
      t_filt=t_ekf;
  end
elseif filt_select==3
   % Filtre de particules (PF) :
   [x_pf,t_pf] = Particle_Filter(F,ax,ay,q,H,z,os,R,N_P,T,V,no_runs);
   x_filt=x_pf;
   t_filt=t_pf;
end

%Erreurs de localisation avant et après l'application du filtrage :
%-----------------------------------------------------------------%
disp('Erreurs de localisation par fingerprinting :');
disp('-----------------------------------------');
err_fing  = sqrt((x(1,:)-x_fing(1,:)).^2+(x(2,:)-x_fing(2,:)).^2); %Erreur
                                %moyenne en termes de distance Euclidienne.
mean_fing = mean(err_fing);%Erreur moyenne.
disp(['mean_fing     = ' num2str(mean_fing)]);
max_fing  = max(err_fing); %Maximum de l'erreur.
disp(['max_fing      = ' num2str(max_fing)]);
min_fing  = min(err_fing); %Minimum de l'erreur.
disp(['min_fing      = ' num2str(min_fing)]);
err_cdf_fing = prctile(err_fing,[25 50 70 95]); %Erreur en pourcentage CDF.

disp(['err_cdf_fing = ' num2str(err_cdf_fing)]);
t_fing=t_obs; %Temps moyen de traitement d'une estimation de localisation.
disp(['temps_fing   = ' num2str(t_fing)]); disp(' ');

disp('Erreurs de localisation après application du filtrage :');
disp('-----------------------------------------------------');
err_filt = sqrt((x(1,:)-x_filt(1,:)).^2+(x(2,:)-x_filt(2,:)).^2); %Erreur
```

```
                                          %après l'application du filtrage.
mean_filt = mean(err_filt);%Erreur moyenne après l'application du filtrage.

disp(['mean_filt     = ' num2str(mean_filt)]);
gain_filt = 100*abs(mse_filt-mse_fing)/mean_fing; %Gain en erreur moyenne
                     %après l'application du filtrage au fingerprinting.
disp(['gain_filt    = ' num2str(gain_filt)]);
max_filt = max(err_filt); %Maximum de l'erreur
                          %après l'application du filtrage.
disp(['max_filt     = ' num2str(max_filt)]);
min_filt = min(err_filt); %Minimum de l'erreur après l'application du
                          %filtrage.
disp(['min_filt     = ' num2str(min_filt)]);
err_cdf_filt = prctile(err_filt,[25 50 70 95]); %Erreur en pourcentage CDF
                                          %après l'application du filtrage.
disp(['err_cdf_filt = ' num2str(err_cdf_filt)]);
t_filt; %Temps de traitement d'une estimation pour le filtrage tout seul.
disp(['temps_filt   = ' num2str(t_filt)]);

%Courbes de l'erreur de localisation instantannée :
%-------------------------------------------------%
clf figure(1) hold on;
plot(err_fing,'r*'); %Courbe de l'erreur instannée du fingerprinting.
plot(err_filt,'-'); %Courbe de l'erreur instannée obtenue apès
                    %l'application du filtrage.
hold off
legend('Fingerprinting', 'Filtrage', 2)
xlabel('Points de mesures')
ylabel('Erreur (mètres)')
title('Erreur de localisation instantannée')

%Courbes de distribution cumulative (CDF) de l'erreur de localisation :
%---------------------------------------------------------------------%
figure(2) hold on;
cdfplot(err_fing); %Courbe CDF de l'erreur obtenue par  fingerprinting.
cdfplot(err_filt); %Courbe CDF de l'erreur obtenue apès filtrage.
hold off
legend('Fingerprinting', 'Filtrage', 2)
xlabel(Erreur de localisation')
ylabel('Probabilité d'erreur')
title('CDF de l'erreur de localisation')

%Courbes de la trajectoire de suivi du mobile :
%---------------------------------------------%
figure(3) hold on
plot(x(1,:), x(2,:), 'k-'); %Courbe de la trajectoire de suivi du mobile
                            % réelle.
plot(x_fing(1,:), x_fing(2,:), 'b*'); %Courbe de la trajectoire de suivi
                                %du mobile obtenue par le fingerprinting.
```

```
plot(x_filt(1,:), x_filt(2,:), 'r-'); %Courbe de la trajectoire de suivi
                        %du mobile obtenue après application du filtrage.
legend('Réelle', 'Fingerprinting', 'Filtrage', 2)
xlabel('x (m)')
ylabel('y (m)')
title('Trajectoire de suivi du mobile')
hold off
```

ANNEXE II

FOIRE AUX QUESTIONS

Q.1 Quel est l'objectif de ce mémoire ?

R.1 Améliorer les performances des systèmes de localisation et de suivi des mobiles en environnements internes.

Q.2 Quels sont les services que peuvent offrir les sytèmes de localisation et de suivi dans un environnement interne ?

R.2 Parmi ces services on cite par exemples :

- La localisation des soldats, des policiers et des pompiers pour l'accomplissement de leurs missions.
- La localisation dans les hôpitaux des médecins ou des patients et le contrôle des équipements à coûts élevés.
- La localisation et le suivi des robots.
- La visite guidée dans un musée ou dans un campus universitaire.
- La localisation des téléphones sans fil appellant un numéro d'urgence.
- La mise à disposition d'un plan d'étage ou d'indications descriptives de la localisation actuelle d'un utilisateur.

Q.3 Pourquoi les sytèmes GPS ne sont pas performants pour la localisation dans un environnement interne ?

R.3 À cause de la dégradation des signaux émis par les différents satellites dans ces milieux, due au phénomène de multitrajet et à la non-visibilité directe entre les satellites et le terminal GPS.

Q.4 Quel est le principe de base du système de localisation proposé dans ce mé-

moire ?

R.4 Il est basé sur la technologie sans fil WLAN et la technique de fingerprinting.

Q.5 Quels sont les avantages de la technique de localisation de fingerprinting ?

R.5 Cette technique est mieux adaptée aux environnements internes pour le fait qu'elle utilise le profil de propagation du signal par multitrajet comme information utile et ne requiert pas une visibilité directe entre l'émetteur et le récepteur.

Q.6 Quel est le principe de fonctionnement de la technique de fingerprinting ?

R.6 Le principe est constitué de deux phases. Une phase de collecte de données (enregistrement d'une base de données de signatures recueillies en différentes positions de localisation du mobile) et une phase d'estimation de la position de localisation actuelle du mobile, pendant laquelle un algorithme de jumelage est utilisé pour identifier la plus proche des signatures enregistrées dans la base de données par rapport à la signature actuelle du mobile et en déduire ainsi la position de localisation du mobile.

Q.7 Quelle est la raison du choix de l'information de signature RSS pour la technique de localisation de fingerprinting ?

R.7 La signature RSS peut être extraite facilement à partir des balises émises périodiquement par les points d'accès du réseau WLAN et reçues au niveau de la carte sans fils de l'équipement mobile. Aucune modification sur les équipements ne doit être effectuée.

Q.8 Quelle est la motivation derrière le choix du réseau WLAN pour implémenter le processus de localisation ?

R.8 Les réseaux WLANs sont déployés dans beaucoup d'immeubles (hôpitaux, universités, aéroports, centres commerciaux, etc.), ce qui rend la mise en œvre

du processus de localisation simple et moins coûteux.

Q.9 Comment peut-on mesurer les informations de signatures RSS ?

R.9 Pour cela, le logiciel NetStumbler a été installé sur un ordinateur portable. IL permet de mesurer les signaux RSS émis par les points d'accès du réseau WLAN et détectés par la carte WiFi du portable.

Q.10 Quels sont les algorithmes de jumelage ont été examinés dans ce mémoire ?

R.10 L'algorithme des K-plus proches voisins (KNN) et les algorithmes basés sur les réseaux de neurones artificiels de types MLP (Multicouche Perceptron) et GRNN (Generalized Regression Neural Network).

Q.11 Quel est le principe de fonctionnement de l'algorithme KNN ?

R.11 Cet algorithme utilise les K mesures RSS de la base de données de signatures, qui possèdent le plus de similitudes avec la mesure RSS actuelle, en se basant sur le critère de minimisation de la distance Euclidienne. La position du mobile sera déduite en fonction de cette similitude et par rapport aux positions qui correspondent aux K plus proches signatures extraites de la base de données.

Q.12 Quel est le principe de fonctionnement des réseaux de neurones artificiels ?

R.12 Ils permettent de modéliser le comportement non-linéaire entre les signatures RSS et la position du mobile, en se basant sur une structure élémentaire qui est le neurone artificiel, dont la conception est inspirée du fonctionnement du neurone biologique. Leurs modèle est défini par une règle de transmission, une architecture et une règle d'apprentissage.

Q.13 Comment est défini le réseau de neurones de type MLP ?

R.13 Ce réseau est défini par une architecture non-récurrente "feed-forward" et

par un apprentissage supervisé basé sur la rétro-propagation du gradient de l'erreur ("back-propagation") qui minimise l'erreur quadrature moyenne.

Q.14 Comment est défini le réseau de neurones de type GRNN ?

R.14 Ce réseau a une architecture assez semblable à celle des réseaux de neurones à base radiale (RBF), utilisant un apprentissage supérivisé et une architecture multicouche (deux couches de neurones : les neurones de la couche cachée ont une fonction d'activation à base radiale et ceux de la couche de sortie ont une fonction d'activation linéaire).

Q.15 De quoi dépend la précison de localisation de la technique de fingerprinting ?

R.15 Elle dépend essentiellement du nombre de points d'accès du réseau WLAN déployé, du nombre de mesures RSS enregistrées dans la base de données et de l'algorithme de jumelage utilisé.

Q.16 Quelles sont les propriétés qui influence le fonctionnement de l'algorithme de jumelage ?

R.16 Le fonctionnement de l'algorithme de jumelage dépend de la propriété de répétitivité (même signature mesurée pour une même position et pour des temps distincts), de la propriété d'unicité (signatures mesurées distinctes pour des positions distinctes) et de la propriété de généralisation (bonne estimation même s'il n'y a aucune signature contenue dans la base de données qui correspond à celle mesurée).

Q.17 Quelles sont les sources d'erreurs de localisation et de suivi pour la technique de localisation fingerprinting basée sur les signatures RSS d'un réseau WLAN ?

R.17 Les fluctuations des niveaux RSS au cours du temps (dues à la non-stationnarité de l'environnement interne) et le fait que la technique de finger-

printing fournie des estimations statiques (ne tenant pas compte du mouvement du mobile) impliquent une dégradation de la precision de localisation et des discontinuités au niveau de la trajectoire du mobile.

Q.18 Quelle est l'approche proposée dans ce mémoire pour améliorer la localisation de la technique de fingerprinting ?

R.18 L'approche proposée est basée sur l'application du filtrage de Kalman et de ses extensions à la technique de fingerprinting à base du réseau GRNN.

Q.19 Quelle a été la cause du choix du réseau GRNN comme algorithme de jumelage pour la technique de fingerprinting ?

R.19 Car il a donné de meilleures performances de localisation que le réseau MLP et la méthode KNN.

Q.20 Quel est l'avantage d'appliquer le filtrage de Kalman et de ses extensions au cas de la localisation et du suivi ?

R.20 Son avantage est qu'il propose une approche d'estimation dynamique qui tient compte de la mesure, des bruits ambiants, de l'historique et des caractéristiques du mouvement du mobile.

Q.21 Quel est le principe de fonctionnement du filtre de Kalman ?

R.21 Le filtre de Kalman est un estimateur récursif qui nécessite un modèle d'état (caractérisant la dynamique du système) et un modèle d'observation (caractérisant les mesures relatives à l'état du système). Il cherche à estimer l'état du système d'une manière optimale en minimisant l'erreur quadratique moyenne, sous les contraintes d'un système linéaire et des bruits gaussiens.

Q.22 Quel est le principe de fonctionnement du filtre de Kalman étendu ?

R.22 Le filtre de Kalman étendu est une extension du filtre de Kalman classique appliqué dans le cas d'un système non-linéaire avec des bruits gaussiens. Il repose sur une linéarisation au premier ordre, par le développement de Taylor, des modèles d'état et d'observation non-linéaires.

Q.23 Quel est le principe de fonctionnement du filtre de particules ?

R.23 Le filtre de particules est une extension du filtre de Kalman appliqué dans le cas d'un système linéaire et/ou des bruits gaussiens. Il est basé sur l'utilisation des approximations des méthodes de Monte-Carlo pour estimer la densité a posteriori de l'état du système.

Q.24 Quel est l'intérêt d'ajouter une phase de rééchantillonnage dans le filtrage de particules ?

R.24 Le rééchantillonnage permet d'éviter le problème de la dégénérescence du filtre de particules (poids de particules tendent vers des valeurs disparates) en redistribuant les particules ayant un poids significatif.

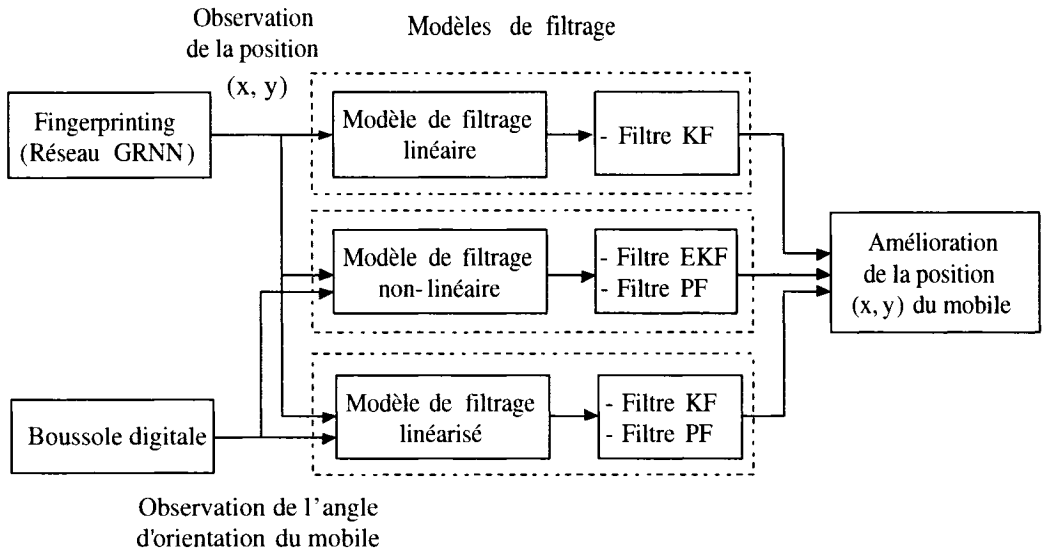
Q.25 Que caractérisent les bruits d'état et de mesure ?

R.25 Le bruit d'état caractérise les aléas du système (erreur de modélisation de la loi d'évolution de l'état du système) et le bruit de mesure caractérise les erreurs de mesures commises par les différents capteurs (observateurs).

Q.26 Quelles sont les variantes de l'approche d'amélioration qui ont été proposées dans ce mémoire ?

R.26 Trois variantes d'amélioration relatives au modèle de filtrage (caractérisant l'état et l'observation de la dynamique d'un piéton) ont été proposées comme suit :

Q.27 Comment a été obtenu le modèle de filtrage linéaire ?



R.27 En caractérisant l'espace d'état (défini par les équations du mouvement linéaire en 2D d'un piéton) et l'espace d'observation (défini par les mesures de la position fournies par la technique de fingerprinting à base du réseau GRNN).

Q.28 Comment a été obtenu le modèle de filtrage non-linéaire ?

R.28 En introduisant dans le modèle de filtrage linéaire l'observation sur l'angle d'orientation du mobile mesurée par une boussole.

Q.29 Quelle est la raison du choix de la boussole ?

R.29 La boussole est un système inertiel de moindre coût par rapport à d'autre systèmes (accéléromètre ou gyroscope), permettant d'observer l'angle d'orientation du mobile (qui est une mesure significative pour décrire le comportement réel du mobile).

Q.30 Comment a été réglé le problème de modélisation de la loi récursive régissant l'angle d'orientation du mobile ?

R.30 En considérant que l'estimation a posteriori de l'angle d'orientation est la même avec la mesure fournie par la boussole.

Q.31 Comment a été obtenu le modèle de filtrage linéarisé ?

R.31 En remplaçant dans le modèle de filtrage non-linéaire l'angle d'orientation par les deux composantes axiales de la vitesse du mobile.

Q.32 Quelles sont les raisons pour lesquelles le modèle de filtrage linéarisé a été proposé ?

R.32 Ce modèle a permis d'appliquer du filtre de Kalman (ce qui n'était pas le cas dans le modèle de filtrage non-linéaire) et de bien modéliser la loi récursive régissant l'angle d'orientation du mobile (en remplaçant cette variable par les deux de vitesse axiales du mobiles).

Q.33 Comment a été caractérisé le bruit d'état ?

R.33 Il a été caractérisé par les écarts-types modélisant les accélérations du piéton suivant x et y.

Q.34 Comment a été caractérisé le bruit de mesure ?

R.34 Il a été caractérisé par la marge de confiance à donner aux estimations de localisation mesurées par le réseau GRNN.

Q.35 Quels ont été les crières de performances de l'approche proposée ?

R.35 La précision de la position de localisation du mobile (en termes d'erreur de position définie par la distance Euclidienne), la fonction de distribution cumulative (CDF) et le temps de traitement d'une estimation de localisation ont été considérés comme critères de performances.

Q.36 Comment peut-on expliquer le fait que le filtre de particules donne les meilleures précisons de localisation que les filtres de Kalman classique et étendu ?

R.36 Ça peut être expliqué par le fait que le filtre de particules essaye de modéliser le comportement réel des bruits affectant l'état du système (en se basant sur l'estimation de la densité de probabilité réelle de l'état du système), alors que dans le cas des filtres de Kalman classique et étendu, les bruits d'état et de mesure sont supposée blancs gaussiens (la densité de probabilité de l'état du système est supposée gaussienne), ce qui ne reflète pas le cas réel.

Q.37 Pourquoi le filtre de particules consomme beaucoup plus de temps pour traiter une estimation donnée ?

R.37 À cause des différents échantillons (particules) générés pour traiter l'estimation considérée.

Q.38 Quelles ont été les améliorations sur la précision de localisation obtenues avec les différentes variantes proposées dans ce mémoire ?

R.38 Elles sont résumées dans la table qui suit :

	Algorithme	Moyenne (m)	Gain filtrage/GRNN (%)
	GRNN	2.68	-
Modèle linéaire	KF	2.08	22.40
Modèle non-linéaire	EKF	1.79	33.21
	PF	1.62	39.18
Modèle linéarisé	KF	1.83	31.75
	PF	1.33	50.37

Q.39 Quels ont été les paramètres qui influencent le fonctionnement de l'approche de localisation et de suivi proposée dans ce mémoire ?

R.39 Le bon fonctionnement de l'approche proposé dépend de ce qui suit :

- Choix du nombre de points d'accès.
- Richesse de la base de données de signatures.
- Valeur du "spread" du réseau GRNN.
- Modélisation des lois récursives régissant la dynamique du mobile.
- Temps d'échantillonnage.
- Variances des bruits d'état et de mesure.
- Nombre de particules pour le filtrage particulaire.

Q.40 Quelles sont les perspectives des travaux futurs ?

R.40 Comme perspectives, il est souhaitable d'observer de nouvelles informations sur le mouvement du mobile (acceleration ou rotation), examiner d'autres algorithmes de localisation (comme par exemple, la méthode SVM), utiliser d'autres technologies de détection pour des fins de localisation dans un environnement (comme par exemple, la technologie UWB), introduire dans le système de localisation un plan électronique de l'environnent de localisation avec un système à interface graphique, implanter un système de localisation sur un circuit numérique (par exemple, les circuits programmables FPGA) ou encore appliquer l'approche proposée dans ce mémoire au cas d'un robot mobile.